SPM8 Manual

The FIL Methods Group
(and honorary members)

John Ashburner
Chun-Chuan Chen
Guillaume Flandin
Rik Henson
Stefan Kiebel
James Kilner
Vladimir Litvak
Rosalyn Moran
Will Penny
Klaas Stephan
Chloe Hutton
Volkmar Glauche
Jérémie Mattout
Christophe Phillips

Functional Imaging Laboratory
Wellcome Trust Centre for Neuroimaging
Institute of Neurology, UCL
12 Queen Square, London WC1N 3BG, UK
May 29, 2009
http://www.fil.ion.ucl.ac.uk/spm/

Contents

I Temporal processing 13

1 Slice Timing 15
- 1.1 Data . . . 16
  - 1.1.1 Session . . . 16
- 1.2 Number of Slices . . . 16
- 1.3 TR . . . 16
- 1.4 TA . . . 16
- 1.5 Slice order . . . 16
- 1.6 Reference Slice . . . 16
- 1.7 Filename Prefix . . . 17

II Spatial processing 19

2 Realign 21
- 2.1 Realign: Estimate . . . 21
  - 2.1.1 Data . . . 21
  - 2.1.2 Estimation Options . . . 22
- 2.2 Realign: Reslice . . . 23
  - 2.2.1 Images . . . 23
  - 2.2.2 Reslice Options . . . 23
- 2.3 Realign: Estimate & Reslice . . . 24
  - 2.3.1 Data . . . 24
  - 2.3.2 Estimation Options . . . 24
  - 2.3.3 Reslice Options . . . 25

3 Realign & Unwarp 27
- 3.1 Data . . . 30
  - 3.1.1 Session . . . 30
- 3.2 Estimation Options . . . 30
  - 3.2.1 Quality . . . 30
  - 3.2.2 Separation . . . 30
  - 3.2.3 Smoothing (FWHM) . . . 30
  - 3.2.4 Num Passes . . . 30
  - 3.2.5 Interpolation . . . 31
  - 3.2.6 Wrapping . . . 31
  - 3.2.7 Weighting . . . 31
- 3.3 Unwarp Estimation Options . . . 31
  - 3.3.1 Basis Functions . . . 31
  - 3.3.2 Regularisation . . . 31
  - 3.3.3 Reg. Factor . . . 31
  - 3.3.4 Jacobian deformations . . . 31
  - 3.3.5 First-order effects . . . 32
  - 3.3.6 Second-order effects . . . 32
  - 3.3.7 Smoothing for unwarp (FWHM) . . . 32

3.3.8 Re-estimate movement params . . . 32
3.3.9 Number of Iterations . . . 32
3.3.10 Taylor expansion point . . . 32
3.4 Unwarp Reslicing Options . . . 32
3.4.1 Resliced images (unwarp)? . . . 32
3.4.2 Interpolation . . . 33
3.4.3 Wrapping . . . 33
3.4.4 Masking . . . 33
3.4.5 Filename Prefix . . . 33

4 Coreg **35**
4.1 Coreg: Estimate . . . 35
4.1.1 Reference Image . . . 35
4.1.2 Source Image . . . 36
4.1.3 Other Images . . . 36
4.1.4 Estimation Options . . . 36
4.2 Coreg: Reslice . . . 36
4.2.1 Image Defining Space . . . 36
4.2.2 Images to Reslice . . . 36
4.2.3 Reslice Options . . . 36
4.3 Coreg: Estimate & Reslice . . . 37
4.3.1 Reference Image . . . 37
4.3.2 Images to Reslice . . . 37
4.3.3 Other Images . . . 37
4.3.4 Estimation Options . . . 37
4.3.5 Reslice Options . . . 38

5 Segment **39**
5.1 Data . . . 40
5.2 Output Files . . . 40
5.2.1 Grey Matter . . . 43
5.2.2 White Matter . . . 43
5.2.3 Cerebro-Spinal Fluid . . . 43
5.2.4 Bias Corrected . . . 43
5.2.5 Clean up any partitions . . . 43
5.3 Custom . . . 43
5.3.1 Tissue probability maps . . . 43
5.3.2 Gaussians per class . . . 44
5.3.3 Affine Regularisation . . . 44
5.3.4 Warping Regularisation . . . 44
5.3.5 Warp Frequency Cutoff . . . 45
5.3.6 Bias regularisation . . . 45
5.3.7 Bias FWHM . . . 45
5.3.8 Sampling distance . . . 45
5.3.9 Masking image . . . 45

6 Normalise **47**
6.1 Normalise: Estimate . . . 48
6.1.1 Data . . . 48
6.1.2 Estimation Options . . . 48
6.2 Normalise: Write . . . 49
6.2.1 Data . . . 49
6.2.2 Writing Options . . . 50
6.3 Normalise: Estimate & Write . . . 50
6.3.1 Data . . . 50
6.3.2 Estimation Options . . . 51
6.3.3 Writing Options . . . 52

7 Smooth **53**
7.1 Images to Smooth . . . 53
7.2 FWHM . . . 53
7.3 Data Type . . . 53
7.4 Filename Prefix . . . 53

III fMRI Statistics **55**

8 fMRI model specification **57**
8.1 Timing parameters . . . 57
8.1.1 Units for design . . . 58
8.1.2 Interscan interval . . . 58
8.1.3 Microtime resolution . . . 58
8.1.4 Microtime onset . . . 58
8.2 Data & Design . . . 59
8.2.1 Subject/Session . . . 60
8.3 Factorial design . . . 61
8.3.1 Factor . . . 62
8.4 Basis Functions . . . 62
8.4.1 Canonical HRF . . . 62
8.4.2 Other basis sets . . . 62
8.5 Model Interactions (Volterra) . . . 63
8.6 Directory . . . 63
8.7 Global normalisation . . . 63
8.8 Explicit mask . . . 63
8.9 Serial correlations . . . 64
8.10 Reviewing your design . . . 64

9 fMRI model estimation **67**
9.1 Select SPM.mat . . . 67
9.2 Method . . . 67
9.2.1 Classical . . . 67
9.2.2 Bayesian 1st-level . . . 68
9.2.3 Bayesian 2nd-level . . . 71
9.3 Output files . . . 72
9.3.1 Classical 1st-level . . . 72
9.3.2 Bayesian 1st-level . . . 72
9.4 Model comparison . . . 72

10 Factorial design specification **75**
10.1 Directory . . . 76
10.2 Design . . . 76
10.2.1 One-sample t-test . . . 76
10.2.2 Two-sample t-test . . . 76
10.2.3 Paired t-test . . . 77
10.2.4 Multiple regression . . . 77
10.2.5 Full factorial . . . 78
10.2.6 Flexible factorial . . . 79
10.3 Covariates . . . 81
10.3.1 Covariate . . . 81
10.4 Masking . . . 82
10.4.1 Threshold masking . . . 82
10.4.2 Implicit Mask . . . 82
10.4.3 Explicit Mask . . . 82
10.5 Global calculation . . . 83
10.5.1 Omit . . . 83

10.5.2 User . . . 83
10.5.3 Mean . . . 83
10.6 Global normalisation . . . 83
10.6.1 Overall grand mean scaling . . . 83
10.6.2 Normalisation . . . 84

IV EEG/MEG **85**

11 SPM for MEG/EEG overview **87**
11.1 Welcome to SPM for M/EEG . . . 87
11.2 Changes from SPM5 to SPM8 . . . 88

12 EEG/MEG preprocessing – Reference **91**
12.1 Conversion of data . . . 91
12.2 Converting arbitrary data . . . 93
12.3 The M/EEG SPM format . . . 93
12.4 Preparing the data after conversion . . . 94
12.5 Integration of SPM and Fieldtrip . . . 96
12.6 Reading of data . . . 97
12.6.1 Syntax . . . 97
12.7 Methods for the `meeg` object . . . 98
12.7.1 Constructor `meeg` . . . 98
12.7.2 display . . . 98
12.7.3 Number methods . . . 98
12.7.4 Reading and manipulation of information . . . 98
12.7.5 Reading of information . . . 100
12.7.6 Manipulation of information . . . 101
12.7.7 struct-like interface . . . 102
12.8 SPM functions . . . 102
12.8.1 Epoching the data: `spm_eeg_epochs` . . . 102
12.8.2 Filtering the data: `spm_eeg_filter` . . . 103
12.8.3 Artefact detection and rejection: `spm_eeg_artefact` . . . 103
12.8.4 Downsampling: `spm_eeg_downsample` . . . 104
12.8.5 Rereferencing: `spm_eeg_montage` . . . 104
12.8.6 Grand mean: `spm_eeg_grandmean` . . . 105
12.8.7 Merge: `spm_eeg_merge` . . . 105
12.8.8 Time-frequency decomposition: `spm_eeg_tf` . . . 105
12.8.9 Averaging: `spm_eeg_average` . . . 105
12.8.10 Contrast of trials: `spm_eeg_weight_epochs` . . . 105
12.9 Displaying data with SPM M/EEG Review . . . 106
12.9.1 Data visualization . . . 106
12.9.2 Source reconstructions visualization . . . 107
12.10 Batching and scripts . . . 108
12.10.1 The new SPM8 batch system . . . 108
12.10.2 Script generation . . . 108

13 Analysis in sensor space **111**

14 3D source reconstruction: Imaging approach **113**
14.1 Introduction . . . 113
14.2 Getting started . . . 114
14.3 Source space modeling . . . 114
14.4 Coregistration . . . 115
14.5 Forward computation (*forward*) . . . 116
14.6 Inverse reconstruction . . . 117
14.7 Summarizing the results of inverse reconstruction as an image . . . 118

14.8 Rendering interface . . . 118
14.9 Group inversion . . . 119
14.10 Appendix: Data structure . . . 119

15 Localization of Equivalent Current Dipoles **121**
15.1 Introduction . . . 121
15.2 Procedure in SPM8 . . . 122
15.2.1 Head and forward model . . . 122
15.2.2 VB-ECD reconstruction . . . 122
15.2.3 Result display . . . 123

16 Dynamic Causal Modelling for M/EEG **125**
16.1 Introduction . . . 125
16.2 Overview . . . 126
16.3 Calling DCM for ERP/ERF . . . 126
16.4 load, save, select model type . . . 127
16.5 Data and design . . . 127
16.6 Electromagnetic model . . . 127
16.7 Neuronal model . . . 128
16.8 Estimation . . . 128
16.9 Results . . . 129
16.10 Steady-State Responses . . . 129
16.10.1 Model specification . . . 129
16.10.2 The Lead-Field . . . 130
16.10.3 Connections . . . 130
16.10.4 Cross Spectral Densities . . . 130
16.10.5 Output and Results . . . 130
16.11 Induced responses . . . 131
16.11.1 Data . . . 131
16.11.2 Electromagnetic model . . . 131
16.11.3 Neuronal model . . . 131
16.11.4 Wavelet transform . . . 131
16.11.5 Results . . . 131
16.12 Phase-coupled responses . . . 131
16.12.1 Data . . . 132
16.12.2 Electromagnetic model . . . 132
16.12.3 Neuronal model . . . 132
16.12.4 Hilbert transform . . . 132
16.12.5 Results . . . 132

V Utilities **133**

17 Display Image **135**
17.1 Image to Display . . . 136

18 Check Registration **139**
18.1 Images to Display . . . 139

19 Image Calculator **141**
19.1 Input Images . . . 141
19.2 Output Filename . . . 141
19.3 Output Directory . . . 141
19.4 Expression . . . 142
19.5 Options . . . 142
19.5.1 Data Matrix . . . 142
19.5.2 Masking . . . 142
19.5.3 Interpolation . . . 142

19.5.4 Data Type . . . 142

20 DICOM Import **143**
20.1 DICOM files . . . 143
20.2 Directory structure for converted files . . . 143
20.3 Output directory . . . 143
20.4 Conversion options . . . 144
20.4.1 Output image format . . . 144
20.4.2 Use ICEDims in filename . . . 144

21 MINC Import **145**
21.1 MINC files . . . 145
21.2 Options . . . 145
21.2.1 Data Type . . . 145
21.2.2 NIFTI Type . . . 145

22 ECAT Import **147**
22.1 ECAT files . . . 147
22.2 Options . . . 147
22.2.1 NIFTI Type . . . 147

23 Deformations **149**
23.1 Composition . . . 149
23.1.1 Imported _sn.mat . . . 149
23.1.2 DARTEL flow . . . 150
23.1.3 Deformation Field . . . 150
23.1.4 Identity . . . 150
23.1.5 Inverse . . . 150
23.1.6 Composition . . . 156
23.2 Save as . . . 163
23.3 Apply to . . . 163
23.4 Interpolation . . . 163

VI Tools **165**

24 FieldMap Toolbox **167**
24.1 Introduction . . . 167
24.2 Creating Field Maps Using the FieldMap GUI . . . 167
24.2.1 Create field map in Hz . . . 167
24.2.2 Create voxel displacement map (VDM) and unwarp EPI . . . 169
24.3 Using the pm_defaults file . . . 171
24.4 Using the SPM8 User Interface . . . 171
24.5 Using FieldMap with multiple sessions or runs . . . 171
24.6 Using the FieldMap in Batch scripts . . . 172
24.7 Using the VDM file with Unwarp . . . 172
24.8 Appendices . . . 172
24.8.1 Processing Hz field maps . . . 172
24.8.2 Converting Hz field map to VDM . . . 173
24.8.3 Matching field map data to EPI data . . . 173

25 New Segment **175**
25.1 Data . . . 176
25.1.1 Channel . . . 176
25.2 Tissues . . . 177
25.2.1 Tissue . . . 177
25.3 Warping . . . 178
25.3.1 Warping Regularisation . . . 178

25.3.2 Affine Regularisation . . . 179
25.3.3 Sampling distance . . . 179
25.3.4 Deformation Fields . . . 179

26 DARTEL Tools **181**
26.1 Initial Import . . . 182
26.1.1 Parameter Files . . . 182
26.1.2 Output Directory . . . 183
26.1.3 Bounding box . . . 183
26.1.4 Voxel size . . . 183
26.1.5 Image option . . . 183
26.1.6 Grey Matter . . . 183
26.1.7 White Matter . . . 183
26.1.8 CSF . . . 183
26.2 Run DARTEL (create Templates) . . . 183
26.2.1 Images . . . 183
26.2.2 Settings . . . 183
26.3 Run DARTEL (existing Templates) . . . 185
26.3.1 Images . . . 185
26.3.2 Settings . . . 185
26.4 Normalise to MNI Space . . . 186
26.4.1 DARTEL Template . . . 186
26.4.2 Select according to . . . 186
26.4.3 Voxel sizes . . . 187
26.4.4 Bounding box . . . 187
26.4.5 Preserve . . . 187
26.4.6 Gaussian FWHM . . . 187
26.5 Create Warped . . . 187
26.5.1 Flow fields . . . 188
26.5.2 Images . . . 188
26.5.3 Modulation . . . 188
26.5.4 Time Steps . . . 188
26.5.5 Interpolation . . . 188
26.6 Jacobian determinants . . . 188
26.6.1 Flow fields . . . 188
26.6.2 Time Steps . . . 188
26.7 Create Inverse Warped . . . 188
26.7.1 Flow fields . . . 189
26.7.2 Images . . . 189
26.7.3 Time Steps . . . 189
26.7.4 Interpolation . . . 189
26.8 Kernel Utilities . . . 189
26.8.1 Kernel from Images . . . 189
26.8.2 Kernel from Flows . . . 190

27 High-Dimensional Warping **191**
27.1 Subjects . . . 191
27.1.1 Subject . . . 191
27.2 Bias Correction Options . . . 192
27.2.1 Iterations . . . 192
27.2.2 Bias FWHM . . . 192
27.2.3 Bias regularisation . . . 192
27.2.4 Levenberg-Marquardt regularisation . . . 192
27.3 Warping Options . . . 192
27.3.1 Iterations . . . 192
27.3.2 Warping regularisation . . . 192

VII Data sets and examples **193**

28 Auditory fMRI data **195**
28.1 Spatial pre-processing . . . 195
28.1.1 Realignment . . . 195
28.1.2 Coregistration . . . 197
28.1.3 Segmentation . . . 198
28.1.4 Normalise . . . 200
28.1.5 Smoothing . . . 202
28.2 Model specification, review and estimation . . . 202
28.2.1 Estimate . . . 206
28.3 Inference . . . 206
28.3.1 Contrast manager . . . 207
28.3.2 Masking . . . 207
28.3.3 Thresholds . . . 208
28.3.4 Files . . . 208
28.3.5 Maximum Intensity Projections . . . 208
28.3.6 Design matrix . . . 210
28.3.7 Statistical tables . . . 210
28.3.8 Plotting responses at a voxel . . . 211
28.3.9 Overlays . . . 212
28.3.10 Miscellaneous . . . 213
28.4 Bayesian analysis . . . 213
28.4.1 Specification . . . 213
28.4.2 Estimation . . . 214
28.4.3 Inference . . . 215

29 Face data **219**
29.1 Spatial pre-processing . . . 219
29.1.1 Display . . . 219
29.1.2 Realignment . . . 219
29.1.3 Slice timing correction . . . 224
29.1.4 Coregistration . . . 224
29.1.5 Segmentation . . . 224
29.1.6 Normalise . . . 227
29.1.7 Smoothing . . . 227
29.2 Modelling categorical responses . . . 229
29.2.1 Estimate . . . 232
29.2.2 Inference for categorical design . . . 232
29.2.3 Statistical tables . . . 233
29.2.4 F-contrasts . . . 234
29.2.5 F-contrasts for testing effects of movement . . . 235
29.3 Modelling parametric responses . . . 237
29.3.1 Estimate . . . 240
29.3.2 Plotting parametric responses . . . 240
29.4 Bayesian analysis . . . 242
29.4.1 Specification . . . 242
29.4.2 Estimation . . . 242
29.4.3 Inference . . . 244

30 Face group data **247**
30.1 Introduction . . . 247
30.2 Data . . . 247
30.3 Canonical HRF . . . 248
30.4 Informed basis set . . . 250
30.4.1 Nonsphericity . . . 252
30.4.2 Informed Results . . . 252

30.4.3 T- and F-contrasts . . . 256
30.5 FIR basis set . . . 258
30.5.1 Nonsphericity again . . . 259
30.5.2 FIR Results . . . 262

31 Verbal Fluency PET data **267**
31.1 Introduction . . . 267
31.2 Single subject . . . 267
31.3 Multiple subjects . . . 268
31.3.1 Subject and Condition design . . . 270
31.3.2 Subject and Time design . . . 270
31.3.3 Subject by Condition design . . . 272
31.3.4 Contrast manager . . . 274
31.3.5 Masking and thresholds . . . 276
31.3.6 MIPs and results tables . . . 277
31.3.7 Small volume correction . . . 279
31.3.8 Extracting data from regions . . . 279
31.3.9 Inclusive Masking . . . 281
31.3.10 Conjunctions . . . 281

32 Dynamic Causal Modeling for fMRI **285**
32.1 Theoretical background . . . 285
32.2 Bayesian model selection . . . 288
32.3 Practical example . . . 289
32.3.1 Defining the GLM . . . 290
32.3.2 Extracting time series . . . 290
32.3.3 Specifying and estimating the DCM . . . 291
32.3.4 Comparing models . . . 293

33 Mismatch negativity data **297**
33.1 Preprocessing . . . 297
33.1.1 Convert . . . 297
33.1.2 Montage . . . 297
33.1.3 Epoch . . . 298
33.1.4 Downsample . . . 298
33.1.5 Filter . . . 298
33.1.6 Artefacts . . . 300
33.1.7 Averaging . . . 300
33.1.8 History . . . 301
33.2 Sensor space analysis . . . 301
33.3 Source reconstruction . . . 304
33.3.1 Mesh . . . 304
33.3.2 Coregister . . . 304
33.3.3 Forward model . . . 305
33.3.4 Invert . . . 305
33.4 Dynamic Causal Modeling . . . 308

34 Using DARTEL **313**
34.1 Using DARTEL for VBM . . . 313
34.1.1 Using Spatial→Segment and DARTEL Tools→Initial Import . . . 314
34.1.2 Using Tools→New Segment . . . 316
34.1.3 Using DARTEL Tools→Run DARTEL (create Template) . . . 316
34.1.4 Using DARTEL Tools→Normalise to MNI Space . . . 318
34.2 Spatially normalising functional data to MNI space . . . 318
34.2.1 An alternative approach for using DARTEL to spatially normalise to MNI Space . . . 321
34.3 Warping Images to Existing Templates . . . 323

34.4 Warping one individual to match another . . . 323

VIII Batch Interface **327**

35 Batch interface **329**
35.1 Batch tutorial - single subject . . . 329
35.1.1 Study specific input data . . . 330
35.1.2 Necessary processing steps . . . 330
35.1.3 Add modules to the batch . . . 330
35.1.4 Configure subject-independent data . . . 331
35.1.5 Data flow . . . 333
35.1.6 Entering subject-specific data . . . 335
35.2 Advanced features . . . 336
35.2.1 Multiple sessions . . . 336
35.2.2 Processing multiple subjects in GUI . . . 336
35.2.3 Command line interface . . . 336
35.2.4 Modifying a saved job . . . 338
35.3 SPM5 to matlabbatch transition guide . . . 338
35.3.1 Code Reorganisation . . . 339
35.3.2 Interfaces between SPM and Matlabbatch . . . 339
35.4 Configuration Code Details . . . 339
35.4.1 Virtual Outputs . . . 340
35.4.2 SPM Startup . . . 340
35.4.3 Defaults Settings . . . 340
35.4.4 Toolbox Migration . . . 340
35.5 Utilities . . . 341
35.5.1 Batch Utilities . . . 341
35.5.2 Matlab Code Generation . . . 341
35.5.3 Configuration Management . . . 341

IX Bibliography **343**

Part I

Temporal processing

Chapter 1

Slice Timing

Contents

1.1 Data 16
1.1.1 Session 16
1.2 Number of Slices 16
1.3 TR 16
1.4 TA 16
1.5 Slice order 16
1.6 Reference Slice 16
1.7 Filename Prefix 17

Note that this option is likely to be removed in future. The authors of SPM do not generally suggest that this correction should be used, but the option is still retained for the few people who like to use it.

Correct differences in image acquisition time between slices. Slice-time corrected files are prepended with an 'a'.

Note: The sliceorder arg that specifies slice acquisition order is a vector of N numbers, where N is the number of slices per volume. Each number refers to the position of a slice within the image file. The order of numbers within the vector is the temporal order in which those slices were acquired. To check the order of slices within an image file, use the SPM Display option and move the cross-hairs to a voxel co-ordinate of z=1. This corresponds to a point in the first slice of the volume.

The function corrects differences in slice acquisition times. This routine is intended to correct for the staggered order of slice acquisition that is used during echo-planar scanning. The correction is necessary to make the data on each slice correspond to the same point in time. Without correction, the data on one slice will represent a point in time as far removed as 1/2 the TR from an adjacent slice (in the case of an interleaved sequence).

This routine "shifts" a signal in time to provide an output vector that represents the same (continuous) signal sampled starting either later or earlier. This is accomplished by a simple shift of the phase of the sines that make up the signal. Recall that a Fourier transform allows for a representation of any signal as the linear combination of sinusoids of different frequencies and phases. Effectively, we will add a constant to the phase of every frequency, shifting the data in time.

Shifter - This is the filter by which the signal will be convolved to introduce the phase shift. It is constructed explicitly in the Fourier domain. In the time domain, it may be described as an impulse (delta function) that has been shifted in time the amount described by TimeShift. The correction works by lagging (shifting forward) the time-series data on each slice using sinc-interpolation. This results in each time series having the values that would have been obtained

had the slice been acquired at the same time as the reference slice. To make this clear, consider a neural event (and ensuing hemodynamic response) that occurs simultaneously on two adjacent slices. Values from slice "A" are acquired starting at time zero, simultaneous to the neural event, while values from slice "B" are acquired one second later. Without correction, the "B" values will describe a hemodynamic response that will appear to have began one second EARLIER on the "B" slice than on slice "A". To correct for this, the "B" values need to be shifted towards the Right, i.e., towards the last value.

This correction assumes that the data are band-limited (i.e. there is no meaningful information present in the data at a frequency higher than that of the Nyquist). This assumption is support by the study of Josephs et al (1997, NeuroImage) that obtained event-related data at an effective TR of 166 msecs. No physio-logical signal change was present at frequencies higher than our typical Nyquist (0.25 HZ).

Written by Darren Gitelman at Northwestern U., 1998. Based (in large part) on ACQCORRECT.PRO from Geoff Aguirre and Eric Zarahn at U. Penn.

1.1 Data

Subjects or sessions. The same parameters specified below will be applied to all sessions.

1.1.1 Session

Select images to acquisition correct.

1.2 Number of Slices

Enter the number of slices

1.3 TR

Enter the TR in seconds

1.4 TA

The TA (in seconds) must be entered by the user. It is usually calculated as TR-(TR/nslices). You can simply enter this equation with the variables replaced by appropriate numbers.

1.5 Slice order

Enter the slice order. Bottom slice = 1. Sequence types and examples of code to enter are given below.

ascending (first slice=bottom): [1:1:nslices]
descending (first slice=top): [nslices:-1:1]
interleaved (middle-top):
for k = 1:nslices,
round((nslices-k)/2 + (rem((nslices-k),2) * (nslices - 1)/2)) + 1,
end
interleaved (bottom -> up): [1:2:nslices 2:2:nslices]
interleaved (top -> down): [nslices:-2:1, nslices-1:-2:1]

1.6 Reference Slice

Enter the reference slice

1.7 Filename Prefix

Specify the string to be prepended to the filenames of the smoothed image file(s). Default prefix is 'a'.

Part II

Spatial processing

Chapter 2

Realign

Contents

2.1 Realign: Estimate 21
2.1.1 Data 21
2.1.2 Estimation Options 22
2.2 Realign: Reslice 23
2.2.1 Images 23
2.2.2 Reslice Options 23
2.3 Realign: Estimate & Reslice 24
2.3.1 Data 24
2.3.2 Estimation Options 24
2.3.3 Reslice Options 25

Within-subject registration of image time series.

2.1 Realign: Estimate

This routine realigns a time-series of images acquired from the same subject using a least squares approach and a 6 parameter (rigid body) spatial transformation [24]. The first image in the list specified by the user is used as a reference to which all subsequent scans are realigned. The reference scan does not have to the the first chronologically and it may be wise to chose a "representative scan" in this role.

The aim is primarily to remove movement artefact in fMRI and PET time-series (or more generally longitudinal studies). The headers are modified for each of the input images, such that. they reflect the relative orientations of the data. The details of the transformation are displayed in the results window as plots of translation and rotation. A set of realignment parameters are saved for each session, named rp_*.txt. These can be modelled as confounds within the general linear model [24].

2.1.1 Data

Add new sessions for this subject. In the coregistration step, the sessions are first realigned to each other, by aligning the first scan from each session to the first scan of the first session. Then the images within each session are aligned to the first image of the session. The parameter estimation is performed this way because it is assumed (rightly or not) that there may be systematic differences in the images between sessions.

Session

Select scans for this session. In the coregistration step, the sessions are first realigned to each other, by aligning the first scan from each session to the first scan of the first session. Then the images within each session are aligned to the first image of the session. The parameter estimation is performed this way because it is assumed (rightly or not) that there may be systematic differences in the images between sessions.

2.1.2 Estimation Options

Various registration options. If in doubt, simply keep the default values.

Quality

Quality versus speed trade-off. Highest quality (1) gives most precise results, whereas lower qualities gives faster realignment. The idea is that some voxels contribute little to the estimation of the realignment parameters. This parameter is involved in selecting the number of voxels that are used.

Separation

The separation (in mm) between the points sampled in the reference image. Smaller sampling distances gives more accurate results, but will be slower.

Smoothing (FWHM)

The FWHM of the Gaussian smoothing kernel (mm) applied to the images before estimating the realignment parameters.

* PET images typically use a 7 mm kernel.
* MRI images typically use a 5 mm kernel.

Num Passes

Register to first: Images are registered to the first image in the series. Register to mean: A two pass procedure is used in order to register the images to the mean of the images after the first realignment.

PET images are typically registered to the mean. This is because PET data are more noisy than fMRI and there are fewer of them, so time is less of an issue.

MRI images are typically registered to the first image. The more accurate way would be to use a two pass procedure, but this probably wouldn't improve the results so much and would take twice as long to run.

Interpolation

The method by which the images are sampled when estimating the optimum transformation. Higher degree interpolation methods provide the better interpolation, but they are slower because they use more neighbouring voxels [69, 70, 71].

Wrapping

This indicates which directions in the volumes the values should wrap around in. For example, in MRI scans, the images wrap around in the phase encode direction, so (e.g.) the subject's nose may poke into the back of the subject's head. These are typically:

No wrapping - for PET or images that have already been spatially transformed. Also the recommended option if you are not really sure.

Wrap in Y - for (un-resliced) MRI where phase encoding is in the Y direction (voxel space).

Weighting

The option of providing a weighting image to weight each voxel of the reference image differently when estimating the realignment parameters. The weights are proportional to the inverses of the standard deviations. This would be used, for example, when there is a lot of extra-brain motion - e.g., during speech, or when there are serious artifacts in a particular region of the images.

2.2 Realign: Reslice

This function reslices a series of registered images such that they match the first image selected voxel-for-voxel. The resliced images are named the same as the originals, except that they are prefixed by 'r'.

2.2.1 Images

Select scans to reslice to match the first.

2.2.2 Reslice Options

Various reslicing options. If in doubt, simply keep the default values.

Resliced images

All Images (1..n) : This reslices all the images - including the first image selected - which will remain in its original position.

Images 2..n : Reslices images 2..n only. Useful for if you wish to reslice (for example) a PET image to fit a structural MRI, without creating a second identical MRI volume.

All Images + Mean Image : In addition to reslicing the images, it also creates a mean of the resliced image.

Mean Image Only : Creates the mean resliced image only.

Interpolation

The method by which the images are sampled when being written in a different space.Nearest Neighbour is fastest, but not recommended for image realignment. Bilinear Interpolation is probably OK for PET, but not so suitable for fMRI because higher degree interpolation generally gives better results [69, 70, 71]. Although higher degree methods provide better interpolation, but they are slower because they use more neighbouring voxels. Fourier Interpolation [21, 18] is another option, but note that it is only implemented for purely rigid body transformations. Voxel sizes must all be identical and isotropic.

Wrapping

This indicates which directions in the volumes the values should wrap around in. For example, in MRI scans, the images wrap around in the phase encode direction, so (e.g.) the subject's nose may poke into the back of the subject's head. These are typically:

No wrapping - for PET or images that have already been spatially transformed.

Wrap in Y - for (un-resliced) MRI where phase encoding is in the Y direction (voxel space).

Masking

Because of subject motion, different images are likely to have different patterns of zeros from where it was not possible to sample data. With masking enabled, the program searches through the whole time series looking for voxels which need to be sampled from outside the original images. Where this occurs, that voxel is set to zero for the whole set of images (unless the image format can represent NaN, in which case NaNs are used where possible).

Filename Prefix

Specify the string to be prepended to the filenames of the resliced image file(s). Default prefix is 'r'.

2.3 Realign: Estimate & Reslice

This routine realigns a time-series of images acquired from the same subject using a least squares approach and a 6 parameter (rigid body) spatial transformation [24]. The first image in the list specified by the user is used as a reference to which all subsequent scans are realigned. The reference scan does not have to be the first chronologically and it may be wise to chose a "representative scan" in this role.

The aim is primarily to remove movement artefact in fMRI and PET time-series (or more generally longitudinal studies) [5]. The headers are modified for each of the input images, such that. they reflect the relative orientations of the data. The details of the transformation are displayed in the results window as plots of translation and rotation. A set of realignment parameters are saved for each session, named rp_*.txt. After realignment, the images are resliced such that they match the first image selected voxel-for-voxel. The resliced images are named the same as the originals, except that they are prefixed by 'r'.

2.3.1 Data

Add new sessions for this subject. In the coregistration step, the sessions are first realigned to each other, by aligning the first scan from each session to the first scan of the first session. Then the images within each session are aligned to the first image of the session. The parameter estimation is performed this way because it is assumed (rightly or not) that there may be systematic differences in the images between sessions.

Session

Select scans for this session. In the coregistration step, the sessions are first realigned to each other, by aligning the first scan from each session to the first scan of the first session. Then the images within each session are aligned to the first image of the session. The parameter estimation is performed this way because it is assumed (rightly or not) that there may be systematic differences in the images between sessions.

2.3.2 Estimation Options

Various registration options. If in doubt, simply keep the default values.

Quality

Quality versus speed trade-off. Highest quality (1) gives most precise results, whereas lower qualities gives faster realignment. The idea is that some voxels contribute little to the estimation of the realignment parameters. This parameter is involved in selecting the number of voxels that are used.

Separation

The separation (in mm) between the points sampled in the reference image. Smaller sampling distances gives more accurate results, but will be slower.

Smoothing (FWHM)

The FWHM of the Gaussian smoothing kernel (mm) applied to the images before estimating the realignment parameters.

* PET images typically use a 7 mm kernel.
* MRI images typically use a 5 mm kernel.

Num Passes

Register to first: Images are registered to the first image in the series. Register to mean: A two pass procedure is used in order to register the images to the mean of the images after the first realignment.

PET images are typically registered to the mean. This is because PET data are more noisy than fMRI and there are fewer of them, so time is less of an issue.

MRI images are typically registered to the first image. The more accurate way would be to use a two pass procedure, but this probably wouldn't improve the results so much and would take twice as long to run.

Interpolation

The method by which the images are sampled when estimating the optimum transformation. Higher degree interpolation methods provide the better interpolation, but they are slower because they use more neighbouring voxels [69, 70, 71].

Wrapping

This indicates which directions in the volumes the values should wrap around in. For example, in MRI scans, the images wrap around in the phase encode direction, so (e.g.) the subject's nose may poke into the back of the subject's head. These are typically:

No wrapping - for PET or images that have already been spatially transformed. Also the recommended option if you are not really sure.

Wrap in Y - for (un-resliced) MRI where phase encoding is in the Y direction (voxel space).

Weighting

The option of providing a weighting image to weight each voxel of the reference image differently when estimating the realignment parameters. The weights are proportional to the inverses of the standard deviations. This would be used, for example, when there is a lot of extra-brain motion - e.g., during speech, or when there are serious artifacts in a particular region of the images.

2.3.3 Reslice Options

Various reslicing options. If in doubt, simply keep the default values.

Resliced images

All Images (1..n) : This reslices all the images - including the first image selected - which will remain in its original position.

Images 2..n : Reslices images 2..n only. Useful for if you wish to reslice (for example) a PET image to fit a structural MRI, without creating a second identical MRI volume.

All Images + Mean Image : In addition to reslicing the images, it also creates a mean of the resliced image.

Mean Image Only : Creates the mean resliced image only.

Interpolation

The method by which the images are sampled when being written in a different space.Nearest Neighbour is fastest, but not recommended for image realignment. Bilinear Interpolation is probably OK for PET, but not so suitable for fMRI because higher degree interpolation generally gives better results [69, 70, 71]. Although higher degree methods provide better interpolation, but they are slower because they use more neighbouring voxels. Fourier Interpolation [21, 18] is another option, but note that it is only implemented for purely rigid body transformations. Voxel sizes must all be identical and isotropic.

Wrapping

This indicates which directions in the volumes the values should wrap around in. For example, in MRI scans, the images wrap around in the phase encode direction, so (e.g.) the subject's nose may poke into the back of the subject's head. These are typically:

No wrapping - for PET or images that have already been spatially transformed.

Wrap in Y - for (un-resliced) MRI where phase encoding is in the Y direction (voxel space).

Masking

Because of subject motion, different images are likely to have different patterns of zeros from where it was not possible to sample data. With masking enabled, the program searches through the whole time series looking for voxels which need to be sampled from outside the original images. Where this occurs, that voxel is set to zero for the whole set of images (unless the image format can represent NaN, in which case NaNs are used where possible).

Filename Prefix

Specify the string to be prepended to the filenames of the resliced image file(s). Default prefix is 'r'.

Chapter 3

Realign & Unwarp

Contents

3.1 Data . . . 30
3.1.1 Session . . . 30
3.2 Estimation Options . . . 30
3.2.1 Quality . . . 30
3.2.2 Separation . . . 30
3.2.3 Smoothing (FWHM) . . . 30
3.2.4 Num Passes . . . 30
3.2.5 Interpolation . . . 31
3.2.6 Wrapping . . . 31
3.2.7 Weighting . . . 31
3.3 Unwarp Estimation Options . . . 31
3.3.1 Basis Functions . . . 31
3.3.2 Regularisation . . . 31
3.3.3 Reg. Factor . . . 31
3.3.4 Jacobian deformations . . . 31
3.3.5 First-order effects . . . 32
3.3.6 Second-order effects . . . 32
3.3.7 Smoothing for unwarp (FWHM) . . . 32
3.3.8 Re-estimate movement params . . . 32
3.3.9 Number of Iterations . . . 32
3.3.10 Taylor expansion point . . . 32
3.4 Unwarp Reslicing Options . . . 32
3.4.1 Resliced images (unwarp)? . . . 32
3.4.2 Interpolation . . . 33
3.4.3 Wrapping . . . 33
3.4.4 Masking . . . 33
3.4.5 Filename Prefix . . . 33

Within-subject registration and unwarping of time series.

The realignment part of this routine realigns a time-series of images acquired from the same subject using a least squares approach and a 6 parameter (rigid body) spatial transformation. The first image in the list specified by the user is used as a reference to which all subsequent scans are realigned. The reference scan does not have to the the first chronologically and it may be wise to chose a "representative scan" in this role.

The aim is primarily to remove movement artefact in fMRI and PET time-series (or more generally longitudinal studies). ".mat" files are written for each of the input images. The details of the transformation are displayed in the results window as plots of translation and rotation. A set of realignment parameters are saved for each session, named rp_*.txt.

In the coregistration step, the sessions are first realigned to each other, by aligning the first scan from each session to the first scan of the first session. Then the images within each session are aligned to the first image of the session. The parameter estimation is performed this way because it is assumed (rightly or not) that there may be systematic differences in the images between sessions.

The paper [2] is unfortunately a bit old now and describes none of the newer features. Hopefully we'll have a second paper out any decade now.

See also spm_uw_estimate.m for a detailed description of the implementation. Even after realignment there is considerable variance in fMRI time series that covary with, and is most probably caused by, subject movements [2]. It is also the case that this variance is typically large compared to experimentally induced variance. Anyone interested can include the estimated movement parameters as covariates in the design matrix, and take a look at an F-contrast encompassing those columns. It is quite dramatic. The result is loss of sensitivity, and if movements are correlated to task specificity. I.e. we may mistake movement induced variance for true activations. The problem is well known, and several solutions have been suggested. A quite pragmatic (and conservative) solution is to include the estimated movement parameters (and possibly squared) as covariates in the design matrix. Since we typically have loads of degrees of freedom in fMRI we can usually afford this. The problems occur when movements are correlated with the task, since the strategy above will discard "good" and "bad" variance alike (i.e. remove also "true" activations.

The "covariate" strategy described above was predicated on a model where variance was assumed to be caused by "spin history" effects, but will work pretty much equally good/bad regardless of what the true underlying cause is. Others have assumed that the residual variance is caused mainly by errors introduced by the interpolation kernel in the resampling step of the realignment. One has tried to solve this through higher order resampling (huge Sinc kernels, or k-space resampling). Unwarp is based on a different hypothesis regarding the residual variance. EPI images are not particularly faithful reproductions of the object, and in particular there are severe geometric distortions in regions where there is an air-tissue interface (e.g. orbitofrontal cortex and the anterior medial temporal lobes). In these areas in particular the observed image is a severely warped version of reality, much like a funny mirror at a fair ground. When one moves in front of such a mirror ones image will distort in different ways and ones head may change from very elongated to seriously flattened. If we were to take digital snapshots of the reflection at these different positions it is rather obvious that realignment will not suffice to bring them into a common space.

The situation is similar with EPI images, and an image collected for a given subject position will not be identical to that collected at another. We call this effect susceptibility-by-movement interaction. Unwarp is predicated on the assumption that the susceptibility-by- movement interaction is responsible for a sizable part of residual movement related variance.

Assume that we know how the deformations change when the subject changes position (i.e. we know the derivatives of the deformations with respect to subject position). That means that for a given time series and a given set of subject movements we should be able to predict the "shape changes" in the object and the ensuing variance in the time series. It also means that, in principle, we should be able to formulate the inverse problem, i.e. given the observed variance (after realignment) and known (estimated) movements we should be able to estimate how deformations change with subject movement. We have made an attempt at formulating such an inverse model, and at solving for the "derivative fields". A deformation field can be thought of as little vectors at each position in space showing how that particular location has been deflected. A "derivative field" is then the rate of change of those vectors with respect to subject movement. Given these "derivative fields" we should be able to remove the variance caused by the susceptibility-by-movement interaction. Since the underlying model is so restricted we would also expect experimentally induced variance to be preserved. Our experiments have also shown this to be true.

In theory it should be possible to estimate also the "static" deformation field, yielding an

unwarped (to some true geometry) version of the time series. In practise that doesn't really seem to work. Hence, the method deals only with residual movement related variance induced by the susceptibility-by-movement interaction. This means that the time-series will be undistorted to some "average distortion" state rather than to the true geometry. If one wants additionally to address the issue of anatomical fidelity one should combine Unwarp with a measured fieldmap.

The description above can be thought of in terms of a Taylor expansion of the field as a function of subject movement. Unwarp alone will estimate the first (and optionally second, see below) order terms of this expansion. It cannot estimate the zeroth order term (the distortions common to all scans in the time series) since that doesn't introduce (almost) any variance in the time series. The measured fieldmap takes the role of the zeroth order term. Refer to the FieldMap toolbox and the documents FieldMap.man and FieldMap_principles.man for a description of how to obtain fieldmaps in the format expected by Unwarp.

If we think of the field as a function of subject movement it should in principle be a function of six variables since rigid body movement has six degrees of freedom. However, the physics of the problem tells us that the field should not depend on translations nor on rotation in a plane perpendicular to the magnetic flux. Hence it should in principle be sufficient to model the field as a function of out-of-plane rotations (i.e. pitch and roll). One can object to this in terms of the effects of shimming (object no longer immersed in a homogenous field) that introduces a dependence on all movement parameters. In addition SPM/Unwarp cannot really tell if the transversal slices it is being passed are really perpendicular to the flux or not. In practice it turns out thought that it is never (at least we haven't seen any case) necessary to include more than Pitch and Roll. This is probably because the individual movement parameters are typically highly correlated anyway, which in turn is probably because most heads that we scan are attached to a neck around which rotations occur. On the subject of Taylor expansion we should mention that there is the option to use a second-order expansion (through the defaults) interface. This implies estimating also the rate-of-change w.r.t. to some movement parameter of the rate-of-change of the field w.r.t. some movement parameter (colloquially known as a second derivative). It can be quite interesting to watch (and it is amazing that it is possible) but rarely helpful/necessary.

In the defaults there is also an option to include Jacobian intensity modulation when estimating the fields. "Jacobian intensity modulation" refers to the dilution/concentration of intensity that ensue as a consequence of the distortions. Think of a semi-transparent coloured rubber sheet that you hold against a white background. If you stretch a part of the sheet (induce distortions) you will see the colour fading in that particular area. In theory it is a brilliant idea to include also these effects when estimating the field (see e.g. Andersson et al, NeuroImage 20:870-888). In practice for this specific problem it is NOT a good idea.

It should be noted that this is a method intended to correct data afflicted by a particular problem. If there is little movement in your data to begin with this method will do you little good. If on the other hand there is appreciable movement in your data (>1deg) it will remove some of that unwanted variance. If, in addition, movements are task related it will do so without removing all your "true" activations. The method attempts to minimise total (across the image volume) variance in the data set. It should be realised that while (for small movements) a rather limited portion of the total variance is removed, the susceptibility-by-movement interaction effects are quite localised to "problem" areas. Hence, for a subset of voxels in e.g. frontal-medial and orbitofrontal cortices and parts of the temporal lobes the reduction can be quite dramatic (>90). The advantages of using Unwarp will also depend strongly on the specifics of the scanner and sequence by which your data has been acquired. When using the latest generation scanners distortions are typically quite small, and distortion-by-movement interactions consequently even smaller. A small check list in terms of distortions is

a) Fast gradients->short read-out time->small distortions

b) Low field (i.e. <3T)->small field changes->small distortions

c) Low res (64x64)->short read-out time->small distortions

d) SENSE/SMASH->short read-out time->small distortions

If you can tick off all points above chances are you have minimal distortions to begin with and you can say "sod Unwarp" (but not to our faces!).

3.1 Data

Data sessions to unwarp.

3.1.1 Session

Only add similar session data to a realign+unwarp branch, i.e., choose Data or Data+phase map for all sessions, but don't use them interchangeably.

In the coregistration step, the sessions are first realigned to each other, by aligning the first scan from each session to the first scan of the first session. Then the images within each session are aligned to the first image of the session. The parameter estimation is performed this way because it is assumed (rightly or not) that there may be systematic differences in the images between sessions.

Images

Select scans for this session.

In the coregistration step, the sessions are first realigned to each other, by aligning the first scan from each session to the first scan of the first session. Then the images within each session are aligned to the first image of the session. The parameter estimation is performed this way because it is assumed (rightly or not) that there may be systematic differences in the images between sessions.

Phase map (vdm* file)

Select pre-calculated phase map, or leave empty for no phase correction. The vdm* file is assumed to be already in alignment with the first scan of the first session.

3.2 Estimation Options

Various registration options that could be modified to improve the results. Whenever possible, the authors of SPM try to choose reasonable settings, but sometimes they can be improved.

3.2.1 Quality

Quality versus speed trade-off. Highest quality (1) gives most precise results, whereas lower qualities gives faster realignment. The idea is that some voxels contribute little to the estimation of the realignment parameters. This parameter is involved in selecting the number of voxels that are used.

3.2.2 Separation

The separation (in mm) between the points sampled in the reference image. Smaller sampling distances gives more accurate results, but will be slower.

3.2.3 Smoothing (FWHM)

The FWHM of the Gaussian smoothing kernel (mm) applied to the images before estimating the realignment parameters.

* PET images typically use a 7 mm kernel.
* MRI images typically use a 5 mm kernel.

3.2.4 Num Passes

Register to first: Images are registered to the first image in the series. Register to mean: A two pass procedure is used in order to register the images to the mean of the images after the first realignment.

* PET images are typically registered to the mean.

* MRI images are typically registered to the first image.

3.2.5 Interpolation

The method by which the images are sampled when being written in a different space.

Nearest Neighbour - Fastest, but not normally recommended.

Bilinear Interpolation - OK for PET, or realigned fMRI.

B-spline Interpolation [69] - Better quality (but slower) interpolation, especially with higher degree splines. Do not use B-splines when there is any region of NaN or Inf in the images.

3.2.6 Wrapping

These are typically:

No wrapping - for images that have already
been spatially transformed.
Wrap in Y - for (un-resliced) MRI where phase encoding
is in the Y direction (voxel space).

3.2.7 Weighting

The option of providing a weighting image to weight each voxel of the reference image differently when estimating the realignment parameters. The weights are proportional to the inverses of the standard deviations. For example, when there is a lot of extra-brain motion - e.g., during speech, or when there are serious artifacts in a particular region of the images.

3.3 Unwarp Estimation Options

Various registration & unwarping estimation options.

3.3.1 Basis Functions

Number of basis functions to use for each dimension. If the third dimension is left out, the order for that dimension is calculated to yield a roughly equal spatial cut-off in all directions. Default: [12 12 *]

3.3.2 Regularisation

Unwarp looks for the solution that maximises the likelihood (minimises the variance) while simultaneously maximising the smoothness of the estimated field (c.f. Lagrange multipliers). This parameter determines how to balance the compromise between these (i.e. the value of the multiplier). Test it on your own data (if you can be bothered) or go with the defaults.

Regularisation of derivative fields is based on the regorder'th (spatial) derivative of the field. The choices are 0, 1, 2, or 3. Default: 1

3.3.3 Reg. Factor

Regularisation factor. Default: Medium.

3.3.4 Jacobian deformations

In the defaults there is also an option to include Jacobian intensity modulation when estimating the fields. "Jacobian intensity modulation" refers to the dilution/concentration of intensity that ensue as a consequence of the distortions. Think of a semi-transparent coloured rubber sheet that you hold against a white background. If you stretch a part of the sheet (induce distortions) you will see the colour fading in that particular area. In theory it is a brilliant idea to include also these effects when estimating the field (see e.g. Andersson et al, NeuroImage 20:870-888). In practice for this specific problem it is NOT a good idea. Default: No

3.3.5 First-order effects

Theoretically (ignoring effects of shimming) one would expect the field to depend only on subject out-of-plane rotations. Hence the default choice ("Pitch and Roll", i.e., [4 5]). Go with that unless you have very good reasons to do otherwise

Vector of first order effects to model. Movements to be modelled are referred to by number. 1= x translation; 2= y translation; 3= z translation 4 = x rotation, 5 = y rotation and 6 = z rotation.

To model pitch & roll enter: [4 5]

To model all movements enter: [1:6]

Otherwise enter a customised set of movements to model

3.3.6 Second-order effects

List of second order terms to model second derivatives of. This is entered as a vector of movement parameters similar to first order effects, or leave blank for NONE

Movements to be modelled are referred to by number:

1= x translation; 2= y translation; 3= z translation 4 = x rotation, 5 = y rotation and 6 = z rotation.

To model the interaction of pitch & roll enter: [4 5]

To model all movements enter: [1:6]

The vector will be expanded into an n x 2 matrix of effects. For example [4 5] will be expanded to:

[4 4
4 5
5 5]

3.3.7 Smoothing for unwarp (FWHM)

FWHM (mm) of smoothing filter applied to images prior to estimation of deformation fields.

3.3.8 Re-estimate movement params

Re-estimation means that movement-parameters should be re-estimated at each unwarping iteration. Default: Yes.

3.3.9 Number of Iterations

Maximum number of iterations. Default: 5.

3.3.10 Taylor expansion point

Point in position space to perform Taylor-expansion around. Choices are ('First', 'Last' or 'Average'). 'Average' should (in principle) give the best variance reduction. If a field-map acquired before the time-series is supplied then expansion around the 'First' MIGHT give a slightly better average geometric fidelity.

3.4 Unwarp Reslicing Options

Various registration & unwarping estimation options.

3.4.1 Resliced images (unwarp)?

All Images (1..n)

This reslices and unwarps all the images.

All Images + Mean Image

In addition to reslicing the images, it also creates a mean of the resliced images.

3.4.2 Interpolation

The method by which the images are sampled when being written in a different space.
Nearest Neighbour - Fastest, but not normally recommended.
Bilinear Interpolation - OK for PET, or realigned fMRI. B-spline Interpolation[69]
- Better quality (but slower) interpolation, especially with higher degree splines. Do not use B-splines when there is any region of NaN or Inf in the images.

3.4.3 Wrapping

These are typically:
No wrapping - for PET or images that have already been spatially transformed.
Wrap in Y - for (un-resliced) MRI where phase encoding is in the Y direction (voxel space).

3.4.4 Masking

Because of subject motion, different images are likely to have different patterns of zeros from where it was not possible to sample data. With masking enabled, the program searches through the whole time series looking for voxels which need to be sampled from outside the original images. Where this occurs, that voxel is set to zero for the whole set of images (unless the image format can represent NaN, in which case NaNs are used where possible).

3.4.5 Filename Prefix

Specify the string to be prepended to the filenames of the smoothed image file(s). Default prefix is 'u'.

Chapter 4

Coreg

Contents

4.1 Coreg: Estimate . . . 35
4.1.1 Reference Image . . . 35
4.1.2 Source Image . . . 36
4.1.3 Other Images . . . 36
4.1.4 Estimation Options . . . 36
4.2 Coreg: Reslice . . . 36
4.2.1 Image Defining Space . . . 36
4.2.2 Images to Reslice . . . 36
4.2.3 Reslice Options . . . 36
4.3 Coreg: Estimate & Reslice . . . 37
4.3.1 Reference Image . . . 37
4.3.2 Images to Reslice . . . 37
4.3.3 Other Images . . . 37
4.3.4 Estimation Options . . . 37
4.3.5 Reslice Options . . . 38

Within-subject registration using a rigid-body model. A rigid-body transformation (in 3D) can be parameterised by three translations and three rotations about the different axes.

You get the options of estimating the transformation, reslicing images according to some rigid-body transformations, or estimating and applying rigid-body transformations.

4.1 Coreg: Estimate

The registration method used here is based on work by Collignon et al [17]. The original interpolation method described in this paper has been changed in order to give a smoother cost function. The images are also smoothed slightly, as is the histogram. This is all in order to make the cost function as smooth as possible, to give faster convergence and less chance of local minima.

At the end of coregistration, the voxel-to-voxel affine transformation matrix is displayed, along with the histograms for the images in the original orientations, and the final orientations. The registered images are displayed at the bottom.

Registration parameters are stored in the headers of the "source" and the "other" images.

4.1.1 Reference Image

This is the image that is assumed to remain stationary (sometimes known as the target or template image), while the source image is moved to match it.

4.1.2 Source Image

This is the image that is jiggled about to best match the reference.

4.1.3 Other Images

These are any images that need to remain in alignment with the source image.

4.1.4 Estimation Options

Various registration options, which are passed to the Powell optimisation algorithm [67].

Objective Function

Registration involves finding parameters that either maximise or minimise some objective function. For inter-modal registration, use Mutual Information [17, 72], Normalised Mutual Information [68], or Entropy Correlation Coefficient [53].For within modality, you could also use Normalised Cross Correlation.

Separation

The average distance between sampled points (in mm). Can be a vector to allow a coarse registration followed by increasingly fine ones.

Tolerances

The accuracy for each parameter. Iterations stop when differences between successive estimates are less than the required tolerance.

Histogram Smoothing

Gaussian smoothing to apply to the 256x256 joint histogram. Other information theoretic coregistration methods use fewer bins, but Gaussian smoothing seems to be more elegant.

4.2 Coreg: Reslice

Reslice images to match voxel-for-voxel with an image defining some space. The resliced images are named the same as the originals except that they are prefixed by 'r'.

4.2.1 Image Defining Space

This is analogous to the reference image. Images are resliced to match this image (providing they have been coregistered first).

4.2.2 Images to Reslice

These images are resliced to the same dimensions, voxel sizes, orientation etc as the space defining image.

4.2.3 Reslice Options

Various reslicing options.

Interpolation

The method by which the images are sampled when being written in a different space. Nearest Neighbour is fastest, but not normally recommended. It can be useful for re-orienting images while preserving the original intensities (e.g. an image consisting of labels). Bilinear Interpolation is OK for PET, or realigned and re-sliced fMRI. If subject movement (from an fMRI time series) is included in the transformations then it may be better to use a higher degree approach. Note that higher degree B-spline interpolation [69, 70, 71] is slower because it uses more neighbours.

Wrapping

These are typically:
No wrapping - for PET or images that have already been spatially transformed.
Wrap in Y - for (un-resliced) MRI where phase encoding is in the Y direction (voxel space).

Masking

Because of subject motion, different images are likely to have different patterns of zeros from where it was not possible to sample data. With masking enabled, the program searches through the whole time series looking for voxels which need to be sampled from outside the original images. Where this occurs, that voxel is set to zero for the whole set of images (unless the image format can represent NaN, in which case NaNs are used where possible).

Filename Prefix

Specify the string to be prepended to the filenames of the resliced image file(s). Default prefix is 'r'.

4.3 Coreg: Estimate & Reslice

The registration method used here is based on work by Collignon et al [17]. The original interpolation method described in this paper has been changed in order to give a smoother cost function. The images are also smoothed slightly, as is the histogram. This is all in order to make the cost function as smooth as possible, to give faster convergence and less chance of local minima.

At the end of coregistration, the voxel-to-voxel affine transformation matrix is displayed, along with the histograms for the images in the original orientations, and the final orientations. The registered images are displayed at the bottom.

Registration parameters are stored in the headers of the "source" and the "other" images. These images are also resliced to match the source image voxel-for-voxel. The resliced images are named the same as the originals except that they are prefixed by 'r'.

4.3.1 Reference Image

This is the image that is assumed to remain stationary (sometimes known as the target or template image), while the source image is moved to match it.

4.3.2 Images to Reslice

These images are resliced to the same dimensions, voxel sizes, orientation etc as the space defining image.

4.3.3 Other Images

These are any images that need to remain in alignment with the source image.

4.3.4 Estimation Options

Various registration options, which are passed to the Powell optimisation algorithm [67].

Objective Function

Registration involves finding parameters that either maximise or minimise some objective function. For inter-modal registration, use Mutual Information [17, 72], Normalised Mutual Information [68], or Entropy Correlation Coefficient [53].For within modality, you could also use Normalised Cross Correlation.

Separation

The average distance between sampled points (in mm). Can be a vector to allow a coarse registration followed by increasingly fine ones.

Tolerances

The accuracy for each parameter. Iterations stop when differences between successive estimates are less than the required tolerance.

Histogram Smoothing

Gaussian smoothing to apply to the 256x256 joint histogram. Other information theoretic coregistration methods use fewer bins, but Gaussian smoothing seems to be more elegant.

4.3.5 Reslice Options

Various reslicing options.

Interpolation

The method by which the images are sampled when being written in a different space. Nearest Neighbour is fastest, but not normally recommended. It can be useful for re-orienting images while preserving the original intensities (e.g. an image consisting of labels). Bilinear Interpolation is OK for PET, or realigned and re-sliced fMRI. If subject movement (from an fMRI time series) is included in the transformations then it may be better to use a higher degree approach. Note that higher degree B-spline interpolation [69, 70, 71] is slower because it uses more neighbours.

Wrapping

These are typically:
No wrapping - for PET or images that have already been spatially transformed.
Wrap in Y - for (un-resliced) MRI where phase encoding is in the Y direction (voxel space).

Masking

Because of subject motion, different images are likely to have different patterns of zeros from where it was not possible to sample data. With masking enabled, the program searches through the whole time series looking for voxels which need to be sampled from outside the original images. Where this occurs, that voxel is set to zero for the whole set of images (unless the image format can represent NaN, in which case NaNs are used where possible).

Filename Prefix

Specify the string to be prepended to the filenames of the resliced image file(s). Default prefix is 'r'.

Chapter 5

Segment

Contents

5.1 Data 40
5.2 Output Files 40
5.2.1 Grey Matter 43
5.2.2 White Matter 43
5.2.3 Cerebro-Spinal Fluid 43
5.2.4 Bias Corrected 43
5.2.5 Clean up any partitions 43
5.3 Custom 43
5.3.1 Tissue probability maps 43
5.3.2 Gaussians per class 44
5.3.3 Affine Regularisation 44
5.3.4 Warping Regularisation 44
5.3.5 Warp Frequency Cutoff 45
5.3.6 Bias regularisation 45
5.3.7 Bias FWHM 45
5.3.8 Sampling distance 45
5.3.9 Masking image 45

Segment, bias correct and spatially normalise - all in the same model [9]. This function can be used for bias correcting, spatially normalising or segmenting your data. Note that this module needs the images to be roughly aligned with the tissue probability maps before you begin. If strange results are obtained, then this is usually because the images were poorly aligned beforehand. The Display option can be used to manually reposition the images so that the AC is close to coordinate 0,0,0 (within a couple of cm) and the orientation is within a few degrees of the tissue probability map data.

Many investigators use tools within older versions of SPM for a technique that has become known as "optimised" voxel-based morphometry (VBM). VBM performs region-wise volumetric comparisons among populations of subjects. It requires the images to be spatially normalised, segmented into different tissue classes, and smoothed, prior to performing statistical tests [73, 56, 7, 8]. The "optimised" pre-processing strategy involved spatially normalising subjects' brain images to a standard space, by matching grey matter in these images, to a grey matter reference. The historical motivation behind this approach was to reduce the confounding effects of non-brain (e.g. scalp) structural variability on the registration. Tissue classification in older versions of SPM required the images to be registered with tissue probability maps. After registration, these maps represented the prior probability of different tissue classes being found at each location in an

image. Bayes rule can then be used to combine these priors with tissue type probabilities derived from voxel intensities, to provide the posterior probability.

This procedure was inherently circular, because the registration required an initial tissue classification, and the tissue classification requires an initial registration. This circularity is resolved here by combining both components into a single generative model. This model also includes parameters that account for image intensity non-uniformity. Estimating the model parameters (for a maximum a posteriori solution) involves alternating among classification, bias correction and registration steps. This approach provides better results than simple serial applications of each component.

Note that multi-spectral segmentation (e.g. from a registered T1 and T2 image) is not yet implemented, but is planned for a future SPM version.

5.1 Data

Select scans for processing. This assumes that there is one scan for each subject. Note that multi-spectral (when there are two or more registered images of different contrasts) processing is not yet implemented for this method.

5.2 Output Files

This routine produces spatial normalisation parameters (*_seg_sn.mat files) by default. These can be used for writing spatially normalised versions of your data, via the "Normalise: Write" option. This mechanism may produce superior results than the "Normalise: Estimate" option (but probably not as good as those produced using DARTEL).

In addition, it also produces files that can be used for doing inverse normalisation. If you have an image of regions defined in the standard space, then the inverse deformations can be used to warp these regions so that it approximately overlay your image. To use this facility, the bounding-box and voxel sizes should be set to non-finite values (e.g. [NaN NaN NaN] for the voxel sizes, and ones(2,3)*NaN for the bounding box. This would be done by the spatial normalisation module, which allows you to select a set of parameters that describe the nonlinear warps, and the images that they should be applied to.

There are a number of options about what data you would like the routine to produce. The routine can be used for producing images of tissue classes, as well as bias corrected images. The native space option will produce a tissue class image (c*) that is in alignment with the original (see Figure 5.1). You can also produce spatially normalised versions - both with (mwc*) and without (wc*) modulation (see Figure 5.2). The bounding box and voxel sizes of the spatially normalised versions are the same as that of the tissue probability maps with which they are registered. These can be used for doing voxel-based morphometry with (also see the "Using DARTEL' chapter of the manual). All you need to do is smooth them and do the stats (which means no more questions on the mailing list about how to do "optimized VBM").

Modulation is to compensate for the effect of spatial normalisation. When warping a series of images to match a template, it is inevitable that volumetric differences will be introduced into the warped images. For example, if one subject's temporal lobe has half the volume of that of the template, then its volume will be doubled during spatial normalisation. This will also result in a doubling of the voxels labelled grey matter. In order to remove this confound, the spatially normalised grey matter (or other tissue class) is adjusted by multiplying by its relative volume before and after warping. If warping results in a region doubling its volume, then the correction will halve the intensity of the tissue label. This whole procedure has the effect of preserving the total amount of grey matter signal in the normalised partitions.

A deformation field is a vector field, where three values are associated with each location in the field. The field maps from co-ordinates in the normalised image back to co-ordinates in the original image. The value of the field at co-ordinate [x y z] in the normalised space will be the co-ordinate [x' y' z'] in the original volume. The gradient of the deformation field at a co-ordinate is its Jacobian matrix, and it consists of a 3x3 matrix:

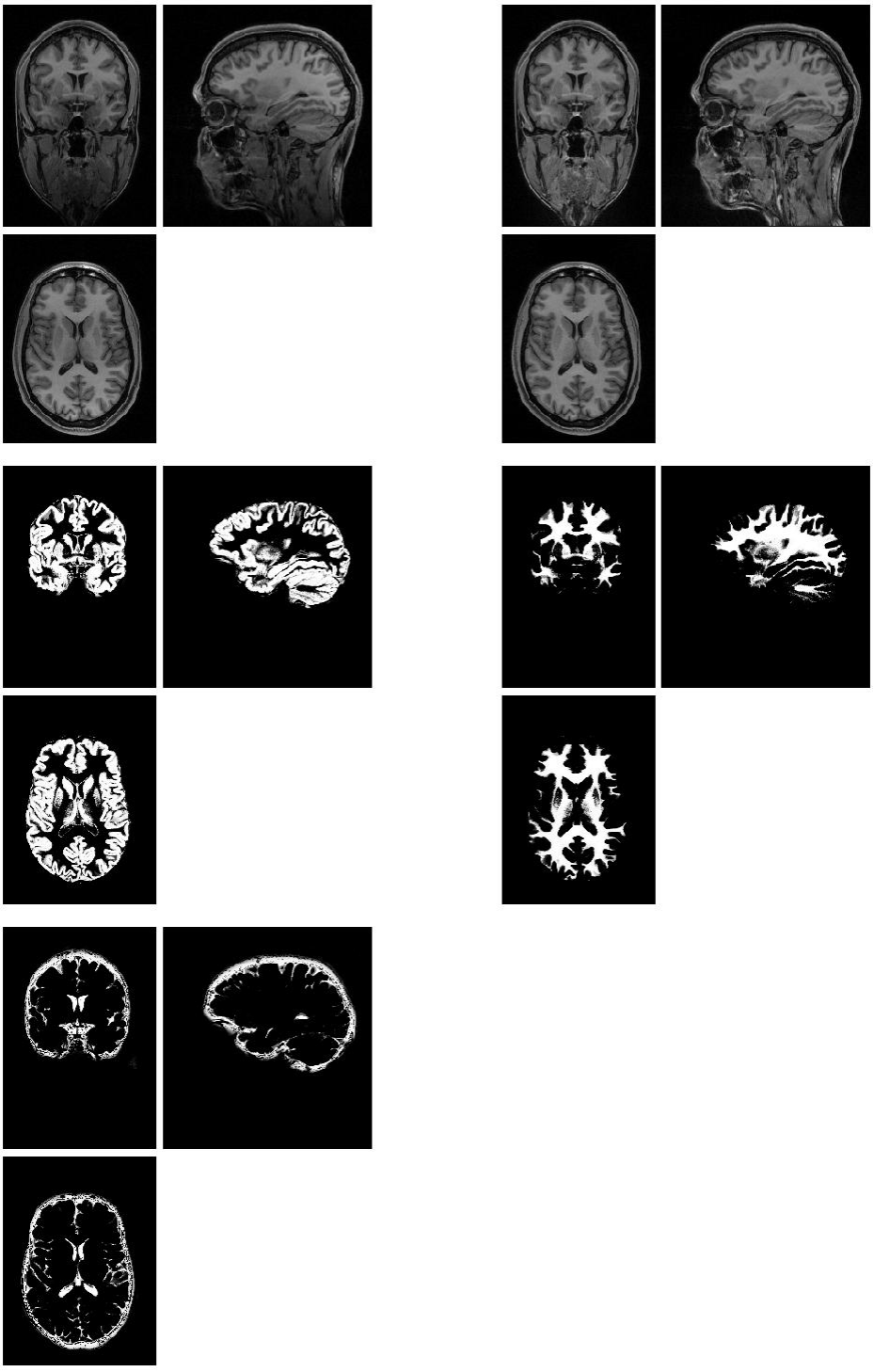


Figure 5.1: Segmentation results. These are the results that can be obtained in the original space of the image (i.e. the results that are not spatially normalised). Top left: original image (X.img). Top right: bias corrected image (mX.img). Middle and bottom rows: segmented grey matter (c1X.img), white matter (c2X.img) and CSF (c3X.img).

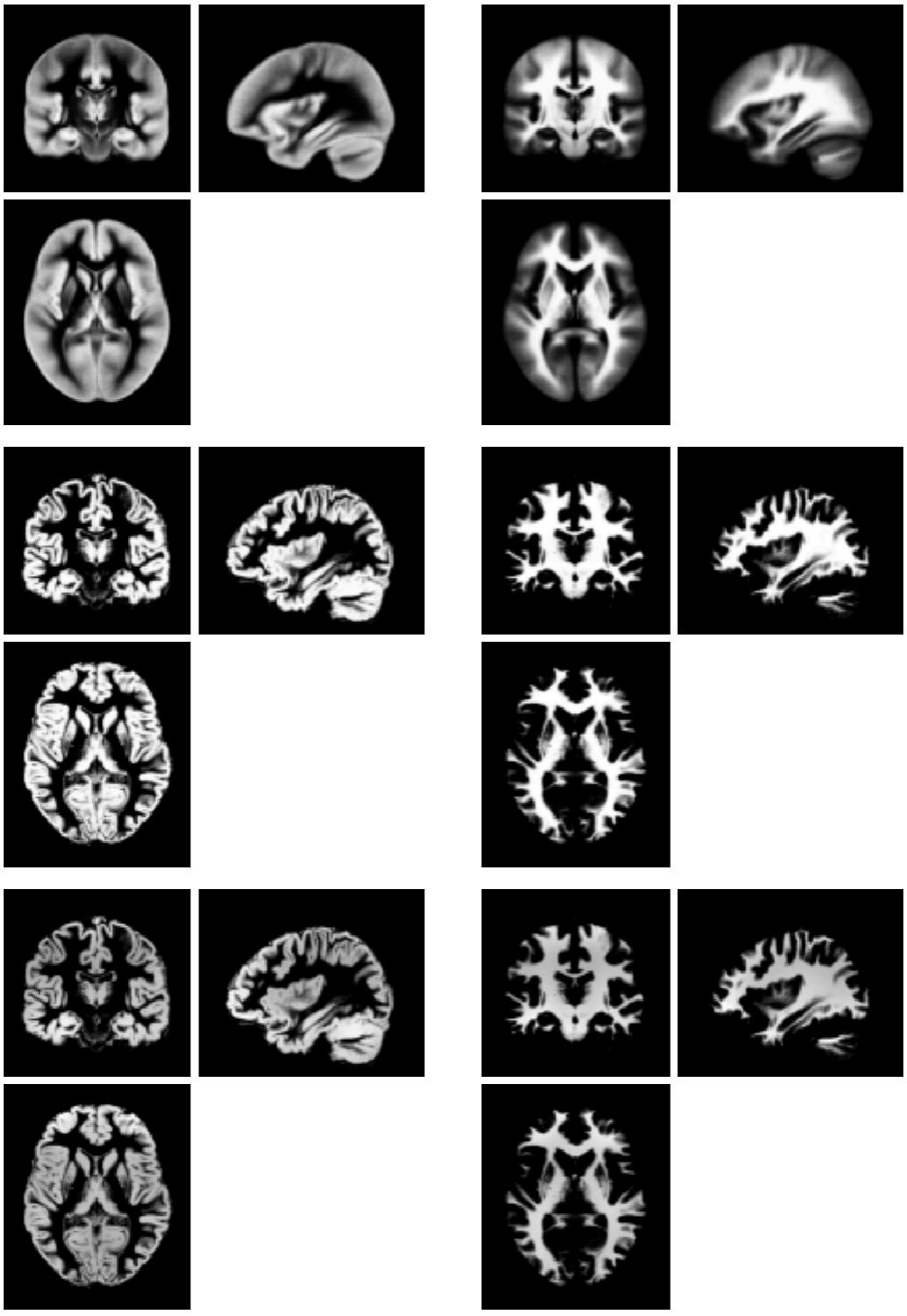


Figure 5.2: Segmentation results. These are the spatially normalised results that can be obtained (note that CSF data is not shown). Top row: The tissue probability maps used to guide the segmentation. Middle row: Spatially normalised tissue maps of grey and white matter (wc1X.img and wc2X.img). Bottom row: Modulated spatially normalised tissue maps of grey and white matter (mwc1X.img and mwc2X.img).

$$\begin{pmatrix} \frac{dx'}{dx} & \frac{dx'}{dy} & \frac{dx'}{dz} \\ \frac{dy'}{dx} & \frac{dy'}{dy} & \frac{dy'}{dz} \\ \frac{dz'}{dx} & \frac{dz'}{dy} & \frac{dz'}{dz} \end{pmatrix}$$

The value of dx'/dy is a measure of how much x' changes if y is changed by a tiny amount. The determinant of the Jacobian is the measure of relative volumes of warped and unwarped structures. The modulation step simply involves multiplying by the relative volumes (see Figure 5.2).

5.2.1 Grey Matter

Options to produce grey matter images: c1*.img, wc1*.img and mwc1*.img.

5.2.2 White Matter

Options to produce white matter images: c2*.img, wc2*.img and mwc2*.img.

5.2.3 Cerebro-Spinal Fluid

Options to produce CSF images: c3*.img, wc3*.img and mwc3*.img.

5.2.4 Bias Corrected

This is the option to produce a bias corrected version of your image. MR images are usually corrupted by a smooth, spatially varying artifact that modulates the intensity of the image (bias). These artifacts, although not usually a problem for visual inspection, can impede automated processing of the images. The bias corrected version should have more uniform intensities within the different types of tissues.

5.2.5 Clean up any partitions

This uses a crude routine for extracting the brain from segmentedimages. It begins by taking the white matter, and eroding it acouple of times to get rid of any odd voxels. The algorithmcontinues on to do conditional dilations for several iterations,where the condition is based upon gray or white matter being present.This identified region is then used to clean up the grey and whitematter partitions, and has a slight influences on the CSF partition.

If you find pieces of brain being chopped out in your data, then you may wish to disable or tone down the cleanup procedure.

5.3 Custom

Various options can be adjusted in order to improve the performance of the algorithm with your data. Knowing what works best should be a matter of empirical exploration. For example, if your data has very little intensity non-uniformity artifact, then the bias regularisation should be increased. This effectively tells the algorithm that there is very little bias in your data, so it does not try to model it.

5.3.1 Tissue probability maps

Select the tissue probability images. These should be maps of grey matter, white matter and cerebro-spinal fluid probability. A nonlinear deformation field is estimated that best overlays the tissue probability maps on the individual subjects' image. The default tissue probability maps are modified versions of the ICBM Tissue Probabilistic Atlases.These tissue probability maps are kindly provided by the International Consortium for Brain Mapping, John C. Mazziotta and Arthur W. Toga. http://www.loni.ucla.edu/ICBM/ICBM_TissueProb.html. The original data

are derived from 452 T1-weighted scans, which were aligned with an atlas space, corrected for scan inhomogeneities, and classified into grey matter, white matter and cerebrospinal fluid. These data were then affine registered to the MNI space and downsampled to 2mm resolution.

Rather than assuming stationary prior probabilities based upon mixing proportions, additional information is used, based on other subjects' brain images. Priors are usually generated by registering a large number of subjects together, assigning voxels to different tissue types and averaging tissue classes over subjects. Three tissue classes are used: grey matter, white matter and cerebro-spinal fluid. A fourth class is also used, which is simply one minus the sum of the first three. These maps give the prior probability of any voxel in a registered image being of any of the tissue classes - irrespective of its intensity.

The model is refined further by allowing the tissue probability maps to be deformed according to a set of estimated parameters. This allows spatial normalisation and segmentation to be combined into the same model. This implementation uses a low-dimensional approach, which parameterises the deformations by a linear combination of about a thousand cosine transform bases. This is not an especially precise way of encoding deformations, but it can model the variability of overall brain shape. Evaluations by Hellier et al have shown that this simple model can achieve a registration accuracy comparable to other fully automated methods with many more parameters.

5.3.2 Gaussians per class

The number of Gaussians used to represent the intensity distribution for each tissue class can be greater than one. In other words, a tissue probability map may be shared by several clusters. The assumption of a single Gaussian distribution for each class does not hold for a number of reasons. In particular, a voxel may not be purely of one tissue type, and instead contain signal from a number of different tissues (partial volume effects). Some partial volume voxels could fall at the interface between different classes, or they may fall in the middle of structures such as the thalamus, which may be considered as being either grey or white matter. Various other image segmentation approaches use additional clusters to model such partial volume effects. These generally assume that a pure tissue class has a Gaussian intensity distribution, whereas intensity distributions for partial volume voxels are broader, falling between the intensities of the pure classes. Unlike these partial volume segmentation approaches, the model adopted here simply assumes that the intensity distribution of each class may not be Gaussian, and assigns belonging probabilities according to these non-Gaussian distributions. Typical numbers of Gaussians could be two for grey matter, two for white matter, two for CSF, and four for everything else.

5.3.3 Affine Regularisation

The procedure is a local optimisation, so it needs reasonable initial starting estimates. Images should be placed in approximate alignment using the Display function of SPM before beginning. A Mutual Information affine registration with the tissue probability maps (D'Agostino et al, 2004) is used to achieve approximate alignment. Note that this step does not include any model for intensity non-uniformity. This means that if the procedure is to be initialised with the affine registration, then the data should not be too corrupted with this artifact.If there is a lot of intensity non-uniformity, then manually position your image in order to achieve closer starting estimates, and turn off the affine registration.

Affine registration into a standard space can be made more robust by regularisation (penalising excessive stretching or shrinking). The best solutions can be obtained by knowing the approximate amount of stretching that is needed (e.g. ICBM templates are slightly bigger than typical brains, so greater zooms are likely to be needed). For example, if registering to an image in ICBM/MNI space, then choose this option. If registering to a template that is close in size, then select the appropriate option for this.

5.3.4 Warping Regularisation

The objective function for registering the tissue probability maps to the image to process, involves minimising the sum of two terms. One term gives a function of how probable the data

is given the warping parameters. The other is a function of how probable the parameters are, and provides a penalty for unlikely deformations. Smoother deformations are deemed to be more probable. The amount of regularisation determines the tradeoff between the terms. Pick a value around one. However, if your normalised images appear distorted, then it may be an idea to increase the amount of regularisation (by an order of magnitude). More regularisation gives smoother deformations, where the smoothness measure is determined by the bending energy of the deformations.

5.3.5 Warp Frequency Cutoff

Cutoff of DCT bases. Only DCT bases of periods longer than the cutoff are used to describe the warps. The number actually used will depend on the cutoff and the field of view of your image. A smaller cutoff frequency will allow more detailed deformations to be modelled, but unfortunately comes at a cost of greatly increasing the amount of memory needed, and the time taken.

5.3.6 Bias regularisation

MR images are usually corrupted by a smooth, spatially varying artifact that modulates the intensity of the image (bias). These artifacts, although not usually a problem for visual inspection, can impede automated processing of the images.

An important issue relates to the distinction between intensity variations that arise because of bias artifact due to the physics of MR scanning, and those that arise due to different tissue properties. The objective is to model the latter by different tissue classes, while modelling the former with a bias field. We know a priori that intensity variations due to MR physics tend to be spatially smooth, whereas those due to different tissue types tend to contain more high frequency information. A more accurate estimate of a bias field can be obtained by including prior knowledge about the distribution of the fields likely to be encountered by the correction algorithm. For example, if it is known that there is little or no intensity non-uniformity, then it would be wise to penalise large values for the intensity non-uniformity parameters. This regularisation can be placed within a Bayesian context, whereby the penalty incurred is the negative logarithm of a prior probability for any particular pattern of non-uniformity.

5.3.7 Bias FWHM

FWHM of Gaussian smoothness of bias. If your intensity non-uniformity is very smooth, then choose a large FWHM. This will prevent the algorithm from trying to model out intensity variation due to different tissue types. The model for intensity non-uniformity is one of i.i.d. Gaussian noise that has been smoothed by some amount, before taking the exponential. Note also that smoother bias fields need fewer parameters to describe them. This means that the algorithm is faster for smoother intensity non-uniformities.

5.3.8 Sampling distance

The approximate distance between sampled points when estimating the model parameters. Smaller values use more of the data, but the procedure is slower.

5.3.9 Masking image

The segmentation can be masked by an image that conforms to the same space as the images to be segmented. If an image is selected, then it must match the image(s) voxel-for voxel, and have the same voxel-to-world mapping. Regions containing a value of zero in this image do not contribute when estimating the various parameters.

Chapter 6

Normalise

Contents

6.1 Normalise: Estimate 48
6.1.1 Data 48
6.1.2 Estimation Options 48
6.2 Normalise: Write 49
6.2.1 Data 49
6.2.2 Writing Options 50
6.3 Normalise: Estimate & Write 50
6.3.1 Data 50
6.3.2 Estimation Options 51
6.3.3 Writing Options 52

This very ancient module spatially (stereotactically) normalises MRI, PET or SPECT images into a standard space defined by some ideal model or template image[s]. The template images supplied with SPM conform to the space defined by the ICBM, NIH P-20 project, and approximate that of the the space described in the atlas of Talairach and Tournoux (1988). The transformation can also be applied to any other image that has been coregistered with these scans. A few researchers may wish to continue using this strategy, but (when good quality anatomical MRI scans are available) the DARTEL approach is now generally recommended instead.

Generally, the algorithms work by minimising the sum of squares difference between the image which is to be normalised, and a linear combination of one or more template images. For the least squares registration to produce an unbiased estimate of the spatial transformation, the image contrast in the templates (or linear combination of templates) should be similar to that of the image from which the spatial normalisation is derived. The registration simply searches for an optimum solution. If the starting estimates are not good, then the optimum it finds may not find the global optimum.

The first step of the normalisation is to determine the optimum 12-parameter affine transformation. Initially, the registration is performed by matching the whole of the head (including the scalp) to the template. Following this, the registration proceeded by only matching the brains together, by appropriate weighting of the template voxels. This is a completely automated procedure (that does not require "scalp editing') that discounts the confounding effects of skull and scalp differences. A Bayesian framework is used, such that the registration searches for the solution that maximises the a posteriori probability of it being correct [10] . i.e., it maximises the product of the likelihood function (derived from the residual squared difference) and the prior function (which is based on the probability of obtaining a particular set of zooms and shears).

The affine registration is followed by estimating nonlinear deformations, whereby the deformations are defined by a linear combination of three dimensional discrete cosine transform (DCT)

basis functions [6] . The default options result in each of the deformation fields being described by 1176parameters, where these represent the coefficients of the deformations in three orthogonal directions. The matching involved simultaneously minimising the membrane energies of the deformation fields and the residual squared difference between the images and template(s).

The primarily use is for stereotactic normalisation to facilitate inter-subject averaging and precise characterisation of functional anatomy [5] . It is not necessary to spatially normalise the data (this is only a pre-requisite for inter-subject averaging or reporting in the Talairach space). If you wish to circumnavigate this step (e.g. if you have single slice data or do not have an appropriate high resolution MRI scan) simply specify where you think the anterior commissure is with the ORIGIN in the header of the first scan (using the 'Display' facility) and proceed directly to 'Smoothing'or 'Statistics'.

All normalised *.img scans are written to the same subdirectory as the original *.img, prefixed with a 'w' (i.e. w*.img). The details of the transformations are displayed in the results window, and the parameters are saved in the "*_sn.mat" file.

6.1 Normalise: Estimate

Computes the warp that best registers a source image (or series of source images) to match a template, saving it to a file imagename'_sn.mat'.

6.1.1 Data

List of subjects. Images of each subject should be warped differently.

Subject

Data for this subject. The same parameters are used within subject.

Source Image The image that is warped to match the template(s). The result is a set of warps, which can be applied to this image, or any other image that is in register with it.

Source Weighting Image Optional weighting images (consisting of pixel values between the range of zero to one) to be used for registering abnormal or lesioned brains. These images should match the dimensions of the image from which the parameters are estimated, and should contain zeros corresponding to regions of abnormal tissue.

6.1.2 Estimation Options

Various settings for estimating warps.

Template Image

Specify a template image to match the source image with. The contrast in the template must be similar to that of the source image in order to achieve a good registration. It is also possible to select more than one template, in which case the registration algorithm will try to find the best linear combination of these images in order to best model the intensities in the source image.

Template Weighting Image

Applies a weighting mask to the template(s) during the parameter estimation. With the default brain mask, weights in and around the brain have values of one whereas those clearly outside the brain are zero. This is an attempt to base the normalisation purely upon the shape of the brain, rather than the shape of the head (since low frequency basis functions can not really cope with variations in skull thickness).

The option is now available for a user specified weighting image. This should have the same dimensions and mat file as the template images, with values in the range of zero to one.

Source Image Smoothing

Smoothing to apply to a copy of the source image. The template and source images should have approximately the same smoothness. Remember that the templates supplied with SPM have been smoothed by 8mm, and that smoothnesses combine by Pythagoras' rule.

Template Image Smoothing

Smoothing to apply to a copy of the template image. The template and source images should have approximately the same smoothness. Remember that the templates supplied with SPM have been smoothed by 8mm, and that smoothnesses combine by Pythagoras' rule.

Affine Regularisation

Affine registration into a standard space can be made more robust by regularisation (penalising excessive stretching or shrinking). The best solutions can be obtained by knowing the approximate amount of stretching that is needed (e.g. ICBM templates are slightly bigger than typical brains, so greater zooms are likely to be needed). If registering to an image in ICBM/MNI space, then choose the first option. If registering to a template that is close in size, then select the second option. If you do not want to regularise, then choose the third.

Nonlinear Frequency Cutoff

Cutoff of DCT bases. Only DCT bases of periods longer than the cutoff are used to describe the warps. The number used will depend on the cutoff and the field of view of the template image(s).

Nonlinear Iterations

Number of iterations of nonlinear warping performed.

Nonlinear Regularisation

The amount of regularisation for the nonlinear part of the spatial normalisation. Pick a value around one. However, if your normalised images appear distorted, then it may be an idea to increase the amount of regularisation (by an order of magnitude) - or even just use an affine normalisation. The regularisation influences the smoothness of the deformation fields.

6.2 Normalise: Write

Allows previously estimated warps (stored in imagename'_sn.mat' files) to be applied to series of images.

6.2.1 Data

List of subjects. Images of each subject should be warped differently.

Subject

Data for this subject. The same parameters are used within subject.

Parameter File Select the '_sn.mat' file containing the spatial normalisation parameters for that subject.

Images to Write These are the images for warping according to the estimated parameters. They can be any images that are in register with the "source" image used to generate the parameters.

6.2.2 Writing Options

Various options for writing normalised images.

Preserve

Preserve Concentrations: Spatially normalised images are not "modulated". The warped images preserve the intensities of the original images.

Preserve Total: Spatially normalised images are "modulated" in order to preserve the total amount of signal in the images. Areas that are expanded during warping are correspondingly reduced in intensity.

Bounding box

The bounding box (in mm) of the volume which is to be written (relative to the anterior commissure).

Voxel sizes

The voxel sizes (x, y & z, in mm) of the written normalised images.

Interpolation

The method by which the images are sampled when being written in a different space.

Nearest Neighbour: - Fastest, but not normally recommended.

Bilinear Interpolation: - OK for PET, or realigned fMRI.

B-spline Interpolation: - Better quality (but slower) interpolation [69], especially with higher degree splines. Do not use B-splines when there is any region of NaN or Inf in the images.

Wrapping

These are typically:

No wrapping: for PET or images that have already been spatially transformed.

Wrap in Y: for (un-resliced) MRI where phase encoding is in the Y direction (voxel space).

Filename Prefix

Specify the string to be prepended to the filenames of the normalised image file(s). Default prefix is 'w'.

6.3 Normalise: Estimate & Write

Computes the warp that best registers a source image (or series of source images) to match a template, saving it to the file imagename'_sn.mat'. This option also allows the contents of the imagename'_sn.mat' files to be applied to a series of images.

6.3.1 Data

List of subjects. Images of each subject should be warped differently.

Subject

Data for this subject. The same parameters are used within subject.

Source Image The image that is warped to match the template(s). The result is a set of warps, which can be applied to this image, or any other image that is in register with it.

Source Weighting Image Optional weighting images (consisting of pixel values between the range of zero to one) to be used for registering abnormal or lesioned brains. These images should match the dimensions of the image from which the parameters are estimated, and should contain zeros corresponding to regions of abnormal tissue.

Images to Write These are the images for warping according to the estimated parameters. They can be any images that are in register with the "source" image used to generate the parameters.

6.3.2 Estimation Options

Various settings for estimating warps.

Template Image

Specify a template image to match the source image with. The contrast in the template must be similar to that of the source image in order to achieve a good registration. It is also possible to select more than one template, in which case the registration algorithm will try to find the best linear combination of these images in order to best model the intensities in the source image.

Template Weighting Image

Applies a weighting mask to the template(s) during the parameter estimation. With the default brain mask, weights in and around the brain have values of one whereas those clearly outside the brain are zero. This is an attempt to base the normalisation purely upon the shape of the brain, rather than the shape of the head (since low frequency basis functions can not really cope with variations in skull thickness).

The option is now available for a user specified weighting image. This should have the same dimensions and mat file as the template images, with values in the range of zero to one.

Source Image Smoothing

Smoothing to apply to a copy of the source image. The template and source images should have approximately the same smoothness. Remember that the templates supplied with SPM have been smoothed by 8mm, and that smoothnesses combine by Pythagoras' rule.

Template Image Smoothing

Smoothing to apply to a copy of the template image. The template and source images should have approximately the same smoothness. Remember that the templates supplied with SPM have been smoothed by 8mm, and that smoothnesses combine by Pythagoras' rule.

Affine Regularisation

Affine registration into a standard space can be made more robust by regularisation (penalising excessive stretching or shrinking). The best solutions can be obtained by knowing the approximate amount of stretching that is needed (e.g. ICBM templates are slightly bigger than typical brains, so greater zooms are likely to be needed). If registering to an image in ICBM/MNI space, then choose the first option. If registering to a template that is close in size, then select the second option. If you do not want to regularise, then choose the third.

Nonlinear Frequency Cutoff

Cutoff of DCT bases. Only DCT bases of periods longer than the cutoff are used to describe the warps. The number used will depend on the cutoff and the field of view of the template image(s).

Nonlinear Iterations

Number of iterations of nonlinear warping performed.

Nonlinear Regularisation

The amount of regularisation for the nonlinear part of the spatial normalisation. Pick a value around one. However, if your normalised images appear distorted, then it may be an idea to increase the amount of regularisation (by an order of magnitude) - or even just use an affine normalisation. The regularisation influences the smoothness of the deformation fields.

6.3.3 Writing Options

Various options for writing normalised images.

Preserve

Preserve Concentrations: Spatially normalised images are not "modulated". The warped images preserve the intensities of the original images.

Preserve Total: Spatially normalised images are "modulated" in order to preserve the total amount of signal in the images. Areas that are expanded during warping are correspondingly reduced in intensity.

Bounding box

The bounding box (in mm) of the volume which is to be written (relative to the anterior commissure).

Voxel sizes

The voxel sizes (x, y & z, in mm) of the written normalised images.

Interpolation

The method by which the images are sampled when being written in a different space.

Nearest Neighbour: - Fastest, but not normally recommended.

Bilinear Interpolation: - OK for PET, or realigned fMRI.

B-spline Interpolation: - Better quality (but slower) interpolation [69], especially with higher degree splines. Do not use B-splines when there is any region of NaN or Inf in the images.

Wrapping

These are typically:

No wrapping: for PET or images that have already been spatially transformed.

Wrap in Y: for (un-resliced) MRI where phase encoding is in the Y direction (voxel space).

Filename Prefix

Specify the string to be prepended to the filenames of the normalised image file(s). Default prefix is 'w'.

Chapter 7

Smooth

Contents

7.1 Images to Smooth 53
7.2 FWHM 53
7.3 Data Type 53
7.4 Filename Prefix 53

This is for smoothing (or convolving) image volumes with a Gaussian kernel of a specified width. It is used as a preprocessing step to suppress noise and effects due to residual differences in functional and gyral anatomy during inter-subject averaging.

7.1 Images to Smooth

Specify the images to smooth. The smoothed images are written to the same subdirectories as the original *.img and are prefixed with a 's' (i.e. s*.img). The prefix can be changed by an option setting.

7.2 FWHM

Specify the full-width at half maximum (FWHM) of the Gaussian smoothing kernel in mm. Three values should be entered, denoting the FWHM in the x, y and z directions.

7.3 Data Type

Data-type of output images. SAME indicates the same datatype as the original images.

7.4 Filename Prefix

Specify the string to be prepended to the filenames of the smoothed image file(s). Default prefix is 's'.

Part III

fMRI Statistics

Chapter 8

fMRI model specification

Statistical analysis of fMRI data uses a mass-univariate approach based on General Linear Models (GLMs). It comprises the following steps (1) specification of the GLM design matrix, fMRI data files and filtering (2) estimation of GLM parameters using classical or Bayesian approaches and (3) interrogation of results using contrast vectors to produce Statistical Parametric Maps (SPMs) or Posterior Probability Maps (PPMs).

The design matrix defines the experimental design and the nature of hypothesis testing to be implemented. The design matrix has one row for each scan and one column for each effect or explanatory variable. (eg. regressor or stimulus function). You can build design matrices with separable session-specific partitions. Each partition may be the same (in which case it is only necessary to specify it once) or different.

Responses can be either event- or epoch related, the only distinction is the duration of the underlying input or stimulus function. Mathematically they are both modeled by convolving a series of delta (stick) or box functions (u), indicating the onset of an event or epoch with a set of basis functions. These basis functions model the hemodynamic convolution, applied by the brain, to the inputs. This convolution can be first-order or a generalized convolution modeled to second order (if you specify the Volterra option). The same inputs are used by the Hemodynamic model or Dynamic Causal Models which model the convolution explicitly in terms of hidden state variables.

Event-related designs may be stochastic or deterministic. Stochastic designs involve one of a number of trial-types occurring with a specified probability at successive intervals in time. These probabilities can be fixed (stationary designs) or time-dependent (modulated or non-stationary designs). The most efficient designs obtain when the probabilities of every trial type are equal. A critical issue in stochastic designs is whether to include null events. If you wish to estimate the evoked response to a specific event type (as opposed to differential responses) then a null event must be included (even if it is not modeled explicitly).

In SPM, analysis of data from multiple subjects typically proceeds in two stages using models at two "levels". The 'first level' models are used to implement a within-subject analysis. Typically there will be as many first level models as there are subjects. Analysis proceeds as described using the "Specify first level" and "Estimate" options. The results of these analyses can then be presented as "case studies". More often, however, one wishes to make inferences about the population from which the subjects were drawn. This is an example of a "Random-Effects (RFX) analysis" (or, more properly, a mixed-effects analysis). In SPM, RFX analysis is implemented using the "summary-statistic" approach where contrast images from each subject are used as summary measures of subject responses. These are then entered as data into a "second level" model.

Figure 8.1 shows how the SPM graphics window appears during fMRI model specification.

8.1 Timing parameters

Specify various timing parameters needed to construct the design matrix. This includes the units of the design specification and the interscan interval.

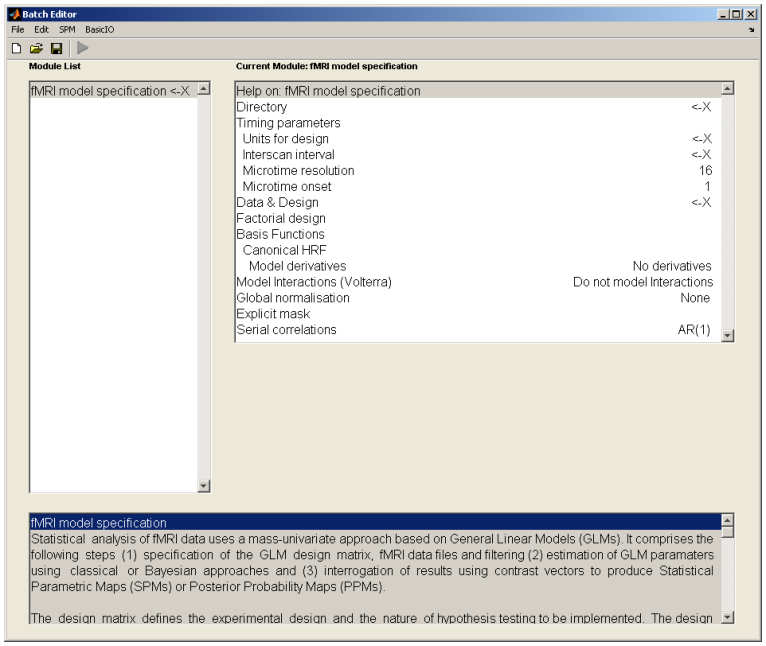


Figure 8.1: *After starting SPM in fMRI mode and pressing the "Specify 1st-level" button, the SPM batch editor window should appear as above. The options for "fMRI model specification" can be examined by clicking on them. A single click will bring up some help text in the lower subwindow (not shown in the above graphic). Options highlighted with a "<-X" are mandatory and must be filled in by the user. Each of the options shown above is described in this chapter.*

Also, with long TRs you may want to shift the regressors so that they are aligned to a particular slice. This is effected by changing the microtime resolution and onset.

8.1.1 Units for design

The onsets of events or blocks can be specified in either scans or seconds.

8.1.2 Interscan interval

Interscan interval, TR, (specified in seconds). This is the time between acquiring a plane of one volume and the same plane in the next volume. It is assumed to be constant throughout.

8.1.3 Microtime resolution

In Echo-Planar Imaging (EPI), data is acquired a plane at a time. To acquire a whole volume of data takes at least a second or two.

It is possible, however, that experimental events may occur between scan (volume) acquisition times. This can be specified when building your design matrix either by (i) specifying your design in scans and using non-integer values or (ii) specifying your design in seconds at a resolution greater than the TR.

SPM takes these timing specifications and builds its regressors using a 'microtime' time-scale. The microtime resolution, t, is the number of time-bins per scan.

Do not change this parameter unless you have a long TR and wish to shift regressors so that they are aligned to a particular slice.

8.1.4 Microtime onset

The microtime onset, t0, is the first time-bin at which the regressors are resampled to coincide with data acquisition. If t0 = 1 then the regressors will be appropriate for the first slice. If you

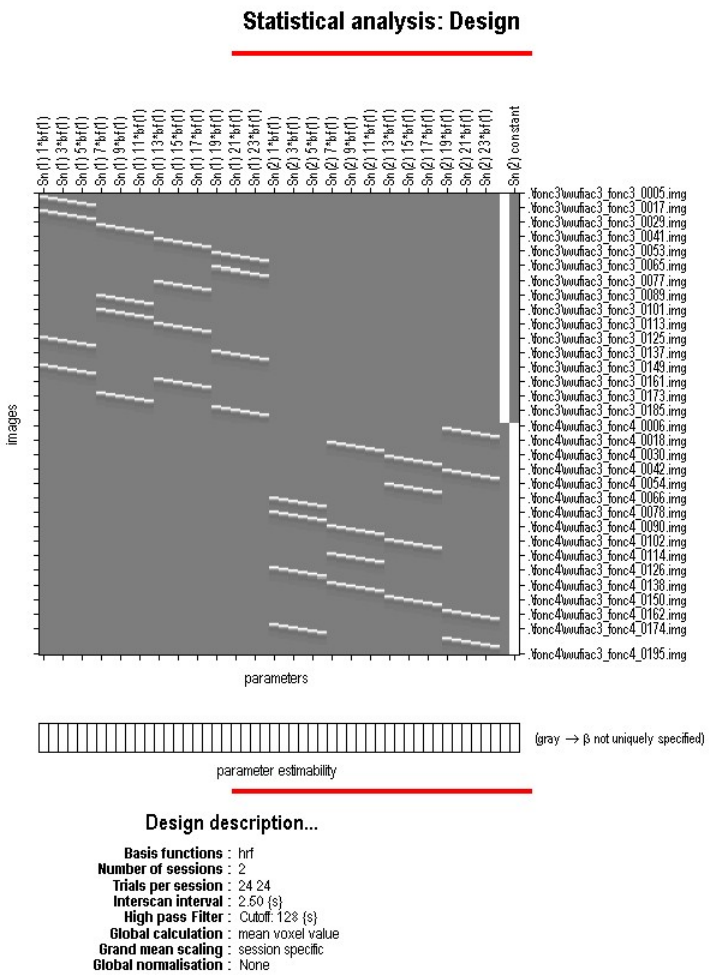


Figure 8.2: *Design matrix for fMRI data from two sessions. There are 24 experimental conditions for each session. The last two columns model the average activity in each session, giving a total of 50 regressors. There are 191 fMRI scans for each session. The overall design matrix therefore has 382 rows and 50 columns.*

want to temporally realign the regressors so that they match responses in the middle slice then make t0 = t/2 (assuming there is a negligible gap between volume acquisitions).

Do not change the default setting unless you have a long TR.

A typical use of the t and t0 parameters is to set them to correspond to the results of any slice timing correction you have made eg. if you have 24 slices and have made slice 12 the reference slice you would set t=24, t0=12.

8.2 Data & Design

The design matrix defines the experimental design and the nature of hypothesis testing to be implemented. The design matrix has one row for each scan and one column for each effect or explanatory variable. (e.g. regressor or stimulus function). Figure 8.2 shows an example of a design matrix.

You can build design matrices with separable session-specific partitions. Each partition may be the same (in which case it is only necessary to specify it once) or different. Responses can be either event- or epoch related, where the latter model involves prolonged and possibly time-varying responses to state-related changes in experimental conditions. Event-related response are

modelled in terms of responses to instantaneous events. Mathematically they are both modelled by convolving a series of delta (stick) or box-car functions, encoding the input or stimulus function. with a set of hemodynamic basis functions.

8.2.1 Subject/Session

The design matrix for fMRI data consists of one or more separable, session-specific partitions. These partitions are usually either one per subject, or one per fMRI scanning session for that subject.

Scans

Select the fMRI scans for this session. They must all have the same image dimensions, orientation, voxel size etc. This is implemented using SPM's file selector.

Conditions

You are allowed to combine both event- and epoch-related responses in the same model and/or regressor. Any number of condition (event or epoch) types can be specified. Epoch and event-related responses are modeled in exactly the same way by specifying their onsets [in terms of onset times] and their durations. Events are specified with a duration of 0. If you enter a single number for the durations it will be assumed that all trials conform to this duration.For factorial designs, one can later associate these experimental conditions with the appropriate levels of experimental factors.

Condition An array of input functions is constructed, specifying occurrence events or epochs (or both). These are convolved with a basis set at a later stage to give regressors that enter into the design matrix. Interactions of evoked responses with some parameter (time or a specified variate) enter at this stage as additional columns in the design matrix with each trial multiplied by the [expansion of the] trial-specific parameter. The 0th order expansion is simply the main effect in the first column.

Name Condition Name

Onsets Specify a vector of onset times for this condition type. This can be entered using the keyboard eg. typing in "100 300" and then hitting return or "100;300" or "[100,300]" or "[100,300]".

More usually, however, this specification takes place using variables that have been created before and loaded into matlab. For example, an `my_onsets` cell array[1] might exist in a file you created earlier called `my_design.mat`. You would then type `load my_design` at the matlab command prompt before pressing the 'Specify 1st-level' button.

You could then specify the onsets for condition 2 by typing in eg. `my_onsets{2}` instead of entering the numbers via the keyboard.

Durations Specify the event durations (in seconds). Epoch and event-related responses are modeled in exactly the same way but by specifying their different durations. Events are specified with a duration of 0. If you enter a single number for the durations it will be assumed that all trials conform to this duration. If you have multiple different durations, then the number must match the number of onset times.

Time Modulation This option allows for the characterisation of nonstationary responses. Specifically, you can model either linear or nonlinear time effects. For example, 1st order modulation would model the stick functions and a linear change of the stick function heights over time. Higher order modulation will introduce further columns that contain the stick functions scaled by time squared, time cubed etc.

[1] Cell arrays are usually used in preference to matrices as different event types can then have different numbers of events.

Parametric Modulations The stick function itself can be modulated by some parametric variate (this can be time or some trial-specific variate like reaction time) modeling the interaction between the trial and the variate. The events can be modulated by zero or more parameters.

See [15, 14] for further details of parametric modulations.

Multiple conditions

If you have multiple conditions then entering the details a condition at a time is very inefficient. This option can be used to load all the required information in one go.

You will need to create a `*.mat` file containing the relevant information. This `*.mat` file must include the following cell arrays: names, onsets and durations eg. `names{2}='SSent-DSpeak'`, `onsets{2}=[3 5 19 222]`, `durations{2}=[0 0 0 0]` contain the required details of the second condition. These cell arrays may be made available by your stimulus delivery program eg. COGENT. The duration vectors can contain a single entry if the durations are identical for all events.

You then need to use SPM's file selector to select this `*.mat` file.

Regressors

Regressors are additional columns included in the design matrix, which may model effects that would not be convolved with the haemodynamic response. One such example would be the estimated movement parameters, which may confound the data.

Regressor

Name Enter name of regressor eg. First movement parameter

Value Enter the values that the regressor takes. This could also be, for example, the name of a variable in MATLAB's work space that you have previously loaded in from a file. This might be a subjects movement parameters or reaction times.

Multiple regressors

If you have mutliple regressors eg. realignment parameters, then entering the details a regressor at a time is very inefficient. This option can be used to load all the required information in one go.

You will first need to create a `*.mat` file containing a matrix R. Each column of R will contain a different regressor. When SPM creates the design matrix the regressors will be named R1, R2, R3, ..etc.

You then need to use SPM's file selector to select this `*.mat` file.

High-pass filter

The default high-pass filter cutoff is 128 seconds. Slow signal drifts with a period longer than this will be removed. Use "Explore design" to ensure this cut-off is not removing too much experimental variance. This is described later in section 8.10. High-pass filtering is implemented using a residual forming matrix (i.e. it is not a convolution) and is simply a way to remove confounds without estimating their parameters explicitly. The constant term is also incorporated into this filter matrix.

8.3 Factorial design

If you have a factorial design then SPM can automatically generate the contrasts necessary to test for the main effects and interactions.

This includes the F-contrasts necessary to test for these effects at the within-subject level (first level) and the simple contrasts necessary to generate the contrast images for a between-subject (second-level) analysis.

To use this option, create as many factors as you need and provide a name and number of levels for each. SPM assumes that the condition numbers of the first factor change slowest, the second factor next slowest etc. It is best to write down the contingency table for your design to ensure this condition is met. This table relates the levels of each factor to the conditions.

For example, if you have 2-by-3 design your contingency table has two rows and three columns where the the first factor spans the rows, and the second factor the columns. The numbers of the conditions are 1,2,3 for the first row and 4,5,6 for the second.

See [38] for more information on SPM and factorial designs.

8.3.1 Factor

Add a new factor to your experimental design.

Name

Name of factor, eg. 'Repetition'

Levels

Enter number of levels for this factor, eg. 2

8.4 Basis Functions

SPM uses basis functions to model the hemodynamic response. This could be a single basis function or a set of functions. The most common choice is the 'Canonical HRF' with or without time and dispersion derivatives.

8.4.1 Canonical HRF

Canonical Hemodynamic Response Function (HRF). This is the default option. Contrasts of these effects have a physical interpretation and represent a parsimonious way of characterising event-related responses. This option is also useful if you wish to look separately at activations and deactivations. This is implemented using a t-contrast with a +1 or -1 entry over the canonical regressor.

Model derivatives

Model HRF Derivatives. The canonical HRF combined with time and dispersion derivatives comprise an 'informed' basis set, as the shape of the canonical response conforms to the hemodynamic response that is commonly observed. The incorporation of the derivative terms allow for variations in subject-to-subject and voxel-to-voxel responses. The time derivative allows the peak response to vary by plus or minus a second and the dispersion derivative allows the width of the response to vary by a similar amount.

A positive estimate of the time-derivative regression coefficient implies that the peak hemodynamic response occurs earlier than usual ie. than would be expected using just the canonical regressor. A positive estimate for the dispersion derivative implies a less dispersed response than usual.

The informed basis set requires an SPMF for inference. T-contrasts over just the canonical are perfectly valid but assume constant delay/dispersion. The informed basis set compares favourably with eg. FIR bases on many data sets [40].

8.4.2 Other basis sets

The other basis sets supported by SPM are

1. Fourier Set
2. Fourier Set (Hanning)

3. Gamma Functions

4. Finite Impulse Response (FIR)

For each of these options you must also specify the **window length** which is the length in seconds of the post-stimulus time window that the basis functions span. You must also specify the **order**, that is, how many basis functions to use.

Usually, an informed basis set should be sufficient for most data sets. If this does not provide a good fit to the data it may be worthwhile re-considering how the neuronal events are modelled ie. is the timing correct ? should events be split into subsets ?

Alternatively, the gamma basis functions are an interesting choice as a particular linear combination of them is actually used to specify the canonical HRF. The FIR approach is of interest as it is equivalent to the method of 'selective averaging'. See [37] for further details.

8.5 Model Interactions (Volterra)

Generalized convolution of inputs, U, with basis set, bf.

For first order expansions the causes are simply convolved (e.g. stick functions) in U by the basis functions in bf to create a design matrix X. For second order expansions new entries appear that correspond to the interaction among the original causes. The basis functions for these effects are two dimensional and are used to assemble the second order kernel.

Interactions or response modulations can enter at two levels. Firstly the stick function itself can be modulated by some parametric variate. This can be time or some trial-specific variate like reaction time modeling the interaction between the trial and the variate. Secondly interactions among the trials themselves can be modeled using a Volterra series formulation that accommodates interactions over time (and therefore within and between trial types).

This last option is useful for accommodating nonlinearities in the hemodynamic response. For example, if two events occur within a second or so of each other then the hemodynamic response to the pair may be less than the sum of the responses to each event when occuring in isolation. This type of 'sub-linear' response can be modelled using Volterra kernels. See [29] for further details.

8.6 Directory

Select a directory where the SPM.mat file containing the specified design matrix will be written. If this directory already contains an SPM.mat file then SPM will warn you of this before overwriting it, when the specification job is run.

8.7 Global normalisation

SPM can normalise fMRI data in one of two ways. These are selected using the options 'None' (the default) and 'Scaling'.

Both methods are based on first estimating the average within-brain fMRI signal, g_{ns}, where n denotes scan and s denotes session. If you select 'Scaling', SPM will multiply each fMRI value in scan n and session s by $100/g_{ns}$.

If you select "None" then SPM computes the grand mean value, $g_s = \frac{\sum_{n=1}^{N} g_{ns}}{N}$ where N is the number of scans in that session. This is the fMRI signal averaged over all voxels within the brain and all time points within session s. SPM then implements "Session-specific grand mean scaling" by multiplying each fMRI data point in session s by $100/g_s$.

See [1] for further discussion of this issue.

8.8 Explicit mask

Specify an image for explicitly masking the analysis. A sensible option here is to use a segmentation of structural images to specify a within-brain mask. If you select that image as an explicit

mask then only those voxels in the brain will be analysed. This both speeds the estimation and restricts SPMs/PPMs to within-brain voxels. Alternatively, if such structural images are unavailable or no masking is required, then leave this field empty.

8.9 Serial correlations

Serial correlations in fMRI time series due to aliased biorhythms and unmodelled neuronal activity can be accounted for using an autoregressive AR(1) model during Classical (ReML) parameter estimation.

This estimate assumes the same correlation structure for each voxel, within each session. ReML estimates are then used to correct for non-sphericity during inference by adjusting the statistics and degrees of freedom appropriately. The discrepancy between estimated and actual correlations are greatest at low frequencies. Therefore specification of the high-pass filter is particularly important.

Serial correlation can be ignored if you choose the "none" option. Note that the above options only apply if you later specify that your model will be estimated using the Classical (ReML) approach. If you choose Bayesian estimation these options will be ignored. For Bayesian estimation, the choice of noise model (AR model order) is made under the estimation options. See [32, 63] for further discussion of these issues.

8.10 Reviewing your design

After you have completed the SPM "job" file for specifying your fMRI design, and have run it, you will then be able to review your design by pressing the "Review" button in SPM's button window (the top-left window). This is particularly useful, for example, for checking that your experimental variance has not been removed by high-pass filtering, as shown in Figure 8.3.

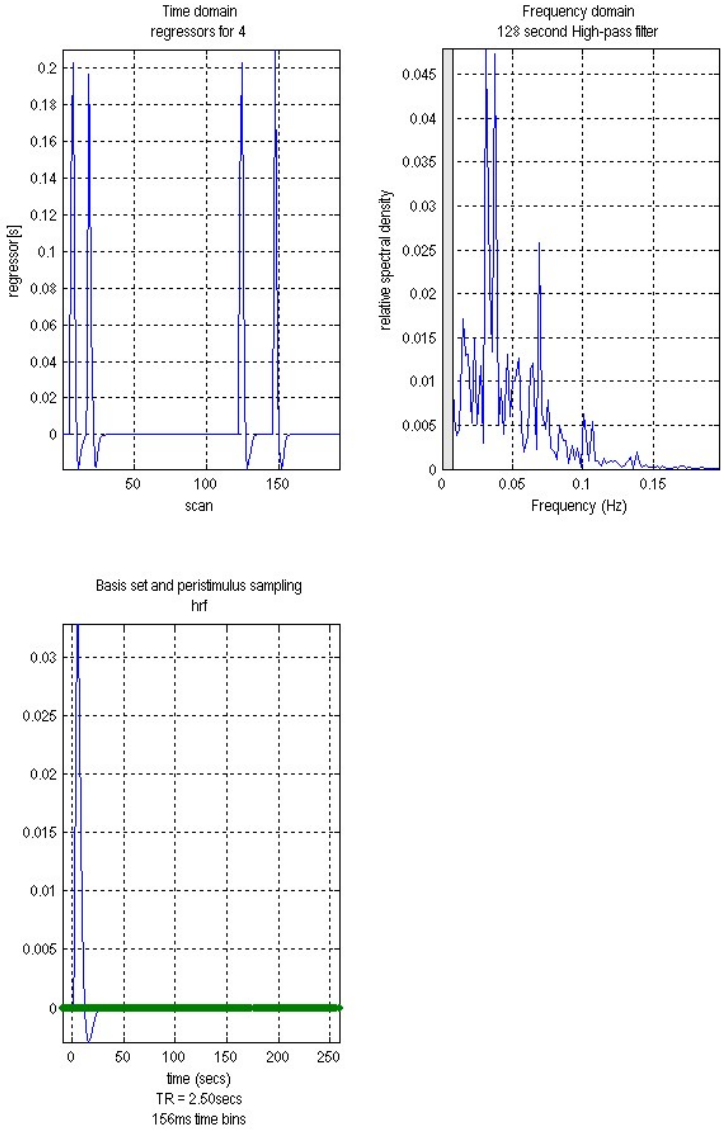


Figure 8.3: *After pressing "Review", selecting the pull-down 'Design' menu, Explore->Session, and selecting the regressor you wish to look at, you should get a plot similar to the one above. The top row shows time and frequency domain plots of the time-series corresponding to this regressor. In this particular case we have four events. Each event or "stick function" has been convolved with the hemodynamic response function shown in the bottom panel. The frequency domain graph is useful for checking that experimental variance is not removed by high-pass filtering. The grayed out section of the frequency plot shows those frequencies which are removed. For this regressor we have plenty of remaining experimental variance (see the peak at about 0.04Hz).*

Chapter 9

fMRI model estimation

Model parameters can be estimated using classical (ReML - Restricted Maximum Likelihood) or Bayesian algorithms. After parameter estimation, the RESULTS button can be used to specify contrasts that will produce Statistical Parametric Maps (SPMs), Effect Size Maps (ESMs) or Posterior Probability Maps (PPMs) and tables of statistics.

9.1 Select SPM.mat

Select the `SPM.mat` file that contains the design specification. SPM will output the results of its analysis into this directory. This includes overwriting the `SPM.mat` file. When the estimation job is run, no warning will be given that the SPM.mat file will be overwritten. A warning is given at the specification stage. When it comes to estimation, SPM assumes that you've now sorted out your directory structures.

9.2 Method

There are three possible estimation procedures for fMRI models (1) classical (ReML) estimation of first or second level models, (2) Bayesian estimation of first level models and (3) Bayesian estimation of second level models. Option (2) uses a Variational Bayes (VB) algorithm introduced in SPM5. Option (3) uses the Empirical Bayes algorithm with global shrinkage priors that was also in SPM2.

To use option (3) you must have already estimated the model using option (1). That is, for second-level models you must run a ReML estimation before running a Bayesian estimation. This is not necessary for option (2). Bayesian estimation of 1st-level models using VB does not require a prior ReML estimation.

9.2.1 Classical

Model parameters are estimated using Restricted Maximum Likelihood (ReML). This assumes the error correlation structure is the same at each voxel. This correlation can be specified using either an AR(1) or an Independent and Identically Distributed (IID) error model. These options are chosen at the model specification stage. ReML estimation should be applied to spatially smoothed functional images. See [32, 25] for further details of the ReML estimation scheme. After estimation, specific profiles of parameters are tested using a linear compound or contrast with the T or F statistic. The resulting statistical map constitutes an SPM. The SPMT/F is then characterised in terms of focal or regional differences by assuming that (under the null hypothesis) the components of the SPM (ie. residual fields) behave as smooth stationary Gaussian fields.

The rest of this chapter describes the Bayesian estimation options. So, please skip to the next chapter if you are interested only in classical estimation and inference.

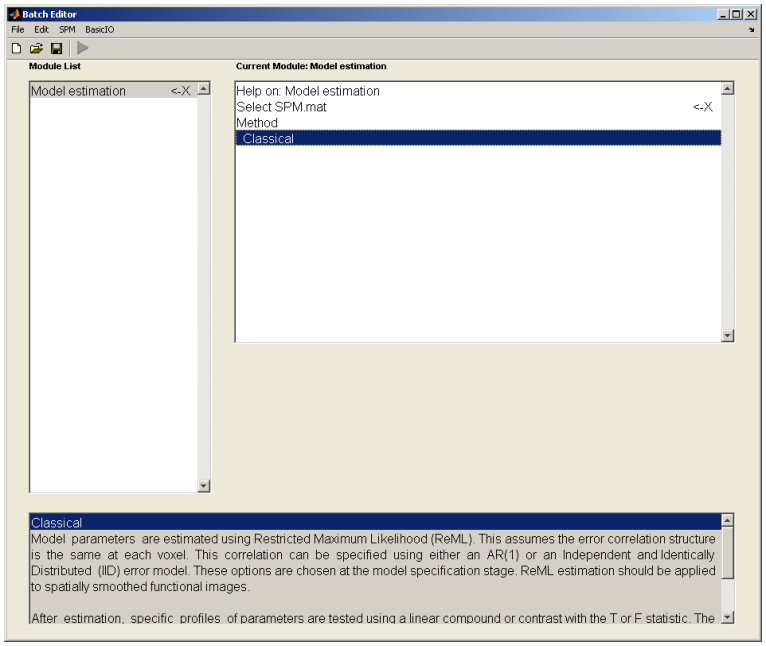


Figure 9.1: *After starting SPM in fMRI mode and pressing the "Estimate" button, the SPM batch editor window should appear as above. The options for "fMRI model estimation" can be examined by clicking on them. A single click will bring up some help text in the lower subwindow (not shown in the above graphic). Options highlighted with a '<-X' are mandatory and must be filled in by the user. Each of the options shown above is described in this chapter.*

9.2.2 Bayesian 1st-level

Model parameters are estimated using Variational Bayes (VB). This allows you to specify spatial priors for regression coefficients and regularised voxel-wise AR(P) models for fMRI noise processes. The algorithm does not require functional images to be spatially smoothed. Estimation will take about 5 times longer than with the classical approach. This is why VB is not the default estimation option. The VB approach has been described in a number of papers [63, 65, 60, 61].

After estimation, contrasts are used to find regions with effects larger than a user-specified size eg. 1 per cent of the global mean signal. These effects are assessed statistically using a Posterior Probability Map (PPM) [30].

Analysis Space

Because estimation can be time consuming options are provided to analyse selected slices or clusters rather than the whole volume.

Volume A volume of data is analysed in "blocks", which can be a slice or 3D subvolume, where the extent of each subvolume is determined using a graph partitioning algorithm. Enter the block type, i.e. "Slices" or "Subvolumes".

Block type Enter the block type, i.e. "Slices" or "Subvolumes".

Slices Enter Slice Numbers. This can be a single slice or multiple slices. If you select a single slice or only a few slices you must be aware of the interpolation options when, after estimation, displaying the estimated images eg. images of contrasts or AR maps. The default interpolation option may need to be changed to nearest neighbour (NN) (see bottom right hand of graphics window) for you slice maps to be visible.

Slice numbers Enter Slice Numbers.

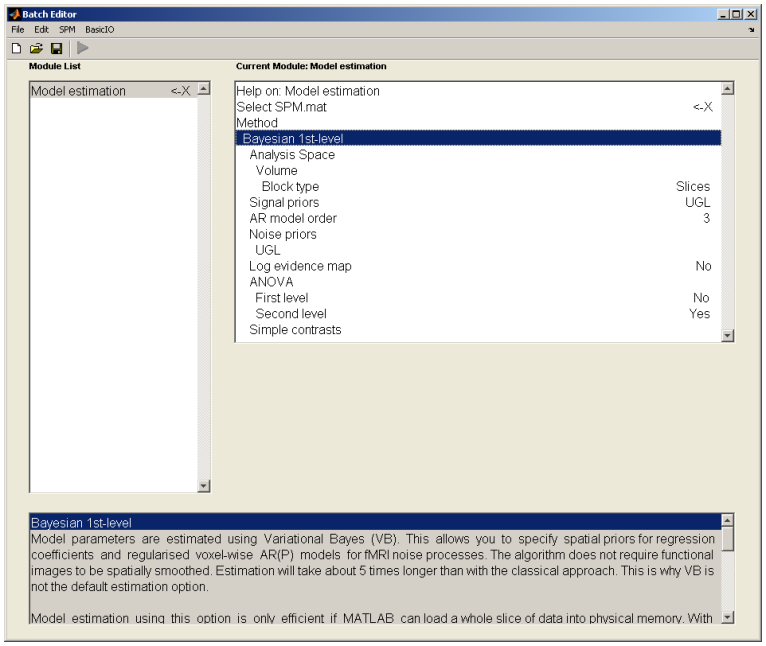


Figure 9.2: *After choosing "Bayesian 1st-level" under "Method", the SPM batch editor window should appear as above. Each of the options shown above is described in this chapter.*

Block type Enter the block type, i.e. "Slices" or "Subvolume".

Clusters Because estimation can be time consuming an option is provided to analyse selected clusters rather than the whole volume.

Cluster mask Select cluster image.

Block type Enter the block type, i.e. "Slices" or "Subvolumes".

Signal priors

- **[UGL] Unweighted Graph Laplacian.** This spatial prior is the recommended option. Regression coefficients at a given voxel are (softly) constrained to be similar to those at nearby voxels. The strength of this constraint is determined by a spatial precision parameter that is estimated from the data. Different regression coefficients have different spatial precisions allowing each putative experimental effect to have its own spatial regularity.
- **[GMRF] Gaussian Markov Random Field.** This is equivalent to a normalized UGL.
- **[LORETA] Low resolution Tomography Prior.** This is equivalent to UGL squared. It is a standatd choice for EEG source localisation algorithms.
- **[WGL] Weighted Graph Laplacian.** This is a generalization of the UGL, where weights can be used to preserve "edges" of functional responses.
- **[Global] Global Shrinkage prior.** This is not a spatial prior in the sense that regression coefficients are constrained to be similar to neighboring voxels. Instead, the average effect over all voxels (global effect) is assumed to be zero and all regression coefficients are shrunk towards this value in proporation to the prior precision. This is the same prior that is used for Bayesian estimation at the second level models, except that here the prior precision is estimated separaetly for each slice.

- **[Uninformative] A flat prior.** Essentially, no prior information is used. If you select this option then VB reduces to Maximum Likelihood (ML)estimation. This option is useful if, for example, you do not wish to use a spatial prior but wish to take advantage of the voxel-wise AR(P) modelling of noise processes. In this case, you would apply the algorithm to images that have been spatially smoothed. For P=0, ML estimation in turn reduces to Ordinary Least Squares (OLS) estimates, and for P>0 ML estimation is equivalent to a weighted least squares (WLS) but where the weights are different at each voxel (reflecting the different noise correlation at each voxel).

AR model order

An AR model order of 3 is the default. Cardiac and respiratory artifacts are periodic in nature and therefore require an AR order of at least 2. In previous work, voxel-wise selection of the optimal model order showed that a value of 3 was the highest order required.

Higher model orders have little effect on the estimation time. If you select a model order of zero this corresponds to the assumption that the errors are IID. This AR specification overrides any choices that were made in the model specification stage.

Voxel-wise AR models are fitted separately for each session of data. For each session this therefore produces maps of AR(1), AR(2) etc coefficients in the output directory.

Noise priors

There are five noise prior options here (1) UGL, (2) GMRF, (3) LORETA, (4) Tissue-type and (5) Robust.

UGL [UGL] Unweighted graph-Laplacian. This is the default option. This spatial prior is the same as that used for the regression coefficients. Spatial precisions are estimated separately for each AR coefficient eg. the AR(1) coefficient over space, AR(2) over space etc.

GMRF [GMRF] Gaussian Markov Random Field. See comments on GMRF priors for regresion coefficients.

LORETA [LORETA] Low resolution Tomography Prior. See comments on LORETA priors for regresion coefficients.

Tissue-type [Tissue-type] AR estimates at each voxel are biased towards typical values for that tissue type (eg. gray, white, CSF). If you select this option you will need to then select files that contain tissue type maps (see below). These are typically chosen to be Grey Matter, White Matter and CSF images derived from segmentation of registered structural scans.

Previous work has shown that there is significant variation in AR values with tissue type. However, GMRF priors have previously been favoured by Bayesian model comparison.

Robust Robust GLM. Uses Mixture of Gaussians noise model.

Log evidence map

Computes the log evidence for each voxel

ANOVA

Perform 1st or 2nd level Analysis of Variance.

First level This is implemented using Bayesian model comparison. For example, to test for the main effect of a factor two models are compared, one where the levels are represented using different regressors and one using the same regressor. This therefore requires explicit fitting of several models at each voxel and is computationally demanding (requiring several hours of computation). The recommended option is therefore NO.

To use this option you must have already specified your factorial design during the model specification stage.

Second level This option tells SPM to automatically generate the simple contrasts that are necessary to produce the contrast images for a second-level (between-subject) ANOVA. Naturally, these contrasts can also be used to characterise simple effects for each subject.

With the Bayesian estimation option it is recommended that contrasts are computed during the parameter estimation stage (see 'simple contrasts' below). The recommended option here is therefore YES.

To use this option you must have already specified your factorial design during the model specification stage.

If you wish to use these contrast images for a second-level analysis then you will need to spatially smooth them to take into account between-subject differences in functional anatomy ie. the fact that one persons V5 may be in a different position than anothers.

Simple contrasts

"Simple" contrasts refers to a contrast that spans one-dimension ie. to assess an effect that is increasing or decreasing.

If you have a factoral design then the contrasts needed to generate the contrast images for a 2nd-level ANOVA (or to assess these simple effects within-subject) can be specified automatically using the ANOVA->Second level option.

When using the Bayesian estimation option it is computationally more efficient to compute the contrasts when the parameters are estimated. This is because estimated parameter vectors have potentially different posterior covariance matrices at different voxels and these matrices are not stored. If you compute contrasts post-hoc these matrices must be recomputed (an approximate reconstruction based on a Taylor series expansion is used). It is therefore recommended to specify as many contrasts as possible prior to parameter estimation.

If you wish to use these contrast images for a second-level analysis then you will need to spatially smooth them to take into account between-subject differences in functional anatomy ie. the fact that one persons V5 may be in a different position than anothers.

Simple contrast

Name Name of contrast eg. "Positive Effect".

Contrast vector These contrasts are used to generate PPMs which characterise effect sizes at each voxel. This is in contrast to SPMs in which eg. maps of t-statistics show the ratio of the effect size to effect variability (standard deviation). SPMs are therefore a-dimensional. This is not the case for PPMs as the size of the effect is of primary interest. Some care is therefore needed about the scaling of contrast vectors. For example, if you are interested in the differential effect size averaged over conditions then the contrast 0.5 0.5 -0.5 -0.5 would be more suitable than the 1 1 -1 -1 contrast which looks at the differential effect size summed over conditions.

9.2.3 Bayesian 2nd-level

Bayesian estimation of 2nd level models. This option uses the Empirical Bayes algorithm with global shrinkage priors that was previously implemented in SPM2. Use of the global shrinkage prior embodies a prior belief that, on average over all voxels, there is no net experimental effect. Some voxels will respond negatively and some positively with a variability determined by the prior precision. This prior precision can be estimated from the data using Empirical Bayes.

9.3 Output files

After estimation a number of files are written to the output directory. These are

- An `SPM.mat` file containing specification of the design and estimated model parameters

9.3.1 Classical 1st-level

For classical 1st-level models the following files are also produced

- Images of estimated regression coefficients `beta_000k.img` where k indexes the kth regression coefficient.
- An image of the variance of the error `ResMS.img`.
- An image `mask.img` indicating which voxels were included in the analysis.
- The image `RPV.img`, the estimated resels per voxel.
- If contrasts have been specified SPM also writes `con_000i.img` if the ith contrast is a t-contrast and the extra sum of squares image `ess_000i.img` if it is an F-contrast.

Type `help spm_spm` at the matlab command prompt for further information.

9.3.2 Bayesian 1st-level

For Bayesian 1st-level models the following files are also produced

- Images of estimated regression coefficients `Cbeta_000k.img` where k indexes the kth regression coefficient. These filenames are prefixed with a "C" indicating that these are the mean values of the 'Conditional' or 'Posterior' density.
- Images of error bars/standard deviations on the regression coefficients `SDbeta_000k.img`.
- An image of the standard deviation of the error `Sess1_SDerror.img`.
- An image `mask.img` indicating which voxels were included in the analysis.
- If a non-zero AR model order is specified then SPM also writes images `Sess1_AR_000p.img` where p indexes the pth AR coefficient.
- If contrasts have been specified SPM also writes `con_000i.img` and `con_sd_000i.img` which are the mean and standard deviation of the ith pre-defined contrast.

Each of these images can be inspected using the "Display" button. Type `help spm_spm_vb` at the MATLAB command prompt for further information.

9.4 Model comparison

Once you have estimated a model you can use SPM's results button to look at the results. You can also extract fMRI data from regions of interest using the ROI button. You can then compare GLMs based on different hemodynamic basis sets using the Bayesian model evidence.

This is described in [61] and implemented using the command line option `spm_vb_roi_basis`. This requires a VOI filename (created using the ROI button) and an SPM data structure. Type `help spm_vb_roi_basis` at the MATLAB command prompt for further information. Figure 9.3 shows an example output from the function indicating that, for the data in this brain region, an informed basis set has the highest model evidence.

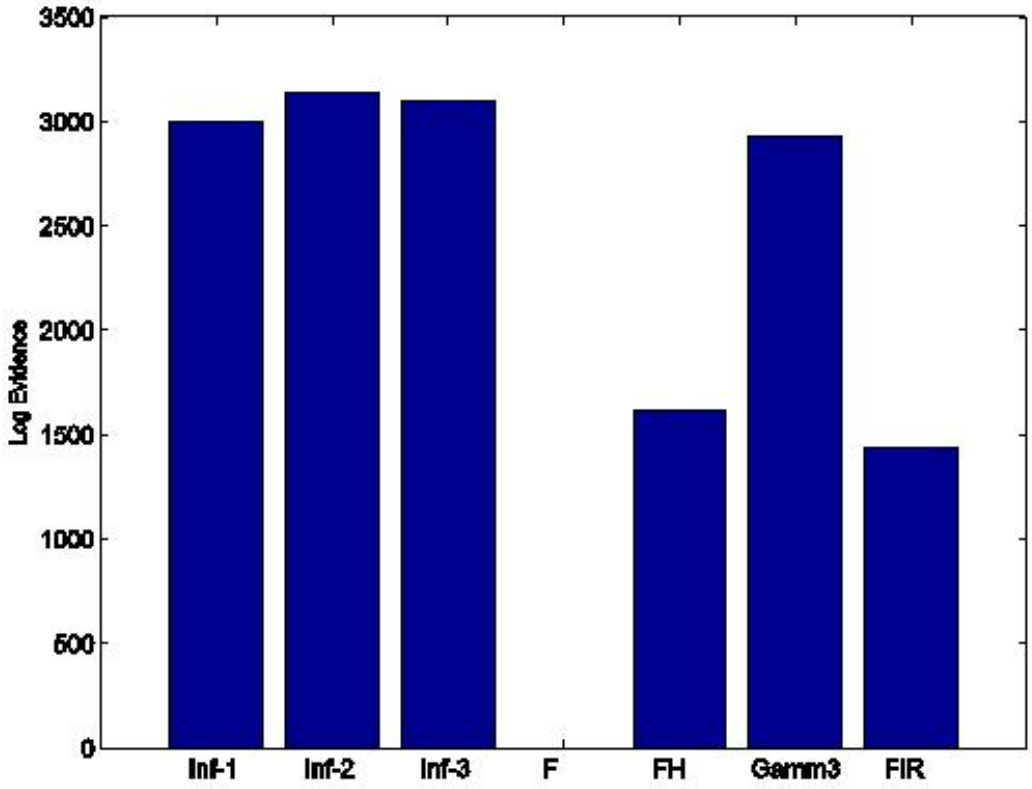


Figure 9.3: *This plot shows the model evidence for a number of different hemodynamic basis sets: Inf1 - Canonical HRF, Inf2 - Canonical plus temporal derivative, Inf3 - Canonical plus temporal and dispersion derivatives, F - Fourier, FH - Fourier with a Hanning Window, Gamm3 - 3 Gamma basis functions and FIR - a Finite Impulse Response function. An informed basis set provides the best model of the data for the selected region.*

Chapter 10

Factorial design specification

Contents

10.1 Directory 76
10.2 Design 76
10.2.1 One-sample t-test 76
10.2.2 Two-sample t-test 76
10.2.3 Paired t-test 77
10.2.4 Multiple regression 77
10.2.5 Full factorial 78
10.2.6 Flexible factorial 79
10.3 Covariates 81
10.3.1 Covariate 81
10.4 Masking 82
10.4.1 Threshold masking 82
10.4.2 Implicit Mask 82
10.4.3 Explicit Mask 82
10.5 Global calculation 83
10.5.1 Omit 83
10.5.2 User 83
10.5.3 Mean 83
10.6 Global normalisation 83
10.6.1 Overall grand mean scaling 83
10.6.2 Normalisation 84

This interface is used for setting up analyses of PET data. It is also used for '2nd level' or 'random effects' analysis which allow one to make a population inference. First level models can be used to produce appropriate summary data, which can then be used as raw data for a second-level analysis. For example, a simple t-test on contrast images from the first-level turns out to be a random-effects analysis with random subject effects, inferring for the population based on a particular sample of subjects.

This interface configures the design matrix, describing the general linear model, data specification, and other parameters necessary for the statistical analysis. These parameters are saved in a configuration file (SPM.mat), which can then be passed on to spm_spm.m which estimates the design. This is achieved by pressing the 'Estimate' button. Inference on these estimated parameters is then handled by the SPM results section.

A separate interface handles design configuration for fMRI time series.

Various data and parameters need to be supplied to specify the design (1) the image files, (2) indicators of the corresponding condition/subject/group (2) any covariates, nuisance variables,

or design matrix partitions (3) the type of global normalisation (if any) (4) grand mean scaling options (5) thresholds and masks defining the image volume to analyse. The interface supports a comprehensive range of options for all these parameters.

10.1 Directory

Select a directory where the SPM.mat file containing the specified design matrix will be written.

10.2 Design

10.2.1 One-sample t-test

Scans

Select the images. They must all have the same image dimensions, orientation, voxel size etc.

10.2.2 Two-sample t-test

Group 1 scans

Select the images from sample 1. They must all have the same image dimensions, orientation, voxel size etc.

Group 2 scans

Select the images from sample 2. They must all have the same image dimensions, orientation, voxel size etc.

Independence

By default, the measurements are assumed to be independent between levels.

If you change this option to allow for dependencies, this will violate the assumption of sphericity. It would therefore be an example of non-sphericity. One such example would be where you had repeated measurements from the same subjects - it may then be the case that, over subjects, measure 1 is correlated to measure 2.

Restricted Maximum Likelihood (REML): The ensuing covariance components will be estimated using ReML in spm_spm (assuming the same for all responsive voxels) and used to adjust the statistics and degrees of freedom during inference. By default spm_spm will use weighted least squares to produce Gauss-Markov or Maximum likelihood estimators using the non-sphericity structure specified at this stage. The components will be found in SPM.xVi and enter the estimation procedure exactly as the serial correlations in fMRI models.

Variance

By default, the measurements in each level are assumed to have unequal variance.

This violates the assumption of 'sphericity' and is therefore an example of 'non-sphericity'.

This can occur, for example, in a 2nd-level analysis of variance, one contrast may be scaled differently from another. Another example would be the comparison of qualitatively different dependent variables (e.g. normals vs. patients). Different variances (heteroscedasticy) induce different error covariance components that are estimated using restricted maximum likelihood (see below).

Restricted Maximum Likelihood (REML): The ensuing covariance components will be estimated using ReML in spm_spm (assuming the same for all responsive voxels) and used to adjust the statistics and degrees of freedom during inference. By default spm_spm will use weighted least squares to produce Gauss-Markov or Maximum likelihood estimators using the non-sphericity structure specified at this stage. The components will be found in SPM.xVi and enter the estimation procedure exactly as the serial correlations in fMRI models.

Grand mean scaling

This option is only used for PET data.

Selecting YES will specify 'grand mean scaling by factor' which could be eg. 'grand mean scaling by subject' if the factor is 'subject'.

Since differences between subjects may be due to gain and sensitivity effects, AnCova by subject could be combined with "grand mean scaling by subject" to obtain a combination of between subject proportional scaling and within subject AnCova.

ANCOVA

This option is only used for PET data.

Selecting YES will specify 'ANCOVA-by-factor' regressors. This includes eg. 'Ancova by subject' or 'Ancova by effect'. These options allow eg. different subjects to have different relationships between local and global measurements.

10.2.3 Paired t-test

Pairs

Pair Add a new pair of scans to your experimental design

Scans [1,2] Select the pair of images.

Grand mean scaling

This option is only used for PET data.

Selecting YES will specify 'grand mean scaling by factor' which could be eg. 'grand mean scaling by subject' if the factor is 'subject'.

Since differences between subjects may be due to gain and sensitivity effects, AnCova by subject could be combined with "grand mean scaling by subject" to obtain a combination of between subject proportional scaling and within subject AnCova.

ANCOVA

This option is only used for PET data.

Selecting YES will specify 'ANCOVA-by-factor' regressors. This includes eg. 'Ancova by subject' or 'Ancova by effect'. These options allow eg. different subjects to have different relationships between local and global measurements.

10.2.4 Multiple regression

Scans

Select the images. They must all have the same image dimensions, orientation, voxel size etc.

Covariates

Covariates

Covariate Add a new covariate to your experimental design

Vector Vector of covariate values

Name Name of covariate

Centering

Intercept

By default, an intercept is always added to the model. If the covariates supplied by the user include a constant effect, the intercept may be omitted.

10.2.5 Full factorial

This option is best used when you wish to test for all main effects and interactions in one-way, two-way or three-way ANOVAs. Design specification proceeds in 2 stages. Firstly, by creating new factors and specifying the number of levels and name for each. Nonsphericity, ANOVA-by-factor and scaling options can also be specified at this stage. Secondly, scans are assigned separately to each cell. This accomodates unbalanced designs.

For example, if you wish to test for a main effect in the population from which your subjects are drawn and have modelled that effect at the first level using K basis functions (eg. K=3 informed basis functions) you can use a one-way ANOVA with K-levels. Create a single factor with K levels and then assign the data to each cell eg. canonical, temporal derivative and dispersion derivative cells, where each cell is assigned scans from multiple subjects.

SPM will also automatically generate the contrasts necessary to test for all main effects and interactions.

Factors

Specify your design a factor at a time.

Factor Add a new factor to your experimental design

Name Name of factor, eg. 'Repetition'

Levels Enter number of levels for this factor, eg. 2

Independence By default, the measurements are assumed to be independent between levels.

If you change this option to allow for dependencies, this will violate the assumption of sphericity. It would therefore be an example of non-sphericity. One such example would be where you had repeated measurements from the same subjects - it may then be the case that, over subjects, measure 1 is correlated to measure 2.

Restricted Maximum Likelihood (REML): The ensuing covariance components will be estimated using ReML in spm_spm (assuming the same for all responsive voxels) and used to adjust the statistics and degrees of freedom during inference. By default spm_spm will use weighted least squares to produce Gauss-Markov or Maximum likelihood estimators using the non-sphericity structure specified at this stage. The components will be found in SPM.xVi and enter the estimation procedure exactly as the serial correlations in fMRI models.

Variance By default, the measurements in each level are assumed to have unequal variance.

This violates the assumption of 'sphericity' and is therefore an example of 'non-sphericity'.

This can occur, for example, in a 2nd-level analysis of variance, one contrast may be scaled differently from another. Another example would be the comparison of qualitatively different dependent variables (e.g. normals vs. patients). Different variances (heteroscedasticy) induce different error covariance components that are estimated using restricted maximum likelihood (see below).

Restricted Maximum Likelihood (REML): The ensuing covariance components will be estimated using ReML in spm_spm (assuming the same for all responsive voxels) and used to adjust the statistics and degrees of freedom during inference. By default spm_spm will use weighted least squares to produce Gauss-Markov or Maximum likelihood estimators using the non-sphericity structure specified at this stage. The components will be found in SPM.xVi and enter the estimation procedure exactly as the serial correlations in fMRI models.

Grand mean scaling This option is only used for PET data.

Selecting YES will specify 'grand mean scaling by factor' which could be eg. 'grand mean scaling by subject' if the factor is 'subject'.

Since differences between subjects may be due to gain and sensitivity effects, AnCova by subject could be combined with "grand mean scaling by subject" to obtain a combination of between subject proportional scaling and within subject AnCova.

ANCOVA This option is only used for PET data.

Selecting YES will specify 'ANCOVA-by-factor' regressors. This includes eg. 'Ancova by subject' or 'Ancova by effect'. These options allow eg. different subjects to have different relationships between local and global measurements.

Specify cells

Enter the scans a cell at a time

Cell Enter data for a cell in your design

Levels Enter a vector or scalar that specifies which cell in the factorial design these images belong to. The length of this vector should correspond to the number of factors in the design

For example, length 2 vectors should be used for two-factor designs eg. the vector [2 3] specifies the cell corresponding to the 2nd-level of the first factor and the 3rd level of the 2nd factor.

Scans Select the images for this cell. They must all have the same image dimensions, orientation, voxel size etc.

10.2.6 Flexible factorial

Create a design matrix a block at a time by specifying which main effects and interactions you wish to be included.

This option is best used for one-way, two-way or three-way ANOVAs but where you do not wish to test for all possible main effects and interactions. This is perhaps most useful for PET where there is usually not enough data to test for all possible effects. Or for 3-way ANOVAs where you do not wish to test for all of the two-way interactions. A typical example here would be a group-by-drug-by-task analysis where, perhaps, only (i) group-by-drug or (ii) group-by-task interactions are of interest. In this case it is only necessary to have two-blocks in the design matrix - one for each interaction. The three-way interaction can then be tested for using a contrast that computes the difference between (i) and (ii).

Design specification then proceeds in 3 stages. Firstly, factors are created and names specified for each. Nonsphericity, ANOVA-by-factor and scaling options can also be specified at this stage.

Secondly, a list of scans is produced along with a factor matrix, I. This is an nscan x 4 matrix of factor level indicators (see xX.I below). The first factor must be 'replication' but the other factors can be anything. Specification of I and the scan list can be achieved in one of two ways (a) the 'Specify All' option allows I to be typed in at the user interface or (more likely) loaded in from the matlab workspace. All of the scans are then selected in one go. (b) the 'Subjects' option allows you to enter scans a subject at a time. The corresponding experimental conditions (ie. levels of factors) are entered at the same time. SPM will then create the factor matrix I. This style of interface is similar to that available in SPM2.

Thirdly, the design matrix is built up a block at a time. Each block can be a main effect or a (two-way) interaction.

Factors

Specify your design a factor at a time.

Factor Add a new factor to your design.

If you are using the 'Subjects' option to specify your scans and conditions, you may wish to make use of the following facility. There are two reserved words for the names of factors. These are 'subject' and 'repl' (standing for replication). If you use these factor names then SPM can automatically create replication and/or subject factors without you having to type in an extra entry in the condition vector.

For example, if you wish to model Subject and Task effects (two factors), under Subjects->Subject->Conditions you can type in simply [1 2 1 2] to specify eg. just the 'Task' factor level. You do not need to eg. for the 4th subject enter the matrix [1 4; 2 4; 1 4; 2 4].

Name Name of factor, eg. 'Repetition'

Independence By default, the measurements are assumed to be independent between levels.

If you change this option to allow for dependencies, this will violate the assumption of sphericity. It would therefore be an example of non-sphericity. One such example would be where you had repeated measurements from the same subjects - it may then be the case that, over subjects, measure 1 is correlated to measure 2.

Restricted Maximum Likelihood (REML): The ensuing covariance components will be estimated using ReML in spm_spm (assuming the same for all responsive voxels) and used to adjust the statistics and degrees of freedom during inference. By default spm_spm will use weighted least squares to produce Gauss-Markov or Maximum likelihood estimators using the non-sphericity structure specified at this stage. The components will be found in SPM.xVi and enter the estimation procedure exactly as the serial correlations in fMRI models.

Variance By default, the measurements in each level are assumed to have unequal variance.

This violates the assumption of 'sphericity' and is therefore an example of 'non-sphericity'.

This can occur, for example, in a 2nd-level analysis of variance, one contrast may be scaled differently from another. Another example would be the comparison of qualitatively different dependent variables (e.g. normals vs. patients). Different variances (heteroscedasticy) induce different error covariance components that are estimated using restricted maximum likelihood (see below).

Restricted Maximum Likelihood (REML): The ensuing covariance components will be estimated using ReML in spm_spm (assuming the same for all responsive voxels) and used to adjust the statistics and degrees of freedom during inference. By default spm_spm will use weighted least squares to produce Gauss-Markov or Maximum likelihood estimators using the non-sphericity structure specified at this stage. The components will be found in SPM.xVi and enter the estimation procedure exactly as the serial correlations in fMRI models.

Grand mean scaling This option is only used for PET data.

Selecting YES will specify 'grand mean scaling by factor' which could be eg. 'grand mean scaling by subject' if the factor is 'subject'.

Since differences between subjects may be due to gain and sensitivity effects, AnCova by subject could be combined with "grand mean scaling by subject" to obtain a combination of between subject proportional scaling and within subject AnCova.

ANCOVA This option is only used for PET data.

Selecting YES will specify 'ANCOVA-by-factor' regressors. This includes eg. 'Ancova by subject' or 'Ancova by effect'. These options allow eg. different subjects to have different relationships between local and global measurements.

Specify Subjects or all Scans & Factors

Subjects

Subject Enter data and conditions for a new subject

Scans Select the images to be analysed. They must all have the same image dimensions, orientation, voxel size etc.

Conditions

Specify all Specify (i) all scans in one go and (ii) all conditions using a factor matrix, I. This option is for 'power users'. The matrix I must have four columns and as as many rows as scans. It has the same format as SPM's internal variable SPM.xX.I.

The first column of I denotes the replication number and entries in the other columns denote the levels of each experimental factor.

So, for eg. a two-factor design the first column denotes the replication number and columns two and three have entries like 2 3 denoting the 2nd level of the first factor and 3rd level of the second factor. The 4th column in I would contain all 1s.

Scans Select the images to be analysed. They must all have the same image dimensions, orientation, voxel size etc.

Factor matrix Specify factor/level matrix as a nscan-by-4 matrix. Note that the first row of I is reserved for the internal replication factor and must not be used for experimental factors.

Main effects & Interactions

Main effect Add a main effect to your design matrix

Factor number Enter the number of the factor.

Interaction Add an interaction to your design matrix

Factor numbers Enter the numbers of the factors of this (two-way) interaction.

10.3 Covariates

This option allows for the specification of covariates and nuisance variables. Unlike SPM94/5/6, where the design was partitioned into effects of interest and nuisance effects for the computation of adjusted data and the F-statistic (which was used to thresh out voxels where there appeared to be no effects of interest), SPM does not partition the design in this way anymore. The only remaining distinction between effects of interest (including covariates) and nuisance effects is their location in the design matrix, which we have retained for continuity. Pre-specified design matrix partitions can be entered.

10.3.1 Covariate

Add a new covariate to your experimental design

Vector

Vector of covariate values

Name

Name of covariate

Interactions

For each covariate you have defined, there is an opportunity to create an additional regressor that is the interaction between the covariate and a chosen experimental factor.

Centering

The appropriate centering option is usually the one that corresponds to the interaction chosen, and ensures that main effects of the interacting factor aren't affected by the covariate. You are advised to choose this option, unless you have other modelling considerations.

10.4 Masking

The mask specifies the voxels within the image volume which are to be assessed. SPM supports three methods of masking (1) Threshold, (2) Implicit and (3) Explicit. The volume analysed is the intersection of all masks.

10.4.1 Threshold masking

Images are thresholded at a given value and only voxels at which all images exceed the threshold are included.

None

No threshold masking

Absolute

Images are thresholded at a given value and only voxels at which all images exceed the threshold are included.

This option allows you to specify the absolute value of the threshold.

Threshold Enter the absolute value of the threshold.

Relative

Images are thresholded at a given value and only voxels at which all images exceed the threshold are included.

This option allows you to specify the value of the threshold as a proportion of the global value.

Threshold Enter the threshold as a proportion of the global value

10.4.2 Implicit Mask

An "implicit mask" is a mask implied by a particular voxel value. Voxels with this mask value are excluded from the analysis.

For image data-types with a representation of NaN (see spm_type.m), NaN's is the implicit mask value, (and NaN's are always masked out).

For image data-types without a representation of NaN, zero is the mask value, and the user can choose whether zero voxels should be masked out or not.

By default, an implicit mask is used.

10.4.3 Explicit Mask

Explicit masks are other images containing (implicit) masks that are to be applied to the current analysis.

All voxels with value NaN (for image data-types with a representation of NaN), or zero (for other data types) are excluded from the analysis.

Explicit mask images can have any orientation and voxel/image size. Nearest neighbour interpolation of a mask image is used if the voxel centers of the input images do not coincide with that of the mask image.

10.5 Global calculation

This option is only used for PET data.

There are three methods for estimating global effects (1) Omit (assumming no other options requiring the global value chosen) (2) User defined (enter your own vector of global values) (3) Mean: SPM standard mean voxel value (within per image fullmean/8 mask)

10.5.1 Omit

Omit

10.5.2 User

User defined global effects (enter your own vector of global values)

Global values

Enter the vector of global values

10.5.3 Mean

SPM standard mean voxel value

This defines the global mean via a two-step process. Firstly, the overall mean is computed. Voxels with values less than 1/8 of this value are then deemed extra-cranial and get masked out. The mean is then recomputed on the remaining voxels.

10.6 Global normalisation

This option is only used for PET data.

Global nuisance effects are usually accounted for either by scaling the images so that they all have the same global value (proportional scaling), or by including the global covariate as a nuisance effect in the general linear model (AnCova). Much has been written on which to use, and when. Basically, since proportional scaling also scales the variance term, it is appropriate for situations where the global measurement predominantly reflects gain or sensitivity. Where variance is constant across the range of global values, linear modelling in an AnCova approach has more flexibility, since the model is not restricted to a simple proportional regression.

'Ancova by subject' or 'Ancova by effect' options are implemented using the ANCOVA options provided where each experimental factor (eg. subject or effect), is defined. These allow eg. different subjects to have different relationships between local and global measurements.

Since differences between subjects may be due to gain and sensitivity effects, AnCova by subject could be combined with "grand mean scaling by subject" (an option also provided where each experimental factor is originally defined) to obtain a combination of between subject proportional scaling and within subject AnCova.

10.6.1 Overall grand mean scaling

Scaling of the overall grand mean simply scales all the data by a common factor such that the mean of all the global values is the value specified. For qualitative data, this puts the data into an intuitively accessible scale without altering the statistics.

When proportional scaling global normalisation is used each image is separately scaled such that it's global value is that specified (in which case the grand mean is also implicitly scaled to that value). So, to proportionally scale each image so that its global value is eg. 20, select <Yes> then type in 20 for the grand mean scaled value.

When using AnCova or no global normalisation, with data from different subjects or sessions, an intermediate situation may be appropriate, and you may be given the option to scale group, session or subject grand means separately.

No

No overall grand mean scaling

Yes

Scaling of the overall grand mean simply scales all the data by a common factor such that the mean of all the global values is the value specified. For qualitative data, this puts the data into an intuitively accessible scale without altering the statistics.

Grand mean scaled value The default value of 50, scales the global flow to a physiologically realistic value of 50ml/dl/min.

10.6.2 Normalisation

Global nuisance effects are usually accounted for either by scaling the images so that they all have the same global value (proportional scaling), or by including the global covariate as a nuisance effect in the general linear model (AnCova). Much has been written on which to use, and when. Basically, since proportional scaling also scales the variance term, it is appropriate for situations where the global measurement predominantly reflects gain or sensitivity. Where variance is constant across the range of global values, linear modelling in an AnCova approach has more flexibility, since the model is not restricted to a simple proportional regression.

'Ancova by subject' or 'Ancova by effect' options are implemented using the ANCOVA options provided where each experimental factor (eg. subject or effect), is defined. These allow eg. different subjects to have different relationships between local and global measurements.

Since differences between subjects may be due to gain and sensitivity effects, AnCova by subject could be combined with "grand mean scaling by subject" (an option also provided where each experimental factor is originally defined) to obtain a combination of between subject proportional scaling and within subject AnCova.

Part IV

EEG/MEG

Chapter 11

SPM for MEG/EEG overview

11.1 Welcome to SPM for M/EEG

With SPM8 you can analyze all kinds of MEG and EEG data. Our group has recently published many articles about M/EEG analysis, in particular about Source Reconstruction[1] and Dynamic Causal Modelling[2] (DCM), which is a spatio-temporal network model to estimate effective connectivity in a network of sources. All these new methods, mostly based on a Bayesian approach, have been implemented in SPM8. We provide a range of tools for the full analysis pipeline, i.e., you can take your raw data from the MEG or EEG machine, and put it through SPM, starting from the conversion of the data through to a statistical analysis of source reconstructed multi-subject data or Dynamic Causal Modelling.

Our overall goal is to provide an academic M/EEG analysis software package that can be used by everyone to apply the most recent methods available for the analysis of M/EEG data. As you may guess, this goal is quite ambitious because there is a large number of different M/EEG formats available, plus there are literally dozens of different analysis strategies that researchers would like to use. Clearly, our rather small group doesn't have the resources to cover all these different approaches. However, we made SPM for M/EEG as open as it possibly can be to allow researchers to use their favourite analysis software for specific processing steps. For example, it is possible to convert raw data to SPM8, then convert them to FieldTrip[3] or EEGLAB[4] (using an SPM conversion routine), use a couple of functions in these packages, convert back to SPM, and do source reconstruction or DCM. Any combination of processing steps should be possible, and we expect that this software-interoperability among analysis software packages (each with its own area of expertise) will lead to a boost of M/EEG researchers trying out new ways of analysing their data with a wide range of sophisticated methods. We are pleased to say that we have a formal collaboration with the excellent FieldTrip package (head developer: Robert Oostenveld, F.C. Donders centre in Nijmegen/Netherlands) on many analysis issues. For example, SPM and FieldTrip share routines for converting data to MATLAB, forward modelling for M/EEG source reconstruction and the SPM8 distribution contains a version of FieldTrip so that you can combine FieldTrip and SPM functions in your custom scripts. SPM and FieldTrip complement each other well, as SPM is geared toward very specific analysis tools as will be described below, whereas FieldTrip is a more general repository of different methods that can be put together in flexible ways to perform a variety of analyses. This flexibility of FieldTrip, however, comes at the expense of accessibility to a non-expert user. FieldTrip does not have a graphical user interface (GUI) and its functions are used by writing custom MATLAB scripts. By combining SPM8 and FieldTrip the flexibility of FieldTrip can be complemented by SPM's GUI tools and batching system. Within this framework, power users can easily and rapidly develop specialized analysis tools with GUIs that can then also be used by non-proficient MATLAB users. Some examples of such tools are available in the MEEG toolbox distributed with SPM. We will also be happy to

[1] Source Reconstruction: http://www.fil.ion.ucl.ac.uk/spm/doc/biblio/Keyword/EEG.html
[2] Dynamic Causal Modelling: http://www.fil.ion.ucl.ac.uk/spm/doc/biblio/Keyword/DCM.html
[3] FieldTrip: http://www.ru.nl/neuroimaging/fieldtrip/
[4] EEGLAB: http://sccn.ucsd.edu/eeglab/

include in this toolbox new tools contributed by other users as long as they are of general interest and applicability.

SPM's speciality is, of course, the statistical analysis of voxel-based images. For statistical analysis, we use exactly the same routines as SPM for fMRI users would. These are robust and validated functions based on the General Linear Model[5] (GLM) and Random Field Theory[6] (RFT). These routines have been developed and used in the fMRI field over many years and are equally applicable to multi- (or single-) subject M/EEG studies.

Furthermore, our group has invested heavily in establishing Bayesian approaches to the source reconstruction of M/EEG data. Good source reconstruction techniques are vital for the M/EEG field, otherwise it would be very difficult to relate sensor data to neuroanatomy or findings from other modalities like fMRI. Bayesian source reconstruction provides a principled way of incorporating prior beliefs about how the data were generated, and enables principled methods for model comparison. With the use of priors and Bayesian model comparison, M/EEG source reconstruction is a very powerful neuroimaging tool, which has a unique macroscopic view on neuronal dynamics.

In addition, we have taken the idea of Dynamic Causal Modelling (DCM) from the fMRI domain, and applied it to the M/EEG field. For M/EEG, DCM is a powerful technique, because the data are highly resolved in time and this makes the identifiability of neurobiologically inspired network models feasible. This means that DCM can make inferences about temporal precedence of sources and can quantify changes in feedforward, backward and lateral connectivity among sources on a neuronal time-scale of milliseonds. Note that DCM/fMRI won't do this for you; DCM/fMRI (or any other connectivity analysis in fMRI) looks at the modulation of connectivity by task, on a time-scale of seconds.

11.2 Changes from SPM5 to SPM8

As in SPM5, SPM8 provides tools for the analysis of EEG and MEG data. However, the SPM8 release is much more robust than the previous version, in many aspects such as conversion of data, source reconstruction, and dynamic causal modelling.

For three years, we have collected valuable experience for the analysis of M/EEG data, and received much valuable feedback from both FIL and external collaborators and users. We had plenty of opportunity to see which things worked well and what can be improved. One of our major insights was that writing a general routine for conversion of M/EEG data from their native to our SPM-format is a major effort. This is simply because there are so many different formats around and it is quite an undertaking for a small development team like ours to write stable software which can read all formats, some of which we have never seen ourselves. This had two consequences. The first is that we now collaborate with the developers of FieldTrip, who had already made available a wide range of MATLAB code to convert M/EEG data, and we thought it a good idea to let both SPM and FieldTrip use and develop the same library. The second major difference is that we changed the internal M/EEG format of SPM in many ways to make reading/writing and manipulating M/EEG data more robust and straightforward for the user. Effectively, we invested a lot of effort into rebuilding the SPM for M/EEG machinery almost from scratch.

There are a couple of other major changes from SPM5 to SPM8.

First, based on our source reconstruction work, we have implemented a number of new routines which provide for a robust and efficient source reconstruction, using Bayesian approaches. The resulting, voxel-based source reconstructions can then be analysed, at the group level, with the same well-tested routines that are used for fMRI data.

Second, Dynamic Causal Modelling, a network analysis for spatiotemporal M/EEG data, has been developed further over the past three years. The DCM routines for modelling evoked

[5]GLM: http://www.fil.ion.ucl.ac.uk/spm/doc/biblio/Keyword/GLM.html
[6]RFT: http://www.fil.ion.ucl.ac.uk/spm/doc/biblio/Keyword/RFT.html

responses or fields have been significantly improved both in functionality and speed. We now provide DCMs for modelling induced responses and local field potentials.

Third, there are now three ways of implementing M/EEG analyses in SPM. These are the graphical user interface, and two different scripting methods (batch analysis). These batch facilities come in handy for multi-subject studies. As in fMRI analysis, many processing steps are repetitive and it is now quite straightforward to automatize the software to a high degree.

Fourth, it is now possible to convert, in working memory, SPM data to FieldTrip or EEGLAB, and back. This feature makes it possible to use, within SPM, many FieldTrip and EEGLAB functions. For example, it is quite straightforward, using a script, to work within SPM, and use FieldTrip functions to do parts of the preprocessing.

The following chapters go through all the EEG/MEG related functionality of SPM8. Most users will probably find the tutorial (chapter 33) useful for a quick start. A further detailed description of the conversion, preprocessing functions, and the display is given in chapter 12. In chapter 13, we explain how one would use SPM's statistical machinery to analyse M/EEG data. The 3D-source reconstruction routines, including dipole modelling, are described in chapter 14. Finally, in chapter 16, we describe the graphical user interface for dynamical causal modelling, for evoked responses, induced responses, and local field potentials.

Chapter 12

EEG/MEG preprocessing – Reference

In this chapter we will describe the function and syntax of all SPM/MEEG preprocessing and display functions. This will be the most detailed description of the functions in this manual. Our goal is to provide a comprehensive description of how the software can be used to preprocess M/EEG data up to the point where one would use one of the source reconstruction techniques or statistical analysis of M/EEG channel data.

These functions can be called either from SPM's graphical user interface (GUI), from the MATLAB command line, or via the batch input system. We recommend beginners use the GUI first, because this will prompt SPM to ask for all relevant information needed to process the data. The batch input system is designed for repetitive analyses of data (eg. from multiple subjects) once the user knows what should be done, and in which order. The command line facilities are very useful for writing scripts, or using SPM's history-to-script functionality to generate scripts automatically. The command line use of SPM for M/EEG will require some MATLAB knowledge.

For the command line, we follow the concept of providing only one input argument to each function. This input argument is usually a structure (struct) that contains all input arguments as fields. This approach has the advantage that the input does not need to follow a specific input argument order. If an obligatory input argument is missing, the function will invoke the GUI and ask the user for the missing argument. When using the GUI, a function is called without any input argument, i.e. SPM will ask for all input arguments. If using the command line (e.g., with a script), you can specify all arguments in advance and effectively use SPM/MEEG functions in batch mode. We provide some sample batch scripts (`history_subject1.m`) in the `man\example_scripts` folder of the SPM8-distribution.

In the following we will go through the conversion of the data, specifics of the M/EEG format in SPM8, how to properly enter additional information about the channels, how to call FieldTrip-functions from SPM, a complete reference of all methods and functions, how to use the display, and finally how to script and batch the preprocessing.

12.1 Conversion of data

The first step of any analysis is the conversion of data from its native machine-dependent format to a MATLABbased, common SPM format. This format stores the data in a `*.dat` file and all other information in a `*.mat` file. The `*.mat` file contains the data structure `D` and the `*.dat` is the M/EEG data. The conversion facility of SPM is based on the "fileio" toolbox[1], which is shared between SPM8, FieldTrip and EEGLAB toolboxes and jointly developed by the users of these toolboxes. At the moment most common EEG and MEG data formats are supported. For some cases, it might be necessary to install additional MATLAB toolboxes. In this case an error message will be displayed with a link where the appropriate toolbox can be downloaded.

[1] fileio: http://www2.ru.nl/fcdonders/fieldtrip/doku.php?id=fieldtrip:development:fileio

If your data format is not recognized by "fileio", it will still try to read the file using the Biosig toolbox[2], if available. This can work for some EEG systems, particularly clinical ones. One of the consequences of using Biosig as 'fallback' is that you may see error messages from Biosig or mentioning Biosig if your data format is not presently supported. In such a case you can extend the "fileio" toolbox and contribute your code to us. See "fileio" page for details.

In the following we will describe the GUI-version for conversion. One could also convert data using the batch or a script. After clicking on the CONVERT button of the M/EEG GUI you will be asked to select the file to be converted. As a rule of thumb, if the dataset consists of several files, the file containing the data (which is usually the largest) should be selected. SPM can usually automatically recognize the data format and apply the appropriate conversion routine. However, in some cases there is not enough information in the data file for SPM to recognize the format. This will typically be the case for files with non-specific extensions (`*.dat`, `*.bin`, `*.eeg`, etc). In these cases the header-, and not the data-, file should be chosen for conversion and if it is recognized, SPM will locate the data file automatically. Note that SPM8 can also convert data in SPM5 format, where the the `*.mat` file should be selected. In some rare cases automatic recognition is not possible or there are several possible low-level readers available for the same format. For these cases there is an option to force SPM to use a particular low-level reader available with the batch tool or in a script (see below).

After the file is chosen, you will be asked ("Define settings?") to choose whether to define some settings for the conversion or "just" read'. The latter option was introduced to enable a simple and convenient conversion of the data with no questions asked. The resulting SPM M/EEG data file can then be explored with SPM's reviewing tool to determine the appropriate conversion parameters for the future. If the "just read" option is chosen, SPM will try to convert the whole dataset preserving as much data as possible. The other option - "yes" - lets you control all features of the conversion, to convert only the data that will be used in subsequent processing.

If this option is chosen, the next question will be whether to read the data as "continuous" or as "trials". Note that some datasets do not contain continuous data to begin with. These datasets can only be converted with the "trials" option.

If the "continuous" option is chosen you will be asked ("Read everything?") whether to convert the whole file ("yes") or a subset of it ("no"). If the answer is "no" you will be asked to specify the time window in seconds. Note that if a data file contains several concatenated long segments (e.g., if the recording was paused and then resumed) only one of these segments at a time can be converted as continuous. Therefore you should specify a time window which does not cross the boundaries between segments.

If the "trial" option is selected, the next question will be where to retrieve the information about trials. There are three options. If "data" is chosen, SPM will attempt to look for information about trials in the dataset. This option is suitable for datasets that are already epoched or datasets which contain information about trials. If "define" is selected, trials can be defined based on information about events which appears in the file. The routine used for this option is identical to the one used in epoching (see below) and the resulting SPM file will be already epoched. The advantage of defining trials at conversion is that only the necessary subset of the raw data is converted. This is useful when the trials of interest are only a small subset of the whole recording. After the trial definition is completed, the results can be saved into a file. This file can be later used instead of repeating the trial definition again. This is done by selecting the third trial definition option - "file".

The next question will be about which channels should be converted. Five options are available. "all" - convert all the available channels. "meg", "eeg" - automatically detect and choose MEG and EEG channels respectively. These options may not work correctly for some MEG and EEG systems because many data formats do not provide information about what data were acquired by a specific channel. However, the most common cases are supported. "gui" - choose the channels to convert using a graphical interface. The overall selection of channels can be saved in a file and this file can later be used by choosing the "file" option.

The final question is by which name the new SPM EEG files should be written to disk. By default SPM will add the prefix `spm8_` to the name of the raw data file if the data is read as continuous and `espm8_` if the data is read by trials.

[2]Biosig: http://biosig.sourceforge.net/

SPM will now convert the data. This may take some time depending on file size and a red bar will inform you about the progress.

12.2 Converting arbitrary data

It might be the case that your data is not in any standard format but is only available as an ASCII or Excel file or as a variable in the MATLAB workspace. Then you have two options depending on whether you would be willing to use a MATLAB script or want to only use the GUI. If you want to only use the GUI, we suggest that you assemble your dataset in EEGLAB and then save it and convert to SPM as any other format. The developers of EEGLAB have invested a lot of effort into making it possible to build a dataset from scratch without using the command line or scripts and we see no reason for reproducing the same functionality in SPM.

If you are willing to write a simple MATLAB script, the most straightforward way to convert your data would be to create a quite simple FieldTrip raw data structure (MATLAB `struct`) and then use SPM's `spm_eeg_ft2spm.m` function to convert this structure to SPM dataset. Missing information can then be supplemented using `meeg` methods and SPM functions.

FieldTrip raw struct must contain the following fields:

- `.fsample` - sampling rate (Hz)
- `.trial` - cell array of trials containing matrices with identical dimensions (channels $\times$ time).
- `.time` - cell array of time vectors (in sec) - one cell per trial, containing a time vector the same length as the second dimension of the data. For SPM, the time vectors must be identical.
- `.label` - cell array of strings, list of channel labels. Same length as the first dimension of the data.

If your data only has one trial (e.g. it is already an average or it is raw continuous data) you should only have one cell in `.trial` and `.time` fields of the raw struct.

An example script for converting LFP data can be found under `toolbox\Neural_Models\spm_lfp_txt2mat.m`

12.3 The M/EEG SPM format

The M/EEG format has changed from SPM5 to SPM8. There were many reasons why we decided that it was time to radically change the format. If you still have some data from your SPM5 analyses, you can convert them to SPM8 (see above).

Here are the reasons why we changed the format. Previously, we placed all header information into a struct, which was then stored in a MATLAB `mat`-file. When the data were read, the struct-file was put into working memory, and the data, contained in the `dat`-file, was linked to the struct, using memory-mapping. We found that making a struct available in working memory was highly problematic. This was because users would manipulate the struct but sometimes introduced an error to the format (e.g., they removed a few trials from the data but did not update all fields relating to the total number of trials). This would generate hard-to-resolve errors when trying to further process the now inconsistent data. To avoid this in the future, we have introduced two changes. The first is that SPM8 now always does a consistency check when loading a file. This means, if, for whatever reason, the data format was made inconsistent, SPM8 will now report this as soon the user tries to load these data. SPM8 will also report where the check flagged an inconsistency. If there is enough information available to fix the inconsistency, it will be fixed on the fly. For instance, you will usually see some messages about missing information when converting a file to SPM8 format. These messages are normal and if no error is generated they do not indicate a problem. Second, when reading the data, we now convert the header struct to an object and only then make it available in working memory. Generally, this MATLAB object can only be manipulated using standardized functions (called methods), which makes it very hard to introduce any inconsistency into SPM M/EEG data in the first place. Also, using methods simplifies internal book-keeping, which makes it much easier to program functions operating on

the M/EEG object. For example, while the SPM5 format kept a variable which contained the number of total trials, the SPM8 object does not have this variable but a method that returns the number of trials by simply counting how many trials of data there are. This makes it easy for a programmer to, for example, remove trials. There is no longer a need to update a variable for the number of trials. Also there is no need to check for the presence of particular fields in the struct in every SPM function. All the methods are guaranteed to work on a valid object. There are many more simplifications along these lines due to the use of "objects" and "methods". In summary, the change in format has resulted in a much more robust, and usable data format.

12.4 Preparing the data after conversion

SPM tries to do its best to extract information automatically from the various data formats. In some cases it can also supplement the converted dataset with information not directly present in the raw data. For instance, SPM can recognize common EEG setups (extended 1020, Biosemi, EGI) based on channel labels and assigns 'EEG' channel type and default electrode locations for these cases. However, there are data types which are either not yet supported in this way or do not contain sufficient information for SPM to make the automatic choices. Also the channel labels do not always correctly describe the actual electrode locations in an experiment. In these cases, further information needs to be supplied by the user. Reading and linking this additional information with the data is the purpose of the `Prepare` interface. This interface is accessed by selecting `Prepare` from the "Other" drop-down menu in the GUI. A menu (easily overlooked) will appear at the top of SPM's interactive window. The same functionality can also be accessed by pressing "Prepare SPM file" in the SPM M/EEG reviewing tool.

In this menu, an SPM M/EEG file can be loaded and saved using the "File" submenu. The "Channel types" submenu allows reviewing and changing the channel types. Use the "Review" option to examine the presently set channel types. During conversion, SPM will make an informed *guess* at the correct channel types but this can sometimes go wrong, especiallly for EEG data. To set a particular channel group to some channel type, select this type from the menu. A list of all channels will appear. Select the subset whose type you would like to set. `Ctrl` and `Shift` buttons can be used to refine the selection. Press OK to apply your choice. It is especially important to correctly specify which are the EEG channels. MEG types are assigned automatically by SPM and cannot be modified using the GUI.

The "Sensors" submenu can be used to supply information about the sensor positions to the file. This information is needed to perform 3D source reconstruction and DCM analysis for EEG and MEG data. Sensor positions for MEG are extracted from the raw data automatically and are already present. For EEG, sensor positions are usually measured by a special device (such as Polhemus) and are not part of the dataset. Even if you do not measure electrode positions routinely in your lab, we recommend to perform at least one initial measurement with the electrode cap you use and use the result as your standard template. In order for SPM to provide a meaningful interpretation of the results of source reconstruction, it should link the coordinate system in which sensor positions are originally represented to the coordinate system of a structural MRI image (MNI coordinates). In general to link between two coordinate systems you will need a set of at least 3 points whose coordinates are known in both systems. This is a kind of *Rosetta stone* that can be used to convert a position of any point from one system to the other. These points are called "fiducials" and the process of providing SPM with all the necessary information to create the *Rosetta stone* for your data is called "coregistration". The most commonly used fiducials are the nose bridge and the two pre-auricular points. The coordinates of these points for SPM's standard template image are hard-coded in SPM code. So if you provide the coordinates of these specific points with your sensor positions, it will be enough for SPM. If you do not have these fiducials but have other anatomical landmarks (for instance 3 EEG electrodes whose positions can be easily marked on a structural image) it will be possible to use them for coregistration as well, but that will require additional input from you. In addition, or as a replacement of fiducials a headshape measurement may be used. This measurement is done by an operator moving his digitizer pen around on the subject's scalp and generates many more data points than just 3 fiducials. EEG sensor and fiducial positions can be added to an SPM file using the "Load EEG sensors" menu. There are 3 options:

- "Assign default" - assigning default sensor positions. If this is possible, it will be done automatically at conversion but this option can be used to revert to default sensor positions after making some changes.

- "From a `*.mat` file" - this option is for the kind of files that were used in SPM5 and can also be used for any kind of locations without trying to get them into one of the standard formats. SPM will ask for two files. The sensors file should contain an $N \times 3$ matrix, where N is the same as the number of channels whose type is set to "EEG" and the order of the rows matches the order of these channels in the SPM file. The fiducials file should contain a $K \times 3$ matrix, where K (usually 3) is the number of fiducials. You will then be asked to provide labels for these fiducials. They should appear in the same order as the rows in the file.

- "Convert locations file" - this option uses a function from the internal "fileio" toolbox that supports several common formats for EEG channel position specification such as `*.sfp` and BESA's `*.elp`. It can also read Polhemus files from FIL and FCDC. In general Polhemus devices do not have a standard data format so if you are using Polhemus at a different site is is most likely that your Polhemus file will not be recognized by SPM directly. You will need to convert it to another format. An `*.sfp` file is the easiest to create (for instance in Excel). It is just an ASCII file containing a column of channel labels and 3 columns of cartesian coordinates. Check "fileio" website[3] for a complete list of supported formats. The file you are importing can also contain positions of fiducial points or any other named points that do not necessarily correspond to channels. The important thing is that there are coordinates for each channel that was assigned "EEG" type.

The fiducials for MEG are automatically loaded from the dataset. However, in some MEG setups the situation is more complicated. For instance, it might be convenient to attach the coils marking MEG fiducials to the top of the head, where there are no clear anatomical landmarks. In this case there should be an additional file measured with a Polhemus-like device that contains the positions of MEG fiducials and something that can be linked to a structural image (either anatomical landmarks or a headshape) in the same coordinate system. The way SPM handles this situation is in two steps. First, this additional file is converted into the same coordinate system in which MEG sensors are represented and it replaces the original MEG fiducials. At a later stage having MEG sensors and fiducials/headshape in the same coordinate system, SPM uses the fiducials/headshape for coregistration with standard MRI template or subject's own structural image. If you can mark the points where your MEG fiducial coils were located on a structural image, the step described below is not necessary. It is also possible that the digitizer measurement is stored with the raw data. Then SPM will read it automatically. Otherwise, the additional fiducials/headshape file can be loaded using the "Load MEG Fiducials/Headshape" menu. The supported formats are the same as for electrode locations. It is also possible to create a fiducials/headshape MATLAB struct and store it in a `*.mat` file. This file will also be recognized by SPM. The struct should be called `shape` and it should contain the following fields: `shape.pnt` - a $K \times 3$ matrix (possibly empty) with headshape points i.e. points that are on the surface of the head and have no labels, `shape.fid.pnt` - $M \times 3$ matrix with fiducial points i.e. points that have labels, `shape.fid.label` - $M \times 1$ cell array of strings with the labels of the points in `shape.fid.pnt`. As mentioned above, M should be at least 3 for the coregistration to work.

If you did not use default 3D positions, after loading the sensor positions you can perform coregistration of your sensors with SPM's template head model. This initial alignment is helpful to verify that the sensor information you supplied were interpreted correctly and should also be done if you would like to generate a 2D sensor template based on your 3D sensor positions (see below). The 2D-coordinates will be used for displaying the data in a topologically meaningful way. This is implemented using the "Coregister" option. For details of how this option works see the 3D source reconstruction chapter 14.

The "2D Projection" menu deals with the generation of representative 2D-coordinates for the sensors. Note that generating 2D-coordinates is not obligatory. If the 2D-coordinates are not specified, the sensors will be, when displaying, presented in a default square grid. Missing out

[3] fileio: http://www2.ru.nl/fcdonders/fieldtrip/doku.php?id=fieldtrip:dataformat

on topographically meaningful 2D-coordinates might be useful when working on few channels. The 2D-coordinates are also used for producing scalp-level SPMs in voxel space when converting M/EEG data to images for later statistical analysis (see below). If you are planning to do 3D source reconstruction or DCM, 2D-coordinates are not necessarily required. Also, you can load 2D-coordinates from a file (several example files are available in the `EEGtemplates` SPM directory). For EEG, 2D-coordinates can also be generated by projecting the 3D sensor positions to a plane. This is done automatically when default 3D coordinates can be assigned, and also for MEG. In case of custom EEG sensor positions coregistration should be performed first (see above). The resulting 2D-coordinates are displayed in SPM's graphics window. You can modify these projected 2D-coordinates manually by adding, deleting and moving sensors. To select a sensor, click on its label. The label will change its color to green. If you then click at a different location, the sensor will be moved to this position. Note that, at this stage, SPM does not check whether there is any correspondence between the labels of the coordinates and the labels of the channels stored in the SPM file. When you are satisfied with the 2D-coordinates, select "Apply" from the menu and the coordinates will be assigned to EEG or MEG channels according to their labels. Note that 2D-coordinates cannot be assigned to channels of other types than M/EEG.

Remember to save the file using "File/Save" after you finished modifying it using the `Prepare` interface. Your changes will not be saved automatically. In case of invoking `Prepare` from the reviewing tool you should press the 'OK' button that will appear at the bottom left of the interactive window, and then save the file with the "Save" button of the reviewing tool.

In the rare case that you neither have measured sensor locations, or fiducials, and the supplied standard templates do not work for you, you can also supply a so-called channel template file, which contains all information necessary. However, remember, that if you do not supply any 2D-coordinates, you can still use all SPM functions, however, SPM will use 2D-coordinates laid out in a topographically unmeaningful rectangular pattern.

A channel template file contains four variables:

`Nchannels`	-	The number of channels
`Cnames`	-	A cell vector of channel names. Each cell can contain either a string or a cell vector of strings. The latter allows for multiple versions of a given channel name. Case can be ignored, i.e., it doesn't matter whether channel names are in small or capital letters.
`Cpos`	-	A $2 \times Nchannels$-matrix of channel coordinates on a 2D plane. In x- and y-direction the minimum coordinate must be ≤ 0.05 and the maximum coordinate must be ≥ 0.95.
`Rxy`	-	A factor that determines the ratio of the width of the display plots to their height when displaying the data. Standard is 1.5.

Note that the channel template files and 3D coordinate files with labels (such as `*.sfp`) can contain many more channel labels than your data file. SPM searches, for each channel in the data, through the labels in the channel template file. If the labels match, the coordinate is used.

12.5 Integration of SPM and Fieldtrip

The SPM8 distribution includes the latest version of the FieldTrip toolbox[4]. FieldTrip is a MATLAB toolbox for MEG and EEG analysis that is being developed at the F.C. Donders Centre (FCDC) together with collaborating institutes. FieldTrip functions can be used for many kinds of analysis which are not supported in SPM proper. However, FieldTrip does not have extensive graphical user interface and its functionality should be accessed by writing scripts. Full reference documentation for FieldTrip including example scripts is available at the FieldTrip website. The SPM distribution also contains some documentation, contained as help comments in FieldTrip functions. These can be found in the directory `external\fieldtrip\private`. Note that in order to prevent function name clashes SPM calls its FieldTrip functions via intermediate or "wrapper" functions whose name always starts with `ft_`. This has the advantage that even if

[4]FieldTrip: http://www.ru.nl/neuroimaging/fieldtrip/

there is a different FieldTrip version in the Matlab path from the one used by SPM, SPM will only use its own version and incompatibilities will be avoided. To adapt a standard FieldTrip script for use with SPM you must add the prefix `ft_` to all FieldTrip function names in the script.

Fieldtrip data structures can be converted to SPM EEG files using the `spm_eeg_ft2spm` function. SPM M/EEG data, once loaded with the function `spm_eeg_load` can be converted to FieldTrip format using the method `ftraw` (with syntax `D.ftraw` or `ftraw(D)`).

12.6 Reading of data

If you use the GUI only, there is no need to read this section because the functions called by the GUI will read the data automatically. However, if you plan to write scripts and access the data and header information more directly, this section should contain all the necessary information to do so.

An SPM8 for M/EEG file can be read using the `spm_eeg_load` function. Without any arguments a file requester asks for the name of the file. With a string argument P, `spm_eeg_load(P)` will attempt to read a file with the name `P`. The behavior in this case is identical to the "Just read" option in the GUI. The SPM-format stores the binary data in a `*.dat` file. All header information are stored in a `*.mat` file. This `*.mat` file contains a single struct named `D` which contains several fields. When using `spm_eeg_load`, the struct is transformed into an object, and the data are linked into this object. The linking is done via memory mapping using `file_array` objects. Note that the data should always be read using the routine `spm_eeg_load`. The memory mapped data can be addressed like a matrix (see below) which is convenient for accessing the data in a random access way. However, a word of caution: If you write new values to this matrix, the matrix is not only changed in the object (in memory), but also physically on the hard disk.

You can also load the header struct using a simple MATLAB `load` but this just returns the header struct, without the data linked in, and SPM functions won't know what to do with this struct. In the following, we will describe the methods that one can use on an M/EEG-object. Note that we will not describe the internal format of the data here. This would be helpful only for programmers who want to write new methods for `meeg` object, because when simply analyzing M/EEG or even writing SPM functions there should never be a need to access the internal format. If you think that the existing methods do not provide the functionality you need, please let us know. For interested power users, there is documentation about the internal format within the `meeg` class constructor. *meeg*.

12.6.1 Syntax

```
D = spm_eeg_load(P)
```

Input

The input string `P` is optional and contains the file name of the `*.mat` file.

Output

The output struct `D` contains all header information about the data. The data are memory mapped and can be accessed directly using something like `d = D(:,:,1)`. This command would put the first trial over all channels and time points into the variable `d`. The first dimension of `D` is channels, the second peri-stimulus time, and the third is trials. If the data are time-frequency transformed, there would be four dimensions, where the frequency dimension is squeezed in at the second position (i.e., channels/frequencies/time/trials). If you wanted to change the values of the data, you would write something like `D(1,2,3) = 1;`, which would change the value of the first channel, second time-point, and third trial to 1.

12.7 Methods for the meeg object

meeg methods are functions that operate on an meeg object, loaded with spm_eeg_load. These methods should not be used, if you just want to analyze your data using the GUI or simple scripts. However, if you write your own scripts or high-level functions that need to read or manipulate the object, you need these methods. In the following, we will provide details about most of the methods. The code for all methods can be found in the @meeg SPM directory. Most methods provide some minimalist help text. In the following, we will assume that the object variable is called D, i.e. previously loaded by using D = spm_eeg_load;.

12.7.1 Constructor meeg

The meeg method is a constructor. Called without any arguments it will produce a consistent, but empty object. In SPM, the constructor is called when a struct has been loaded into memory by *spm_eeg_load*, and is transformed into an meeg object. Importantly, the constructor also checks the consistency of the object.

12.7.2 display

This method will return, in the MATLAB window, some information about the object, e.g., display(D).

12.7.3 Number methods

These are methods which return the number of something; they count the number of channels, etc. For example, to find out how many channels an MEEG object contains, you would use D.nchannels, where D is the object. Number functions are nchannels, nfrequencies, nsamples, ntrials.

12.7.4 Reading and manipulation of information

There are a large number of methods that can be used to either read or write some information. The method name is the same but it depends on the arguments whether something is read or stored. For example, when you use the method badchannels, you can either type D.badchannels, which returns the indices of all bad channels. You could also change information about specific bad channels, e.g., D.badchannels([43:55], 1) will flag channels 43 to 55 as bad. You could also use D.badchannels([43:55], ones(1,13), i.e. you can either use a scalar to change all channels listed, or supply a 0/1-flag for each channel. There are other functions which use the same logic. In the following we will list these functions and describe briefly what they do but won't go into much detail. We believe that you can work it out using the badchannels-example.

selectdata

With this method the data can be indexed using channel labels, times and condition labels instead of indices which you would usually need to find out in your code. For instance D.selectdata('Cz', [0.1 0.12], 'Oddball') will return the waveforms of channel Cz between 100 and 120 ms in peristimulus time for the condition "Oddball".

chanlabels

This method reads or writes the label of the channels (string). Note that the channel labels must be unique.

chantype

This method reads or writes the type of a channel (string). Currently, the types recognized by SPM are: "EEG", "MEG", "EMG", "EOG", or "Other", but in principle type can be any string.

conditions

This method reads or writes the name of the condition of an epoch (string).

events

This method returns the events stored with each trial. Events are records of things that happened during the experiment - stimuli, responses, etc. Before a file is epoched all the events are stored with the only trial and they can be used by the epoching function. For an epoched file SPM stores with each trial the events that occured within that trial or possibly in some time window around it (this is a parameter of the epoching function that can be specified). You can use this information for your analysis (for instance to sort trials by reaction time). Events are represented by a structure array with the following fields:

- `.type` - string (e.g. "front panel trigger")
- `.value` - number or string, can be empty (e.g. "Trig 1").
- `.time` - in seconds in terms of the original file
- `.duration` - in seconds (optional)

Note that in order to find out the time of an event in peristimulus time you will need additional information provided by "trialonset" method.

fname

This method reads or writes the name of the `mat`-file, in which the header information are stored.

fnamedat

This method reads or writes the name of the `dat`-file, in which the data are stored.

frequencies

If the data has been transformed to time-frequency, this method reads or writes the frequencies (Hz) of the data.

fsample

This method reads or writes the sampling rate of the data. In SPM, all data must have the same sampling frequency.

history

This method can read or add to the history of a file. Usually, each time a SPM function (e.g. like converting) does something to a data set, the function name and arguments (possibly after collecting them with the GUI) are stored in the history. Effectively, the history is a documentation of how exactly the data were processed. Of course, the history function can also be used to replicate the processing, or generate (modifiable) scripts for processing other data in the same way.

path

This method reads or writes the path, under which the `mat`- and `dat`-files are stored.

reject

This method reads or writes the indices of rejected (bad) trials.

timeonset

This method reads and writes the time of the first sample in a trial in peristimulus time (in seconds). In SPM all trials should have the same time axis. Therefore there is only one timeonset in a file. For instance, if you have a pre-stimulus baseline of 100 ms and the stimulus comes at time zero, `timeonset` will be -0.1. In general it is possible to define the time axis any way you like and there is no requirement that the stimulus comes at 0 or that there is baseline with negative times (which was the case in SPM5).

trialonset

This method should not be confused with the more commonly used `timeonset` (see above). It returns the times of the first sample of each trial in the original raw data file time. This information is not always available to begin with. It may also be invalidated during processing (for instance if you merge two epoched files). When this happens the information is discarded. For SPM analysis `trialonset` is not usually necessary. However it may be useful if you want to relate something in your analysis to the timing of your experiment, for instance create a regressor for GLM analysis of single trials to account for fatigue. `trialonset` is also necessary for interpretation of events in epoched files.

transformtype

This method reads and writes the type of the data transform (string). For example, when the data are transformed to a time-frequency represention, `transformtype` is set to "TF". For time data, this is "time".

type

This method reads and writes the type of the data (string). Currently, this string can be "continuous", "single", "evoked", or "grandmean".

units

This method reads and writes the units of the measurements (string). The units are channel-specific, i.e., each channel can have its own units.

12.7.5 Reading of information

Some methods can only read information but not change them. These are:

condlist

This method returns a list of condition labels. Multiple entries of labels have been removed.

coor2D

This method returns the 2D-coordinates used for displaying or writing sensor data to voxel-based images.

dtype

This method returns the type under which the data are stored in the `file_array` object.

eogchannels

This method returns which of the channels are EOG channels.

indsample

This method returns the index of the sample which is closest to a specific time point (ms).

meegchannels

This method returns the indices of all channels that are either of the MEG or EEG type.

modality

This method returns the modality of channels (MEG, EEG, etc.).

pickconditions

This method returns the indices of trials of a certain condition. The condition must be supplied by its label (string).

repl

This method returns the number of replications measured for a condition. This method is usually only used on single trial data.

time

This method returns the time (ms) of the samples.

sensors

This method returns the sensor locations structure. There is an additional argument for modality ('EEG' or 'MEG') as we are planning to support datasets with more than one sensor type. The exact way sensors are represented depends on the modality and you can find more information in Fieldtrip documentation as the sensors structure is produced and used by code originally developed at FCDC. Note that in SPM, sensors are not directly linked with channels, unlike for instance in EEGLAB. So there is no requirement for the number of sensors and channels to match or even for any relation between them. Of course loading sensors completely unrelated to your data will not be very useful and will eventually lead to an error. This kind of representation is more powerful than a simple correspondence and it will become even more useful with further development of SPM.

fiducials

This method returns the fiducials. They are represented as `shape` struct (see the discussion of loading fiducials by the `Prepare` function) with an additional field for units that is assigned automatically.

12.7.6 Manipulation of information

There are two functions which only manipulate the objects.

ftraw

This method converts an object to a FieldTrip structure. An additional argument can be supplied to indicate whether the data is memory mapped (1: default) or loaded into memory (0). Note that not all FieldTrip functions can properly handle memory mapped data. In order to avoid corrupting the data, `ftraw` sets a read-only flag on it so in the worst case you might encounter errors in Fieldtrip functions. Please report such errors on the SPM or Fieldtrip mailing list so that we can fix them. `ftraw(0)` is safer in this respect but inefficient in its memory use.

save

This method saves the object to the `mat`- and `dat`-files.

12.7.7 struct-like interface

In addition to pre-defined internal fields that should only be manipulated using methods, the `meeg` object also allows storing additional information in it as long as the names of additional fields do not clash with the names of existing methods. This functionality is used by some SPM functions. For instance, the results of 3D source reconstructions are stored in `D.inv` field for which no methods are necessary to access and modify it. You can use this functionality in your scripts (try commands like `D.myfield = 'hellow world'; disp(D.myfield);`). The methods `rmfield` and `isfield` work for these extra-fields as they would if the `meeg` object was a struct.

12.8 SPM functions

In this section we will describe the high-level SPM functions which are used for preprocessing M/EEG data. These functions are fairly standard and should allow a simple preprocessing of the data (e.g., epoching, filtering, averaging, etc.). Here, we will just describe what each function roughly does and what the input arguments mean. More detailed information about the syntax can be found in the help text of the code. For example, to get detailed help on epoching, type `help spm_eeg_epochs`. The general syntax is the same for all functions. If called from the command-line, and if no input arguments are specified, the function will behave exactly as if you called the function from the GUI by pressing a button or choosing it from the "Other" menu. However, on the command line, or from a script, you can supply input arguments, up to the point when all required input arguments are specified, so that the function will run without any user interaction. In this way, one can write a script that runs without user interaction. See the folder `man\example_scripts` for an example. Input arguments are provided in a struct `S`, whose fields contain the arguments. A typical call, e.g., from a script would be: `D = spm_eeg_epochs(S)`, where S is the input struct, and `D` contains the return argument, the epoched `meeg` object. Note that, with all SPM functions, the object is also always written to hard disk. The filenames of the `mat`- and `dat`-files are generated by prepending a single letter to the old file name. In the example of epoching this would be an *'e'*. The idea is that by calling a sequence of functions on a file, the list of first letters of the file name shows (roughly) which preprocessing steps were called to produce this file. Note that another way of calling SPM functions and specifying all input parameters is to use the new batch interface.

12.8.1 Epoching the data: `spm_eeg_epochs`

Epoching cuts out little chunks of continuous data and saves them as "single trials". In M/EEG research, this is a standard data selection procedure to remove long gaps between each single trial. For each stimulus onset, the epoched trial starts at some user-specified pre-stimulus time and end at some post-stimulus time, e.g. from -100 to 400 milliseconds in peri-stimulus time. The epoched data will also be baseline-corrected, i.e. the mean of the pre-stimulus time is subtracted from the whole trial. The resulting event codes are the same as saved in the `*.mat` file. The prepended output letter is *'e'*.

The epoching function can deal with two different ways of specifying trials you want to epoch. The first is to specify explicitly where each trial is located in the measured time-series. The second is to specify trials using the labeled events stored in the file. For most users, the second way is the most convenient, but sometimes, when the stored triggers in the file do not relate to what you want to epoch or there is no event information available in the raw data (e.g. when you have an external stimuli file), you should use the first.

For the first method, you specify a $N \times 2$ so-called `trl` matrix, where each row contains the start and end of a trial (in samples). Optionally, there can be a third column containing the offset of the trigger with respect to the trial. An offset of 0 (default) means that the first sample of the trial corresponds to the trigger. A positive offset indicates that the first sample is later than the trigger, a negative offset indicates that the trial begins before the trigger. In SPM the offset should be the same for all trials. The need to specify a whole column is for interoperability with FieldTrip where trials can have different time axes. In addition you have to specify `conditionlabels` (a single string or a cell array of strings), either one for each trial or one for all trials. You can also enter a `padding` which will add time points before and after

each trial to allow the user to later cut out this padding again. This is useful, e.g., for filtering epoched data, where one would otherwise, without padding experience "edge effects".

For the second method, one should define the pre- and post-stimulus interval, and also specify the events (triggers) around which the epochs will be "cut". SPM identifies events by their "event type" and "event value". These are strings or numbers which the software run by the EEG or MEG vendor uses when generating the measurement file. If you don't specify parameters for the epoching function, a GUI will pop up, and present the found triggers with their type and value entries. These can sometimes look strange, but if you want to run a batch or script to do the epoching, you have to first find out what the type and value of your event of interest are. Fortunately, these tend to be the same over scanning sessions, so that you can batch multi-subject epoching using the types and values found in one subject. You also have to come up with a "condition label" for each trial type, which can be anything you choose. This is the label that SPM will use to indicate the trial type of a trial at later processing stages. It is possible to use several types of triggers for defining trials with the same label - in the GUI, just select several events using `Shift` or `Ctrl` key.

For both methods of input you also have to set a (0/1)-flag (no/yes) indicating whether you want to review the information for all trials after selecting them. This allows you to make sure that all your trials are there. You should set the review-flag to 0, if you write a non-interactive script. In the GUI you can review a list of the epochs you defined and exclude some of them by hand (e.g. only take the triggers from the first 5 min of recording). You can also choose to save the trial definitions, for example, for re-use of another epoching of the same data.

12.8.2 Filtering the data: `spm_eeg_filter`

Continuous or epoched data can be filtered, over time, with a low-, high-, stop- or bandpass-filter. SPM uses a Butterworth filter to do this. Note that SPM uses the signal processing toolbox. This means that you have to have this toolbox to filter data in SPM. Phase delays are minimised by using MATLAB 's `filtfilt` function which filters the data twice, forwards and backwards. The prepended output letter is *'f'*.

When you use the function in GUI mode, it will automatically use the Butterworth-filter. You can then choose among the four different types of filtering: "lowpass", "highpass", "bandpass", and "stopband". Depending on your choice, SPM will ask for the cutoff(s) in Hz.

12.8.3 Artefact detection and rejection: `spm_eeg_artefact`

Some trials not only contain neuronal signals of interest, but also a large amount of signal from other sources like eye movements or muscular activity. These signal components are referred to as artefacts. In SPM, we use two simple automatic artefact detection schemes. The first is thresholding the data, and the second is robust averaging. One can also choose to detect artefacts manually by visualizing each trial using the display. Another option is to use a more sophisticated artefact detection approach (implemented by other software) and supply that information to SPM.

Thresholding of the data is implemented in two passes. In the first, SPM detects all instances, over trials and channels, where the absolute value is higher than the threshold. If a channel has more than a certain percentage of artefactual trials, it is defined as a bad channel. In a second pass the thresholding is repeated, but without taking into account bad channels. A trial for which the absolute data surpasses the threshold in some channel (excluding bad channels) is then considered artefactual and flagged as a rejected trial.

Note that the function only indicates which trials are artefactual or clean and subsequent processing steps (e.g. averaging) will take this information into account. However, no data is actually removed from the `*.dat` file. The `*.dat` file is actually copied over without any change. The prepended output letter is *'a'*.

When you call the function, you are first asked whether you want to read your own artefact list. This gives you the opportunity to tell SPM which of your trials are artefactual, and which are clean. The two lists of trial numbers don't need to be complete, i.e., you can specify a few trials as artefactual or clean but SPM will still check for artefacts in the remaining trials. The next question is whether you want to use robust averaging for your data. This approach estimates weights, lying between 0 and 1, that indicate how artefactual a trial is. Later on, when averaging

to produce evoked responses, each trial is weighted by this number. For example, if the weight of a trial is close to zero, it doesn't have much influence in the average, and is effectively treated like an artefactual trial.If you choose robust averaging, you first have to choose an offset for the weighting function. This value, default value 3, defines the weighting function used for averaging the data. The value 3 will roughly preserve 95% of data points drawn randomly from a Gaussian distribution. Next, one has to specify the width of a smoothing kernel for the residuals. As robust averaging treats each data point as independent it is necessary to smooth the data after averaging. The value is specified in milliseconds and its default is 20.

SPM will next ask whether you want to threshold your channels. If you choose yes, you can then enter a list of channels, for which you want to check whether the absolute values of a trial surpass this threshold. Next, the threshold itself can be either a vector of thresholds, i.e., one for each channel, or a single threshold, which SPM uses for all channels.

12.8.4 Downsampling: `spm_eeg_downsample`

The data can be downsampled to any sampling rate. This is useful if the data was acquired at a higher sampling rate than one needs for making inferences about low-frequency components. For example, resampling from 1000 Hz to 200 Hz would cut down the resulting file size to 20% of the original file size. Note that SPM's downsampling routine uses the MATLAB function `resample`, which is part of the signal processing toolbox. The prepended output letter is *'d'*.

Here, you choose the new sampling rate (Hz) which must be smaller than the old sampling rate.

12.8.5 Rereferencing: `spm_eeg_montage`

Sometimes it is necessary to re-reference the data to a new reference. In SPM this is done by specifying a weight matrix, which pre-multiplies the data. This is a general approach which allows one to re-reference to the average over channels, to single channels, or any linear combination of channels, e.g. the average over a pair of channels. The prepended output letter is *'M'*. Note that re-referencing should mainly affect the analysis you do at the channel level and should have little effect on 3D source reconstruction and DCM as long as it is not too noisy.

When you call the function, you will first be asked whether you want to use a GUI or read information from a file to specifying the montage. If you choose GUI, you will see, on the left hand side, the montage-matrix, where each row stands for a new channel. This means the labels in the left column describe the new labels. The old labels are on top, that means, each row contains weights for how the old channels must be weighted to produce new channels in the montage. On the right hand side, you see a graphical representation of the current matrix. The default is the identity matrix, i.e., the montage will not change anything. The concept is very general. For example, if you want to remove channels from the data, just delete the corresponding row from the montage matrix. To re-reference to a particular channel the column for this channel should be -1 for all rows, except the row corresponding to itself which should be 0, whereas the other channels should have 1 in the intersection of their column and row (the diagonal of the matrix) and 0 elsewhere. For average reference the matrix should have $(N-1)/N$ (where N is number of channels) at the diagonal and $-1/N$ elsewhere. In principle, any montage can be represented this way. If you are not sure about how to represent a montage you need, ask an expert or write to the SPM mailing list. The specification will only need to be done once for your setup and then you can save the montage and use it routinely. After changing the weights of the matrix, you can check the montage by pressing the button in the lower right below the figure.

If you choose to specify the montage by "file", you have to enter the filename of a `mat`-file, which contains a struct with 3 fields: `labelnew` (labels of new channels), `labelorg` (labels of original channels), and the montage-matrix `tra` ("tra" as in transform).

Finally, you will be asked, whether you want to "keep the other channels". There may be channels that are not involved in the montage. For instance, if you the apply montage defined for your EEG channels but there are also EOG or trigger channels in the file. If you answer "yes", they will just be copied to the new file unaffected. If you answer "no" they will not be included in the new file.

12.8.6 Grand mean: `spm_eeg_grandmean`

The grand mean is usually understood as the average of evoked responses over subjects. The grand mean function in SPM is typically used to do exactly this, but can also be used to average over multiple EEG files, e.g. multiple sessions of a single subject. The averaged file will be written to the same directory as the first selected file. The prepended output letter is *'g'*.

The function will ask you for the name of the output file. Note that in a script, by default, when you don't specify an output filename, SPM will generate a new file with the filename of the first selected file, prepended with a *'g'*.

12.8.7 Merge: `spm_eeg_merge`

Merging several MEEG files can be useful for concatenating multiple sessions of a single subject. Another use is to merge files and then use the display tool on the concatenated file to be able to display in the same graph data coming from different files. This is the preferred way in SPM to display data together that is split up into several files. The merged file will be written into the same directory as the first selected file. The prepended output letter is *'c'*.

The function will first check whether there are at least two files, and whether all data are consistent with each other, i.e., have the same number of channels, time points, and sampling rate. The function will also ask for a number of condition labels, one for each file and condition. This gives you the opportunity to rename labels. This might be useful when you merge files which contain the same conditionlabels, e.g. when you used several sessions for one subject but measured the same conditions in all files. In this case, it might be helpful to rename the conditions like "condition A" to something like "condition A, session 1", etc.

12.8.8 Time-frequency decomposition: `spm_eeg_tf`

The time-frequency decomposition uses a continuous Morlet wavelet transform. The result is written to one or two result files, one containing the instantaneous power and the other, optionally written, the phase estimates. One can select the channels and frequencies for which power and phase should be estimated. Optionally, one can apply a baseline correction to the power estimates, i.e. the mean power of a pre-stimulus interval is subtracted from the power estimates. For power, the prepended output letters are *t1_*, for phase *t2_*.

The function will first ask you, after selecting the M/EEG file, for a list of frequencies, which is a vector of numbers (Hz). For each of these frequencies, SPM will estimate the power and phase at each channel, time-point and trial. The next question is whether you want to remove the "baseline" (i.e. the average power over some time-period in the pre-stimulus interval) from the time-frequency power estimates. If you choose yes, SPM will ask you for the time-period over which you want to form the baseline. Next, SPM will ask you for the so-called "Morlet Wavelet factor". The default is 7. The greater this number, the less resolution you have over time, but the higher the resolution is in frequency. Then you can select the channels for which you want to compute the time-frequency decomposition. SPM will then ask you whether you also want to estimate the phase, in addition to power. The final question is whether you want to "collapse channels". If yes, SPM will average the estimates (within power and phase) over all selected channels.

12.8.9 Averaging: `spm_eeg_average`

Averaging of single trial data is the crucial step to obtain the evoked response. By default, when averaging single trial data, single trials are averaged within trial type. Power data of single trials (see sec. 12.8.8) can also be averaged by using the function `spm_eeg_average_TF`. The prepended output letter is *'m'*.

12.8.10 Contrast of trials: `spm_eeg_weight_epochs`

As an extension to the averaging functionality, SPM can also be used to compute linear combinations of single trials or evoked responses. For example, if you want to compute the difference between two evoked responses, you supply a contrast vector of $[-1; 1]$. Similarly, if you want to

remove some trials from the file, you can do this by using a contrast vector like [1; 0] which would write a new file with only the first evoked response. The prepended output letter is *'m'*.

The function will first ask you to "Enter contrasts". This is a matrix where each contrast is given by a row of this matrix. For example, if you compute just one contrast, you have to enter a vector of the same length as the number of trial types in the file. Note that SPM will zero-pad this vector (or matrix) if you specify fewer contrast weights than you have trials. The next question is whether you want to "Weight by num replications". This is important when you use this function on single trials, where, typically, you have a different number of trials for each trial type. If you then choose to average over multiple trials, this option allows you to choose whether you want to form an average that is weighted by the number of measurements within each trial type. As compared to an average, where you implicitly first form the averages within trial type, and then average with equal weighting.

12.9 Displaying data with SPM M/EEG Review

This tool can be called from the main SPM GUI under "Display" → M/EEG.

SPM M/EEG Review is meant to provide the user with basic visualization (data and source reconstruction) and reviewing (e.g. trial and sensor good/bad status) tools.

When called, SPM M/EEG Review displays in the SPM graphics window information about the SPM data file which is displayed (only for Matlab versions ≥ 7.4).

SPM M/EEG Review uses tabs to easily access different fields in the SPM data file structure (see relevant SPM manual section for SPM EEG data format). The main tabs system, at the top of the graphics windows, offers the following alternatives:

- **EEG** displays EEG type data (if any). These are the data associated with "EEG" sensors. The content of this tab is described below, as well as the "MEG" and "OTHER" tabs.
- **MEG** displays MEG type data (if any).
- **OTHER** displays any other type of data (e.g. HEOG, VEOG, etc).
- **info** (active tab by default): displays basic information about the data file. This tab contains three further sub-tabs[5]: "channels", "trials" and "inv" (the latter shows source reconstructions parameters, if any). Some of this info can be changed by the user (e.g. sensor/trial[6] type, label and status, etc) by editing the table. The changes become effective when clicking on "update". They are actually saved in the data file when clicking on "SAVE".
- **source** displays source reconstructions (if any). See below (2- source reconstructions visualization).

In addition, the user can call the SPM Prepare routine [7] or save any modification in the data file using the top-right buttons "Prepare SPM file" and "SAVE".

12.9.1 Data visualization

The graphics window of SPM Review offers two modes of data visualization: "scalp" and "standard" (default). For continuous (non-epoched) data, only "standard" mode is enabled. For time-frequency data, only "scalp" mode is enabled. For any other type of data, the user can switch to any of these modes using the standard/scalp radio button. These two modes are described below:

[5] Users can also check sensor coregistration when clicking on "sensor positions".

[6] Sensor/trial status (good/bad) can also be changed under the EEG/MEG/OTHER tabs, when visualizing trials (sensor: right-click uicontextmenu ; trials: button 10).

[7] This is part of the SPM EEG preprocessing tools. It mainly concerns the coregistration of the sensors onto the normalized SPM space. See relevant section in the SPM manual.

- **standard** channels are displayed vertically, within the same axes. A channel uicontextmenu can be accessed by right clicking on any time series (e.g. for changing the channel good/bad status). An additional axis (bottom right) provides the user with the temporal and horizontal scale of the displayed data). The size of the plotted time window can be changed using the top left buttons 1 and 2. User can scroll through the data using the temporal slider, at the bottom of the graphics window. A global display scaling factor can be changed using the top buttons 3 and 4. Zooming within the data is done by clicking on button 5. Clicking on button 6 displays a 2D scalp projection of the data.

  When displaying epoched data, the user can select the trial within the list of accessible trials (top right of the window). It is also possible to switch the status of trials (good/bad) by clicking on button 10.

  When displaying continuous data, SPM M/EEG REVIEW allows the user to manage events and selections. After having clicked on button 7, the user is asked to add a new event in the data file, by specifying its temporal bounds (two mouse clicks within the display axes). Basic properties of any events can be accessed either in the "info" table, or by right-clicking on the event marker (vertical line or patch superimposed on the displayed data). This gives access to the event uicontextmenu (e.g. for changing the event label). Buttons 8 and 9 allow the user to scroll through the data from marker to marker (backward and forward in time).

- **scalp** channels are displayed vertically, within the same axes. A channel uicontextmenu can be accessed by right clicking on any time series (e.g. for changing the channel good/bad status). An additional axis (bottom right) provides the user with the temporal and horizontal scale of the displayed data). The size of the plotted time window can be changed using the top left buttons 1 and 2. User can scroll through the data using the temporal slider, at the bottom of the graphics window. A global display scaling factor can be changed using the top buttons 3 and 4. Zooming within the data is done by clicking on button 5. Clicking on button 6 displays a 2D scalp projection of the data.

  When displaying epoched data, the user can select the trial within the list of accessible trials (top right of the window). It is also possible to switch the status of trials (good/bad) by clicking on button 10.

12.9.2 Source reconstructions visualization

SPM M/EEG REVIEW makes use of sub tabs for any source reconstruction that has been stored in the data file[8]. Since these reconstructions are associated with epoched data, the user can choose the trial he/she wants to display using the list of accessible events (top of the main tab). Each sub tab has a label given by the corresponding source reconstruction comment which is specified by the user when source reconstructing the data (see relevant section in the SPM manual). The bottom-left part of each sub tab displays basic infos about the source reconstruction (date, number of included dipoles, number of temporal modes, etc). The top part of the window displays a rendering of the reconstruction on the cortical surface that has been used. User can scroll through peri-stimulus time by using the temporal slider below the rendered surface. Other sliders allow the user to (i) change the transparency of the surface (left slider) and (ii) threshold the colormap (right sliders). In the center, a butterfly plot of the reconstructed intensity of cortical source activity over peri-stimulus time is displayed. If the data file contains more than one source reconstruction, the bottom-right part of the window displays a bar graph of the model evidences of each source reconstruction. This provides the user with a *visual* Bayesian model comparison tool[9]. SPM M/EEG REVIEW allows quick and easy switching between different models and trials, for a visual comparison of cortical source activities.

[8] This concerns any distributed source reconstruction, i.e. also includes imaging DCM analyses, but not ECD reconstructions (so far).

[9] Remember that model evidences $p(y|m)$ can only be compared for the same data. Therefore, if the source reconstructions have different time windows, filters, number of temporal modes, etc., the comparison does not hold. This is why basic information (bottom-left part of the window) has to be recalled when comparing models.

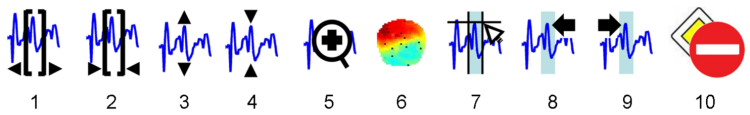


Figure 12.1: *SPM M/EEG* Review *buttons legend 1-2: increase/decrease width of plotted time window, 3-4: increase/decrease global scaling display factor, 5: zoom in, 7: add event, 8-9: scroll backward/forward data from marker to marker, 10: declare event as good/bad*

12.10 Batching and scripts

Although all functions can be called by the GUI, using only the GUI for preprocessing multi-subject data is a cumbersome and error-prone affair. For this reason, we included functions in SPM8 which should make batching jobs feasible, with only a small amount of Matlab knowledge required. How would one batch a series of preprocessing steps in SPM8? There are two ways of doing this, the new batch system of SPM8, and generating scripts.

12.10.1 The new SPM8 batch system

This method uses the new SPM8 batch system described in chapter 35. Using this batch system, you can put together a series of processing steps by choosing options from menus, in any order you like. When the full "job" is specified which may be a sequence of several processing step, one executes the job.

12.10.2 Script generation

Another way of batching jobs is by using scripts, written in Matlab . In SPM8, you can generate these scripts automatically. To do this, you first have to analyze one data set using the GUI or batch system. In SPM8, whenever a preprocessing function is called, all the input arguments, once they have been assembled by the GUI, are stored in a "history". This history can then be used to not only see in detail which functions have been used on a data set, but also to generate a script that repeats the same analysis steps. The big difference is that, this time, no more GUI interactions are necessary because the script already has all the input arguments which you gave during the first run. The history of an `meeg` object can be accessed by `D.history`.

To generate a script from the history of an SPM8 MEEG file, open the file in the M/EEG Review facility and select the `info` tab: a `history` tab is then available that will display all the history of the file. Clicking the Save as script button will ask for the filename of the Matlab script to save and the list of processing steps to save (default is all but it is possible to select only a subset of them). This will generate a script, which, when run, repeats the analysis. The script can also be obtained by directly calling the function `spm_eeg_history`.

Of course, this script can not only be used to repeat an analysis, but the script can also be seen as a template that can be re-used for other analyses. One needs minimal Matlab knowledge for these changes. For example, you can replace the filenames to preprocess a different subject. Or you can change parameters and then re-run the analysis. We have prepared an example, using the same example data set, as in the previous subsection to demonstrate this (see the file `man\example_scripts\history_subject1.m`). Note that these scripts can currently be used to do things that one couldn't do with the batch system. For example, if you want to exclude a channel from the analysis, there is no way to do this via the batch system. In the GUI, you would have to call display and switch the channel to 'bad'. With a script, you simply add a line like `D=badchannels(D, 23, 1)`, which flags channel 23 as bad (see also our example script after the filtering step). In summary, the idea is to preprocess a file through the GUI or batch system, then use the history-function to generate a template, and finally adapt this template to

modify the analysis in some way. To run the example script on your computer, you need the data set that you can download from the SPM webpage ([10]).

[10] http://www.fil.ion.ucl.ac.uk/spm/data/eeg_mmn/

Chapter 13

Analysis in sensor space

Second-level analyses for M/EEG are used to make inferences about the population from which subjects are drawn. This chapter describes how to perform second-level analyses of EEG/MEG or time-frequency (TF) data. This simply requires transforming data from `filename.mat` and `filename.dat` format to image files (NIfTI format). Once the data are in image format second-level analyses for M/EEG are procedurally identical to 2nd level analyses for fMRI. We therefor refer the reader to the fMRI section for further details of this last step.

To display the MEG/EEG/TF data the images are normalised to a standard size. Space (X,Y) dimensions are stretched to be displayed gracefully in SPM glass brain (note that it does not correspond to real `mm` coordinates). Time dimension is renormalised to be expressed as percent, from 0 to 100.

This means that the units are in a normalised space. Hopefully we aim at improving the visualisation in a near future.

In the GUI "other" pull down menu, select the function `Convert to images`. This will ask you to select the `filename.mat` of the data you would like to convert to images.

If you supply a result of a time-frequency decomposition you will first be asked to reduce your data from a 4D data (space(X,Y), time, frequency) to either a 3D image (space(X,Y), time) or a 2D time-frequency image (time, frequency). The prompt asks you over which dimension you wish to average your data.

If you select "electrodes" you will be asked to enter a vector of channels you wish to average over. Next you will be prompted for a "region number". This allows you to average over different electrodes/sensors to create different analyses for more than one region. The image will be created in a new directory with the name `<region number>ROI_TF_trialtype_<condition number>`. The image that will be created will be a 2D image where the dimensions are time and frequency. Before proceeding to the second-level analysis these images should be smoothed. This is performed using the SMOOTH function in the drop-down "Other" menu. Once the images have been smoothed one can proceed to second level analyses.

If you select "frequencies" you will be asked to specify the frequency range over which you wish to average. You will then be prompted to enter the output dimensions of the (interpolated scalp) image that will be produced. Typically we suggest using 64. You will then be asked whether you want to interpolate or remove bad channels from your images. If you chose interpolate then the image will interpolate missing channels. This is the preferred option. You will then be asked for the pixel dimensions. The default value is 3 and relates voxel-size to a mm-coordinate system used later on when displaying images in SPM. This should be changed to 1. This will create an image file with the file name appended by `F<frequency range>`. The image will have 3 dimensions (space(X,Y) and time). As with the TF data these images should be smoothed to accomodate spatial/temporal variability between subjects.

If when calling `Convert to images` you choose a non time-frequency data file, you will then be prompted to enter the output dimensions of the (interpolated scalp) image that will be produced. Typically we suggest using 64. You will then be asked whether you want to interpolate or remove bad channels from your images. If you choose interpolate then the image will interpolate missing channels. This is the preferred option. This will then create an image file in 3D where the dimensions are space (X,Y) and time. As before a separate image is created for each trial type. These images must be smoothed prior to second level analysis using the SMOOTH function in the drop down "Other" menu. Once the images have been smoothed one can proceed to the second level analysis.

Chapter 14

3D source reconstruction: Imaging approach

This chapter describes an Imaging approach to 3D source reconstruction.

14.1 Introduction

This chapter focuses on the imaging (or distributed) method for implementing EEG/MEG source reconstruction in SPM. This approach results in a spatial projection of sensor data into (3D) brain space and considers brain activity as comprising a very large number of dipolar sources spread over the cortical sheet, with fixed locations and orientations. This renders the observation model linear, the unknown variables being the source amplitudes or power.

Given epoched and preprocessed data (see chapter 12), the evoked and/or induced activity for each dipolar source can be estimated, for a single time-sample or a wider peristimulus time window.

The obtained reconstructed activity is in 3D voxel space and can be further analyzed using mass-univariate analysis in SPM.

Contrary to PET/fMRI data reconstruction, EEG/MEG source reconstruction is a non trivial operation. Often compared to estimating a body shape from its shadow, inferring brain activity from scalp data is mathematically ill-posed and requires prior information such as anatomical, functional or mathematical constraints to isolate a unique and most probable solution [11].

Distributed linear models have been around for more than a decade now [19] and the proposed pipeline in SPM for an imaging solution is classical and very similar to common approaches in the field. However, at least two aspects are quite original and should be emphasized here:

- Based on an empirical Bayesian formalism, the inversion is meant to be generic in the sense it can incorporate and estimate the relevance of multiple constraints of varied nature; data-driven relevance estimation being made possible through Bayesian model comparison [32, 66, 55, 28].
- The subject's specific anatomy is incorporated in the generative model of the data, in a fashion that eschews individual cortical surface extraction. The individual cortical mesh is obtained automatically from a canonical mesh in MNI space, providing a simple and efficient way of reporting results in stereotactic coordinates.

The EEG/MEG imaging pipeline is divided into four consecutive steps which characterize any inverse procedure with an additional step of summarizing the results. In this chapter, we go through each of the steps that need completing when proceeding with a full inverse analysis:

1. Source space modeling,
2. Data co-registration,
3. Forward computation,

4. Inverse reconstruction.

5. Summarizing the results of inverse reconstruction as an image.

Whereas the first three steps are part of the whole generative model, the inverse reconstruction step consists in Bayesian inversion, and is the only step involving actual EEG/MEG data.

14.2 Getting started

Everything which is described hereafter is accessible from the SPM user-interface by choosing the "EEG" application, `3D Source Reconstruction` button. When you press this button a new window will appear with a GUI that will guide you through the necessary steps to obtain an imaging reconstruction of your data. At each step, the buttons that are not yet relevant for this step will be disabled. When you open the window the only two buttons you can press are `Load` which enables you to load a pre-processed SPM MEEG dataset and the `Group inversion` button that will be described below. You can load a dataset which is either epoched with single trials for different conditions, averaged with one event related potential (ERP) per condition, or grand-averaged. An important pre-condition for loading a dataset is that it should contain sensors and fiducials. This will be checked when you load a file and loading will fail in case of a problem. You should make sure that for each modality present in the dataset as indicated by channel types (either EEG or MEG) there is a sensor description. If, for instance, you have an MEG dataset with some EEG channels that you don't actually want to use for source reconstruction, change their type to "*Other*" before trying to load the dataset. MEG datasets converted by SPM from their raw formats will always contain sensor and fiducial descriptions. In the case of EEG for some supported channel setups (such as extended 10-20 or BioSemi) SPM will provide default channel locations and fiducials that you can use for your reconstruction. Sensor and fiducial descriptions can be modified using the `Prepare` interface and in this interface you can also verify that these descriptions are sensible by performing a coregistration (see chapter 12 and also below for more details about coregistration).

When you successfully load a dataset you are asked to give a name to the present analysis cell. In SPM it is possible to perform multiple reconstructions of the same dataset with different parameters. The results of these reconstructions will be stored with the dataset if you press the `Save` button. They can be loaded and reviewed again using the 3D GUI and also with the SPM EEG Review tool. From the command line you can access source reconstruction results via the `D.inv` field of the `meeg` object. This field (if present) is a cell array of structures and does not require methods to access and modify it. Each cell contains the results of a different reconstruction. In the GUI you can navigate between these cells using the buttons in the second row. You can also create, delete and clear cells. The label you input at the beginning will be attached to the cell for you to identify it.

14.3 Source space modeling

After entering the label you will see the `Template` and `MRI` button enabled. The `MRI` button will create individual head meshes describing the boundaries of different head compartments based on the subject's structural scan. SPM will ask for the subject's structural image. It might take some time to prepare the model as the image needs to be segmented. The individual meshes are generated by applying the inverse of the deformation field needed to normalize the individual structural image to MNI template to canonical meshes derived from this template. This method is more robust than deriving the meshes from the structural image directly and can work even when the quality of the individual structural images is low.

Presently we recommend the `Template` button for both EEG and MEG. This button uses SPM's template head model based on the MNI brain. The corresponding structural image can be found under `canonical\single_subj_T1.nii` in the SPM directory. When you use the template, different things will happen depending on whether your data is EEG or MEG. For EEG, your electrode positions will be transformed to match the template head. So even if your subject's

head is quite different from the template, you should be able to get good results. For MEG, the template head will be transformed to match the fiducials and headshape that come with the MEG data. In this case having a headshape measurement can be quite helpful in providing SPM with more data to scale the head correctly. From the user's perspective the two options will look quite similar.

No matter whether the `MRI` or `Template` button was used the cortical mesh, which describes the locations of possible sources of EEG and MEG signal, is obtained from a template mesh. In the case of EEG the mesh is used as is, and in the case of MEG it is transformed with the head model. Three cortical mesh sizes are available "coarse", "normal" and "fine" (5124, 8196 and 20484 vertices respectively). It is advised to work with the "normal" mesh. Choose "coarse" if your computer has difficulties handling the "normal" option. "Fine" will only work on 64-bit systems and is probably an overkill.

14.4 Coregistration

In order for SPM to provide a meaningful interpretation of the results of source reconstruction, it should link the coordinate system in which sensor positions are originally represented to the coordinate system of a structural MRI image (MNI coordinates). In general, to link between two coordinate systems you will need a set of at least 3 points whose coordinates are known in both systems. This is a kind of *Rosetta stone* that can be used to convert a position of any point from one system to the other. These points are called "fiducials" and the process of providing SPM with all the necessary information to create the *Rosetta stone* for your data is called "coregistration".

There are two possible ways of coregistrating the EEG/MEG data into the structural MRI space.

1. A Landmark based coregistration (using fiducials only).
   The rigid transformation matrices (Rotation and Translation) are computed such that they match each fiducial in the EEG/MEG space into the corresponding one in sMRI space. The same transformation is then applied to the sensor positions.

2. Surface matching (between some headshape in MEG/EEG space and some sMRI derived scalp tesselation).
   For EEG, the sensor locations can be used instead of the headshape. For MEG, the headshape is first coregistrated into sMRI space; the inverse transformation is then applied to the head model and the mesh.
   Surface matching is performed using an Iterative Closest Point algorithm (ICP). The ICP algorithm [12] is an iterative alignment algorithm that works in three phases:

   - Establish correspondence between pairs of features in the two structures that are to be aligned based on proximity;
   - Estimate the rigid transformation that best maps the first member of the pair onto the second;
   - Apply that transformation to all features in the first structure. These three steps are then reapplied until convergence is concluded. Although simple, the algorithm works quite effectively when given a good initial estimate.

In practice what you will need to do after pressing the `Coregister` button is to specify the points in the sMRI image that correspond to your M/EEG fiducials. If you have more fiducials (which may happen for EEG as in principle any electrode can be used as a fiducial), you will be ask at the first step to select the fiducials you want to use. You can select more than 3, but not less. Then for each M/EEG fiducial you selected you will be asked to specify the corresponding position in the sMRI image in one of 3 ways.

- `select` - locations of some points such as the commonly used nasion and preauricular points and also CTF recommended fiducials for MEG (as used at the FIL) are hard-coded in SPM. If your fiducial corresponds to one of these points you can select this option and then select the correct point from a list.

- `type` - here you can enter the MNI coordinates for your fiducial (1×3 vector). If your fiducial is not on SPM's hard-coded list, it is advised to carefully find the right point on either the template image or on your subject's own image normalized to the template. You can do it by just opening the image using SPM's Display/images functionality. You can then record the MNI coordinates and use them in all coregistrations you need to do using the "type" option.

- `click` - here you will be presented with a structural image where you can click on the right point. This option is good for "quick and dirty" coregistration or to try out different options.

You will also have the option to skip the current fiducial, but remember you can only do it if you eventually specify more than 3 fiducials in total. Otherwise the coregistration will fail.

After you specify the fiducials you will be asked whether to use the headshape points if they are available. For EEG it is advised to always answer "yes". For MEG if you use a head model based on the subject's sMRI and have precise information about the 3 fiducials (for instance by doing a scan with fiducials marked by vitamin E capsules) using the headshape might actually do more harm than good. In other cases it will probably help, as in EEG.

The results of coregistration will be presented in SPM's graphics window. It is important to examine the results carefully before proceeding. In the top plot you will see the scalp, the inner skull and the cortical mesh with the sensors and the fiducials. For EEG make sure that the sensors are on the scalp surface. For MEG check that the head positon in relation to the sensors makes sense and the head does not for instance stick outside the sensor array. In the bottom plot the sensor labels will be shown in topographical array. Check that the top labels correspond to anterior sensors, bottom to posterior, left to left and right to right and also that the labels are where you would expect them to be topographically.

14.5 Forward computation (*forward*)

This refers to computing for each of the dipoles on the cortical mesh the effect it would have on the sensors. The result is a $N \times M$ matrix where N is the number of sensors and M is the number of mesh vertices (that you chose from several options at a previous step). This matrix can be quite big and it is, therefore, not stored in the header, but in a separate `*.mat` file which has `SPMgainmatrix` in its name and is written in the same directory as the dataset. Each column in this matrix is a so called "lead field" corresponding to one mesh vertex.

The lead fields are computed using the "forwinv" toolbox[1] developed by Robert Oostenveld, which SPM shares with FieldTrip. This computation is based on Maxwell's equations and makes assumptions about the physical properties of the head. There are different ways to specify these assumptions which are known as "forward models".

The "forwinv" toolbox can support different kinds of forward models. When you press `Forward Model` button (which should be enabled after successful coregistration), you will have a choice of several head models depending on the modality of your dataset. In SPM8 we recommend useing a single shell model for MEG and 3-shell BEM for EEG. You can also try other options and compare them using model evidence (see below). The first time you use the EEG BEM option with a new structural image (and also the first time you use the `Template` option) a lengthy computation will take place that prepares the BEM model based on the head meshes. The BEM will then be saved in a quite large `*.mat` file with ending `_EEG_BEM.mat` in the same directory with the structural image ("canonical" subdirectory of SPM for the template). When the head model is ready, it will be displayed in the graphics window with the cortical mesh and sensor locations you should verify for the final time that everything fits well together.

The actual lead field matrix will be computed at the beginning of the next step and saved. This is a time-consuming step and it takes longer for high-resolution meshes. The lead field file will be used for all subsequent inversion if you do not change the coregistration and the forward model.

[1] forwinv: http://www2.ru.nl/fcdonders/fieldtrip/doku.php?id=fieldtrip:development:forwinv

14.6 Inverse reconstruction

To get started press the `Invert` button. The first choice you will see is between `Classical`, `VB-ECD` and `DCM`. DCM, which is explained in detail elsewhere, can also be seen as an informed source reconstruction approach. It can, therefore, be invoked at this point. We will not go into further details here as you can find them in chapter 16. For reconstruction based on an empirical Bayesian approach to localize either the evoked response, the evoked power or the induced power, as measured by EEG or MEG press the `Classical` button.

If you have trials belonging to more than one condition in your dataset then the next choice you will have is whether to invert all the conditions together or to choose a subset. It is recommended to invert the conditions together if you are planning to later do a statistical comparison between them. If you have only one condition, or after choosing the conditions, you will get a choice between "Standard" and "Custom" inversion. If you choose "Standard" inversion, SPM will start the computation with default settings. These correspond to the multiple sparse priors (MSP) algorithm [26] which is then applied to the whole input data segment.

If you want to fine-tune the parameters of the inversion, choose the "Custom" option. You will then have the possibility to choose between several types of inversion differing by their hyperprior models (IID - equivalent to classical minimum norm, COH - smoothness prior similar to methods such as LORETA) or optimization scheme for the MSP method (GS - greedy search, ARD - automatic relevance determination).

You can then choose the time window that will be available for inversion. Generally, it is not recommended to limit the time segment you use for inversion just to what you are interested in, unless there are stimulus artefacts that need to be excluded. It is better to let the algorithm model all the brain sources generating the response and then to focus on the sources of interest using the appropriate contrast (see below). There is also an option to apply a hanning taper to the channel time series in order to downweight the possible baseline noise at the beginning and end of the trial. There is also an option to pre-filter the data. Finally, you can restrict solutions to particular brain areas by loading a `*.mat` file with a $K \times 3$ matrix containing MNI coordinates of the areas of interest. This option may initially seem strange, as it may seem to overly bias the source reconstructions returned. However, in the Bayesian inversion framework you can compare different inversions of the same data using Bayesian model comparison. By limiting the solutions to particular brain areas you greatly simplify your model and if that simplification really captures the sources generating the response, then the restricted model will have much higher model evidence than the unrestricted one. If, however, the sources you suggested cannot account for the data, the restriction will result in a worse model fit and depending on how much worse it is, the unrestricted model might be better in the comparison. So using this option with subsequent model comparison is a way, for instance, to integrate prior knowledge from the literature or from fMRI/PET/DTI into your inversion. It also allows for comparison of alternative prior models.

Note that for model comparison to be valid all the settings that affect the input data, like the time window, conditions used and filtering should be identical.

SPM8 imaging source reconstruction also supports multi-modal datasets. These are datasets that have both EEG and MEG data from a simultaneous recording. Datasets from the "Neuromag" MEG system which has two kinds of MEG sensors are also treated as multimodal. If your dataset is multimodal and additional question will appear asking whether to select one of the modalities or "fuse" them. The "fuse" option based on the paper by Henson et al. (submitted) use a heuristic to rescale the data from different modalities so that they can be used together. Presently fusion cannot be used with "group inversion" (see below) but this possibility will be available in the future.

Once the inversion is completed you will see the time course of the region with maximal activity in the top plot of the graphics window. The bottom plot will show the maximal intensity projection (MIP) at the time of the maximal activation. You will also see the log-evidence value that can be used for model comparison, as explained above. Note that not all the output of the inversion is displayed. The full output consists of time courses for all the sources and conditions for the entire time window. You can view more of the results using the controls in the bottom right corner of the 3D GUI. These allow focusing on a particular time, brain area and condition. One can also display a movie of the evolution of neuronal activity.

14.7 Summarizing the results of inverse reconstruction as an image

SPM offers the possibility of writing the results as 3D NIfTI images, so that you can then proceed with GLM-based statistical analysis using Random Field theory. This is similar to the 2nd level analysis in fMRI for making inferences about region and trial-specific effects (at the between subject level).

This entails summarizing the trial- and subject-specific responses with a single 3-D image in source space. Critically this involves prompting for a time-frequency contrast window to create each contrast image. This is a flexible and generic way of specifying the data feature you want to make an inference about (e.g., gamma activity around 300 ms or average response between 80 and 120 ms). This kind of contrast is specified by pressing the `Window` button. You will then be asked about the time window of interest (in ms, peri-stimulus time). It is possible to specify either a single time value or a time segment. The next question is about the frequency band. If you just want to average the source time course leave that at the default, zero. In this case the window will be weighted by a Gaussian. In the case of a single time point this will be a Gaussian with 8 ms full width half maximum (FWHM). If you specify a particular frequency or a frequency band, then a series of Morlet wavelet projectors will be generated summarizing the energy in the time window and band of interest.

There is a difference between specifying a frequency band of interest as zero, as opposed to specifying a wide band that covers the whole frequency range of your data. In the former case the time course of each dipole will be averaged, weighted by a gaussian. Therefore, if within your time window this time course changes polarity, the activity can average out and in an ideal case even a strong response can produce a value of zero. In the latter case the power is integrated over the whole spectrum ignoring phase, and this would be equivalent to computing the sum of squared amplitudes in the time domain.

Finally, you will have a choice between "evoked" and "induced". Here one can specify an epoched rather than an averaged file for the inversion. If you have multiple trials for certain conditions, the projectors generated at the previous step can either be applied to each trial and the results averaged (induced) or applied to the averaged trials (evoked). Thus it is possible to perform localization of induced activity that has no phase-locking to the stimulus. It is also possible to focus on frequency content of the ERP using the "evoked" option. Clearly the results will not be the same. The projectors you specified (bottom plot) and the resulting MIP (top plot) will be displayed when the operation is completed.

At this stage you have the possibility to export your image as NIfTI by pressing the "Image" button. You will be asked to specify the amount of smoothing. An unsmoothed image will be exported in any case. Smoothing can help to get better results in a between-subject statistical test. However, it compromises your spatial resolution. You should try different options to see what works for your data. When you use the group inversion (see below) you may not need smoothing at all.

Note that when a file contains several conditions, the images are normalized so that sum of power across all conditions is 1. Thus you might get differently scaled images if you invert each condition alone as opposed to inverting them together. It is recommended to put all the conditions you intend to compare in the same inversion.

14.8 Rendering interface

By pressing the `Render` button you can open a new GUI window which will show you a rendering of the inversion results on the brain surface. You can rotate the brain, focus on different time points, run a movie and compare the predicted and observed scalp topographies and time series. A useful option is "virtual electrode" which allows you to extract the time course from any point on the mesh and the MIP at the time of maximal activation at this point. Just press the button and click anywhere in the brain.
An additional tool for reviewing the results is available in the SPM M/EEG Review function.

14.9 Group inversion

A problem encountered with MSP inversion is that sometimes it is "too good", producing solutions that were so focal in each subject that the spatial overlap between the activated areas across subjects was not sufficient to yield a significant result in a between-subjects contrast. This could be improved by smoothing, but smoothing compromises the spatial resolution and thus subverts the main advantage of using an inversion method that can produce focal solutions.

To circumvent this problem we proposed a modification of the MSP method [52] that effectively restricts the activated sources to be the same in all subjects with only the degree of activation allowed to vary. We showed that this modification makes it possible to obtain significance levels close to those of non-focal methods such as minimum norm while preserving accurate spatial localization.

The group inversion can yield much better results than individual inversions because it introduces an additional constraint for the ill-posed inverse problem, namely that the responses in all subjects should be explained by the same set of sources. Thus it should be your method of choice when analyzing an entire study with subsequent GLM analysis of the images.

Group inversion works very similarly to what was described above. You can start it by pressing the "Group inversion" button right after opening the 3D GUI. You will be asked to specify a list of M/EEG data sets to invert together. Then the routine will ask you to perform coregistration for each of the files and specify all the inversion parameters in advance. It is also possible to specify the contrast parameters in advance. Then the inversion will proceed by computing the inverse solution for all the files and will write out the output images. The results for each subject will also be saved in the header of the corresponding input file. It is possible to load this file into the 3D GUI after the inversion and explore the results as described above.

14.10 Appendix: Data structure

The MATLAB object describing a given EEG/MEG dataset in SPM is denoted as *D*. Within that structure, each new inverse analysis will be described by a new cell of sub-structure field *D.inv* and will be made of the following fields:

- `method`: character string indicating the method, either "ECD" or "Imaging" in present case;
- `mesh`: sub-structure with relevant variables and filenames for source space and head modeling;
- `datareg`: sub-structure with relevant variables and filenames for EEG/MEG data registration into MRI space;
- `forward`: sub-structure with relevant variables and filenames for forward computation;
- `inverse`: sub-structure with relevant variable, filenames as well as results files;
- `comment`: character string provided by the user to characterize the present analysis;
- `date`: date of the last modification made to this analysis.

Chapter 15

Localization of Equivalent Current Dipoles

This chapter describes source reconstruction based on "Variational Bayes Equivalent Current Dipoles" (VB-ECDs). For more details about the implementation, please refer to the help and comments in the routines themselves, as well as the original paper by [47].

15.1 Introduction

3D imaging (or distributed) reconstruction methods consider all possible source location simultaneously, allowing for large and widely spread clusters of activity. This is to be contrasted with "Equivalent Current Dipole (ECD) approaches which rely on two different hypotheses:

- only a few (say less than 5) sources are active simultaneously, and
- those sources are very focal.

This leads to the ECD model where the observed scalp potential will be explained by a handful of discrete current sources, i.e. dipoles, located inside the brain volume.

In contrast to the 3D imaging reconstruction, the number of ECDs considered in the model, i.e. the number of "active locations, should be defined a priori. This is a crucial step, as the number of sources considered defines the ECD model. This choice should be based on empirical knowledge of the brain activity observed or any other source of information (for example by looking at the scalp potential distribution). In general, each dipole is described by 6 parameters: 3 for its location, 2 for its orientation and 1 for its amplitude. Once the number of ECDs is fixed, a non-linear optimisation algorithm is used to adjust the dipoles parameters (6 times the number of dipoles) to the observed potential.

Classical ECD approaches use a simple best fitting optimisation using least square error criteria. This leads to relatively simple algorithms but presents a few drawbacks:

- constraints on the dipoles are difficult to include in the framework;
- the noise cannot be properly taken into account, as its variance should be estimated alongside the dipole parameters;
- it is difficult to define confidence intervals on the estimated parameters, which could lead to over-confident interpretation of the results;
- models with different numbers of dipoles cannot be compared except through their goodness-of-fit, which can be misleading.

As adding dipoles to a model will necessarily improve the overall goodness of fit, one could erroneously be tempted to use as many ECDs as possible and to perfectly fit the observed signal. Through using Bayesian techniques, however, it is possible to circumvent all of the above limitations of classical approaches.

Briefly, a probabilistic generative model is built providing a likelihood model for the data[1]. The model is completed by a set of priors on the various parameters, leading to a Bayesian model, allowing the inclusion of user-specified prior constraints.

A "variational Bayes (VB) scheme is then employed to estimate the posterior distribution of the parameters through an iterative procedure. The confidence interval of the estimated parameters is therefore directly available through the estimated posterior variance of the parameters. Critically, in a Bayesian context, different models can be compared using their evidence or marginal likelihood. This model comparison is superior to classical goodness-of-fit measures, because it takes into account the complexity of the models (e.g., the number of dipoles) and, implicitly, uncertainty about the model parameters. VB-ECD can therefore provide an objective and accurate answer to the question: Would this data set be better modelled by 2 or 3 ECDs?

15.2 Procedure in SPM8

This section aims at describing how to use the VB-ECD approach in SPM8.

15.2.1 Head and forward model

The engine calculating the projection of the dipolar sources on the scalp electrode comes from Fieldtrip and is the same for the 3D imaging or VB-ECD reconstructions. The head model should thus be prepared the same way, as described in the chapter 14. For the same data set, differences between the VB-ECD and imaging reconstructions would therefore be due to the reconstruction approach only.

15.2.2 VB-ECD reconstruction

To get started, after loading and preparing the head model, press the 'Invert' button[2]. The first choice you will see is between 'Classical', 'VB-ECD' and 'DCM'. The 'Classical' reconstruction corresponds to the imaging solution, as described in chapter 14, and 'DCM' is described in chapter 16. Then you are invited to fill in information about the ECD model and click on buttons in the following order:

1. fill in comments about the current reconstruction.
2. indicate the time bin or time window for the reconstruction, within the epoch length. Note that the data will be averaged over the selected time window! VB-ECD will thus always be calculated for a single time bin.
3. enter the trial type(s) to be reconstructed. Each trial type will be reconstructed separately.
4. add a single (i.e. individual) dipole or a pair of symmetric dipoles to the model. Each "element (single or pair) is added individually to the model.
5. use "Informative or 'Non-informative location priors. "Non-informative means flat priors over the brain volume. With "Informative, you can enter the a priori location of the source[3].
6. use "Informative or 'Non-informative moment priors. "Non-informative means flat priors over all possible directions and amplitude. With "Informative, you can enter the a priori moment of the source[4].
7. if a pair of dipoles was added, you can then choose between "soft or "hard symmetric priors. A "hard prior enforces perfectly symmetric dipoles, i.e. there is a parameter reduction. A "soft prior imposes some (strong) correlation between the dipoles but lets them differ a little in location and moment.

[1] This includes an independent and identically distributed (IID) Normal distribution for the errors, but other distributions could be specified.

[2] The GUI for VB-ECD can also be launched directly from Matlab command line with the instruction: $D = spm_eeg_inv_vbecd_gui$.

[3] For a pair of dipoles, only the right dipole coordinates are required.

[4] For a pair of dipoles, only the right dipole moment is required.

8. go back to step 4 and add some more dipole(s) to the model, or stop adding dipoles.

The routine then proceeds with the VB optimization scheme to estimate the model parameters. Results are finally saved into the data structure D in the field $.inv\{D.val\}.inverse$ and displayed in the graphic window.

15.2.3 Result display

The latest VB-ECD results can be displayed again through the function $D = spm_eeg_inv_vbecd_disp$. If a specific reconstruction should be displayed, then use: *spm_eeg_inv_vbecd_disp('Init',D, ind)*. In the upper part, the 3 main figures display the 3 orthogonal views of the brain with the dipole location and orientation superimposed. The location confidence interval is described by the dotted ellipse around the dipole location on the 3 views. It is not possible to click through the image, as the display is automatically centred on the dipole displayed. It is possible though to zoom into the image, using the right-click context menu.

The lower left table displays the current dipole location, orientation (Cartesian or polar coordinates) and amplitude in various formats.

The lower right table allows for the selection of trial types and dipoles. Display of multiple trial types and multiple dipoles is also possible. The display will center itself on the average location of the dipoles.

Chapter 16

Dynamic Causal Modelling for M/EEG

16.1 Introduction

Dynamic Causal Modelling (DCM) is based on an idea initially developed for fMRI data: The measured data are explained by a network model consisting of a few sources, which are interacting dynamically. This network model is inverted using a Bayesian approach, and one can make inferences about connections between sources, or the modulation of connections by task.

For M/EEG data, DCM is a powerful technique for inferring about parameters that one doesn't observe with M/EEG directly. Instead of asking 'How does the strength of the source in left superior temporal gyrus (STG) change between condition A and B?', one can ask questions like 'How does the backward connection from this left STG source to left primary auditory cortex change between condition A and B?'. In other words, one isn't limited to questions about source strength as estimated using a source reconstruction approach, but can test hypotheses about what is happening between sources, in a network.

As M/EEG data is highly resolved in time, as compared to fMRI, the inferences are about more neurobiologically plausible parameters. These relate more directly to the causes of the underlying neuronal dynamics.

The key DCM for M/EEG methods paper appeared in 2006, and the first DCM studies about mismatch negativity came out in 2007/2008. At its heart DCM for M/EEG is a source reconstruction technique, and for the spatial domain we use exactly the same leadfields as other approaches. However, what makes DCM unique, is that is combines the spatial forward model with a biologically informed temporal forward model, describing e.g. the connectivity between sources. This critical ingredient not only makes the source reconstruction more robust by implicitly constraining the spatial parameters, but also allows one to infer about connectivity.

Our methods group is continuing to work on further improvements and extensions to DCM. At the same time, we take great care that the core functionality implemented in SPM8 is robust and usable. In the following, we will describe the usage of DCM for evoked responses (both MEG and EEG), DCM for induced responses (i.e., based on power data in the time-frequency domain), and DCM for local field potentials (measured as steady-state responses). All three DCMs share the same interface, as many of the parameters that need to be specified are the same for all three approaches. Therefore, we will first describe DCM for evoked responses, and then point out where the differences to the other two DCMs lie.

This manual provides only a procedural guide for the practical use of DCM for M/EEG. If you want to read more about the scientific background, the algorithms used, or how one would typically use DCM in applications, we recommend the following reading. The two key methods contributions can be found in [20] and [48]. Two other contributions using the model for testing interesting hypotheses about neuronal dynamics are described in [49] and [22]. At the time of writing, there were also three application papers published which demonstrate what kind of hy-

potheses can be tested with DCM [35, 34, 33]. Another good source of background information is the recent SPM book [23], where Parts 6 and 7 cover not only DCM for M/EEG but also related research from our group. The DCMs for induced responses and steady-state responses are covered in [16] and [57, 58]. Also note that there is a DCM example file, which we put onto the webpage http://www.fil.ion.ucl.ac.uk/spm/data/eeg_mmn/. After downloading DCMexample.mat, you can load (see below) this file using the DCM GUI, and have a look at the various options, or change some, after reading the description below.

16.2 Overview

In summary, the goal of DCM is to explain measured data (such as evoked responses) as the output of an interacting network consisting of a several areas, some of which receive input (i.e., the stimulus). The differences between evoked responses, measured under different conditions, are modelled as a modulation of selected DCM parameters, e.g. cortico-cortical connections [20]. This interpretation of the evoked response makes hypotheses about connectivity directly testable. For example, one can ask, whether the difference between two evoked responses can be explained by top-down modulation of early areas [35]. Importantly, because model inversion is implemented using a Bayesian approach, one can also compute Bayesian model evidences. These can be used to compare alternative, equally plausible, models and decide which is the best [50].

DCM for evoked responses takes the spatial forward model into account. This makes DCM a spatiotemporal model of the full data set (over channels and peri-stimulus time). Alternatively, one can describe DCM also as a spatiotemporal source reconstruction algorithm which uses additional temporal constraints given by neural mass dynamics and long-range effective connectivity. This is achieved by parameterising the lead-field, i.e., the spatial projection of source activity to the sensors. In the current version, this can be done using two different approaches. The first assumes that the leadfield of each source is modelled by a single equivalent current dipole (ECD) [48]. The second approach posits that each source can be presented as a 'patch' of dipoles on the grey matter sheet (Daunizeau et al., in preparation). This spatial model is complemented by a model of the temporal dynamics of each source. Importantly, these dynamics not only describe how the intrinsic source dynamics evolve over time, but also how a source reacts to external input, coming either from subcortical areas (stimulus), or from other cortical sources.

The GUI allows one to enter all the information necessary for specifying a spatiotemporal model for a given data set. If you want to fit multiple models, we recommend using a batch script. An example of such a script (*DCM_ERP_example*), which can be adapted to your own data, can be found in the *man/example_scripts/* folder of the distribution. You can run this script on example data provided by via the SPM webpage (http://www.fil.ion.ucl.ac.uk/spm/data/eeg_mmn/). However, you first have to preprocess these data to produce an evoked response by going through the preprocessing tutorial (chapter 33) or by running the `history_subject1.m` script in the `example_scripts` folder.

16.3 Calling DCM for ERP/ERF

After calling *spm eeg*, you see SPM's graphical user interface, the top-left window. The button for calling the DCM-GUI is found in the second partition from the top, on the right hand side. When pressing the button, the GUI pops up. The GUI is partitioned into five parts, going from the top to the bottom. The first part is about loading and saving existing DCMs, and selecting the type of model. The second part is about selecting data, the third is for specification of the spatial forward model, the fourth is for specifying connectivity, and the last row of buttons allows you to estimate parameters and view results.

You have to select the data first and specify the model in a fixed order (data selection > spatial model > connectivity model). This order is necessary, because there are dependencies among the three parts that would be hard to resolve if the input could be entered in any order. At any time, you can switch back and forth from one part to the next. Also, within each part, you can specify information in any order you like.

16.4 load, save, select model type

At the top of the GUI, you can load an existing DCM or save the one you are currently working on. In general, you can *save* and *load* during model specification at any time. You can also switch between different DCM analyses (the left menu). The default is 'ERP' which is DCM for evoked responses described here. Currently, the other two types are steady-state responses (SSR) and induced responses (IND); both are also described in this manual. The menu on the right-hand side lets you choose the neuronal model. Currently, there are four model types. The first is 'ERP', which is the standard model described in most of our papers, e.g. [20]. The second is 'SEP', which uses a variant of this model, however, the dynamics tend to be faster [54]. The third is 'NMM', which is a nonlinear neural mass model based on a first-order approximation, and the fourth is 'MFM', which is also nonlinear and is based on a second-order approximation. The latter two options haven't been described in the literature yet, but are under review.

16.5 Data and design

In this part, you select the data and model between-trial effects. The data can be either event-related potentials or fields. These data must be in the SPM-format. On the right-hand side you can enter trial indices of the evoked responses in this SPM-file. For example, if you want to model the second and third evoked response contained within an SPM-file, specify indices 2 and 3. If the two evoked responses, for some reason, are in different files, you have to merge these files first. You can do this with the SPM preprocessing function *merge* (*spm_eeg_merge*), see chapter 12. You can also choose how you want to model multiple evoked responses. When you enter multiple evoked responses, the default is to model the modulations of connectivity parameters independently of each other. You can change this default to model the way connections are modelled over trials. For example, when you want to model three evoked responses, you can model the modulations of a connection strength of the second and third evoked responses as two uncoupled multiplicative gains on the connection strength of the first evoked response. However, you can also choose to couple the connection strength of the first evoked response with the two gains by imposing a linear relationship on how this connection changes over trials. This can be useful when one wants to add constraints on how connections (or other DCM parameters) change over trials. A compelling example of this can be found in [33].

When you hit enter after entering the row vector of trial indices, a file requester will ask you for the name of the SPM-file. Alternatively, you can press the button 'data file'. Under 'time window (ms)' you have to enter the peri-stimulus times which you want to model, e.g. 1 to 200 ms.

You can choose whether you want to model the mean or drifts of the data at sensor level. If you don't, select 0 for 'detrend', otherwise select the number of discrete cosine transform terms you want to use to model the mean (1) or low-frequency drifts (> 1). In DCM, we use a projection of the data to a subspace to reduce the amount of data. The type of spatial projection is described in [22]. You can select the number of modes you wish to keep. The default is 8.

You can also choose to window your data, along peri-stimulus time, with a hanning window (radio button). This windowing will reduce the influence of the beginning and end of the time-series.

If you are happy with your data selection, the projection and the detrending terms, you can click on the $>$ (forward) button, which will bring you to the next stage *electromagnetic model*. From this part, you can press the red $<$ button to get back to the data and design part.

16.6 Electromagnetic model

With the present version of DCM, you have three options for how to spatially model your evoked responses. Either you use a single equivalent current dipole (ECD) for each source, or you use a patch on the cortical surface (IMG), or you don't use a spatial model at all (local field potentials (LFP)). The second option is not described in the peer-reviewed literature yet (Daunizeau et al., in preparation). In all three cases, you have to enter the source names (one name in one row). For ECD and IMG, you have to specify the prior source locations (in mm in MNI coordinates).

Note that by default DCM uses uninformative priors on dipole orientations, but tight priors on locations. This is because tight priors on locations ensure that the posterior location will not deviate to much from its prior location. This means each dipole stays in its designated area and retains its meaning. The prior location for each dipole can be found either by using available anatomical knowledge or by relying on source reconstructions of comparable studies. Also note that the prior location doesn't need to be overly exact, because the spatial resolution of M/EEG is on a scale of several millimeters. You can also load the prior locations from a file ('load'). You can visualize the locations of all sources when you press 'dipoles'.

The onset-parameter determines when the stimulus, presented at 0ms peri-stimulus time, is assumed to cause an impulse. In DCM, we usually do not model the rather small early responses, but start modelling at the first large deflection. Because the propagation of the stimulus impulse through the input nodes causes a delay, we found that the default value of 60 ms onset time is a good value for many evoked responses where the first large deflection is seen around 100 ms. However, this value is a prior, i.e., the inversion routine can adjust it. The prior mean should be chosen according to the specific responses of interest. This is because the time until the first large deflection is dependent on the paradigm or the modality you are working in, e.g. audition or vision. You may also find that changing the onset prior has an effect on how your data are fitted. This is because the onset time has strongly nonlinear effects (a delay) on the data, which might cause differences in which maximum was found at convergence, for different prior values.

When you want to proceed to the next model specification stage, hit the > (forward) button and proceed to the *neuronal model*.

16.7 Neuronal model

There are five (or more) matrices which you need to specify by button presses. The first three are the connection strength parameters for the first evoked response. There are three types of connections, *forward*, *backward* and *lateral*. In each of these matrices you specify a connection *from* a source area *to* a target area. For example, switching on the element (2, 1) in the intrinsic forward connectivity matrix means that you specify a forward connection from area 1 to 2. Some people find the meaning of each element slightly counter-intuitive, because the column index corresponds to the source area, and the row index to the target area.

In the present GUI implementation, there is only one input allowed. This input can go to any area and to multiple areas. You can select the receiving areas by selecting area indices in the C *input* vector.

The B matrix contains all gain modulations of connection strengths as set in the A-matrices. These modulations model the difference between the first and the other modelled evoked responses. For example, for two evoked responses, DCM explains the first response by using the A-matrix only. The 2nd response is modelled by modulating these connections by the weights in the B-matrix.

16.8 Estimation

When you are finished with model specification, you can hit the *estimate* button in the lower left corner. If this is the first estimation and you have not tried any other source reconstructions with this file, DCM will build a spatial forward model. You can use the template head model for quick results. For this, you answer 'no' to the question 'Redefine MRI fiducials?', and 'yes' to 'Use headshape points?' DCM will now estimate model parameters. You can follow the estimation process by observing the model fit in the output window. In the matlab command window, you will see each iteration printed out with expected-maximization iteration number, free energy F, and the predicted and actual change of F following each iteration step. At convergence, DCM saves the results in a DCM file, by default named 'DCM_ERP.mat'. You can save to a different name, eg. if you are estimating multiple models, by pressing 'save' at the top of the GUI and writing to a different name.

16.9 Results

After estimation is finished, you can assess the results by choosing from the pull-down menu at the bottom (middle).

With *ERPs (mode)* you can plot, for each mode, the data for both evoked responses, and the model fit.

When you select *ERPs (sources)*, the dynamics of each area are plotted. The activity of the pyramidal cells (which is the reconstructed source activity) are plotted in solid lines, and the activity of the two interneuron populations are plotted as dotted lines.

The option *coupling (A)* will take you to a summary about the posterior distributions of the connections in the A-matrix. In the upper row, you see the posterior means for all intrinsic connectivities. As above, element $(i,\ j)$ corresponds to a connection from area j to i. In the lower row, you'll find, for each connection, the probability that its posterior mean is different from the prior mean, taking into account the posterior variance.

With the option *coupling(B)* you can access the posterior means for the gain modulations of the intrinsic connectivities and the probability that they are unequal to the prior means.

With *coupling(C)* you see a summary of the posterior distribution for the strength of the input into the input receiving area. On the left hand side, DCM plots the posterior means for each area. On the right hand side, you can see the corresponding probabilities.

The option *Input* shows you the estimated input function. As described by [20], this is a gamma function with the addition of low-frequency terms.

With *Response*, you can plot the selected data, i.e. the data, selected by the spatial modes, but back-projected into sensor space.

With *Response (image)*, you see the same as under Results but plotted as an image in grey-scale.

And finally, with the option *Dipoles*, DCM displays an overlay of each dipole on an MRI template using the posterior means of its 3 orientation and 3 location parameters. This makes sense only if you have selected an ECD model under *electromagnetic model*.

Before estimation, when you press the button 'Initialise' you can assign parameter values as initial starting points for the free-energy gradient ascent scheme. These values are taken from another already estimated DCM, which you have to select.

The button *BMC* allows you do Bayesian model comparison of multiple models. After selecting DCMs, you will see, at the top, a bar plot of the log-model evidences for all models. At the bottom, you will see the probability, for each model, that it produced the data. By convention, a model can be said to be the best among a selection of other models, with strong evidence, if its log-model evidence exceeds all other log-model evidences by at least 3.

16.10 Steady-State Responses

16.10.1 Model specification

DCM for steady state responses can be applied to M/EEG or intracranial data.

The data must first be prepared using the temporal pre-processing option; 'other', then 'prep'. Here you specify the data type (MEG, EEG or LFP), and select the channels for a DCM analysis. In general you select all MEG/EEG channels and all or a subgroup of LFP channels from the areas you are interested in. A sample script to convert raw txt data, called textitspm_lfp_txt2mat can be found in the folder toolbox\Neural_Models.

The top panel of the DCM for ERP window allows you to toggle through available analysis methods. On the top left drop-down menu, select 'SSR'. The second drop-down menu in the right of the top-panel allows you to specify whether the analysis should be performed using a model which is linear in the states, for this you can choose ERP. Alternatively you may use a conductance based model, which is non-linear in the states by choosing, 'NMM' or 'MFM'. (see [54] for a description of the differences).

The steady state (frequency) response is generated automatically from the time domain recordings. The time duration of the frequency response is entered in the second panel in the time-window. The options for detrending allow you to remove either 1st, 2nd, 3rd or 4th order

polynomial drifts from channel data. In the subsampling option you may choose to downsample the data before constructing the frequency response. The number of modes specifies how many components from the leadfield are present in channel data. The specification of between trial effects and design matrix entry is the same as for the case of ERPs, described above.

16.10.2 The Lead-Field

The cross-spectral density is a description of the dependencies among the observed outputs of these neuronal sources. To achieve this frequency domain description we must first specify the likely sources and their location. If LFP data are used then only source names are required. This information is added in the third panel by selecting 'LFP'. Alternatively, x,y,z coordinates are specified for ECD or IMG solutions.

16.10.3 Connections

The bottom panel then allows you to specify the connections between sources and whether these sources can change from trial type to trial type.

On the first row, three connection types may be specified between the areas. For NMM and MFM options these are Excitatory, Inhibitory or Mixed excitatory and inhibitory connections. When using the ERP option the user will specify if connections are 'Forward', 'Backward' or 'Lateral'. To specify a connection, switch on the particular connection matrix entry. For example to specify an Inhibitory connection from source 3 to source 1, turn on the 'Inhib' entry at position (3,1).

On this row the inputs are also specified. These are where external experimental inputs enter the network.

The matrix on the next row allows the user to select which of the connections specified above can change across trial types. For example in a network of two sources with two mixed connections (1,2) and (2,1), you may wish to allow only one of these to change depending on experimental context. In this case, if you wanted the mixed connection from source 2 to source 1 to change depending on trial type, then select entry (2,1) in this final connection matrix.

16.10.4 Cross Spectral Densities

The final selection concerns what frequencies you wish to model. These could be part of a broad frequency range e.g. like the default 4 - 48 Hz, or you could enter a narrow band e.g. 8 to 12 Hz, will model the alpha band in 1Hz increments.

Once you hit the 'invert DCM' option the cross spectral densities are computed automatically (using the spectral-toolbox). The data for inversion includes the auto-spectra and cross-spectra between channels or between channel modes. This is computed using a multivariate autoregressive model, which can accurately measure periodicities in the time-domain data. Overall the spectra are then presented as an upper-triangular, s x s matrix, with auto-spectra on the main diagonal and cross-spectra in the off-diagonal terms.

16.10.5 Output and Results

The results menu provides several data estimates. By examining the 'data', you will be able to see observed spectra in the matrix format described above. Selecting 'Cross-spectral density' gives both observed and predicted responses. To examine the connectivity estimates you can select the 'coupling (A)' results option, or for the modulatory parameters, the 'coupling (B)' option. Also you can examine the input strength at each source by selecting the 'coupling (C)' option, as in DCM for ERPs. The option 'trial-specific effects' shows the change in connectivity parameter estimates (from B) from trial to trial relative to the baseline connection (from A). To examine the spectral input to these sources choose the 'Input' option; this should look like a mixture of white and pink noise. Finally the 'dipoles' option allows visualisation of the a posteriori position and orientation of all dipoles in your model.

16.11 Induced responses

DCM for induced responses aims to model coupling within and between frequencies that are associated with linear and non-linear mechanisms respectively. The procedure to do this is similar to that for DCM for ERP/ERF. In the following, we will just point out the differences in how to specify models in the GUI. Before using the technique, we recommend reading about the principles behind DCM for induced responses [16].

16.11.1 Data

The data to be modelled must be single trial, epoched data. We will model the entire spectra, including both the evoked (phase-locked to the stimulus) and induced (non-phase-locked to the stimulus) components.

16.11.2 Electromagnetic model

Currently, DCM for induced responses uses only the ECD method to capture the data features. Note that a difference to DCM for evoked responses is that the parameters of the spatial model are not optimized. This means that DCM for induced responses will project the data into source space using the spatial locations provided by you.

16.11.3 Neuronal model

This is where you specify the connection architecture. Note that in DCM for induced responses, the A-matrix encodes the linear and nonlinear coupling strength between sources.

16.11.4 Wavelet transform

This function can be called below the connectivity buttons and allows one to transfer data into the time-frequency domain using a Morlet Wavelet transform as part of the feature extraction. There are two parameters: The frequency window defines the desired frequency band and the wavelet number specifies the temporal-frequency resolution. We recommend values greater than 5 to obtain a stable estimation.

16.11.5 Results

Frequency modes

This will display the frequency modes, identified using singular value decomposition of spectral dynamics in source space (over time and sources).

Time-Frequency

This will display the observed time-frequency power data for all pre-specified sources (upper panel) and the fitted data (lower panel).

Coupling (A-Hz)

This will display the coupling matrices representing the coupling strength from source to target frequencies.

16.12 Phase-coupled responses

DCM for phase-coupled responses is based on a weakly coupled oscillator model of neuronal interactions.

16.12.1 Data

The data to be modeled must be multiple trial, epoched data. Multiple trials are required so that the full state-space of phase differences can be explored. This is achieved with multiple trials as each trial is likely to contain different initial relative phase offsets. Information about different trial types is entered as it is with DCM for ERP ie. using a design matrix. DCM for phase coupling is intended to model dynamic transitions toward synchronization states. As these transitions are short it is advisable to use short time windows of data to model and the higher the frequency of the oscillations you are interested in, the shorter this time window should be. DCM for phase coupling will probably run into memory problems if using long time windows or large numbers of trials.

16.12.2 Electromagnetic model

Currently, DCM for phase-coupled responses will work with either ECD or LFP data. Note that a difference to DCM for evoked responses is that the parameters of the spatial model are not optimized. This means that DCM for phase-coupled responses will project the data into source space using the spatial locations you provide.

16.12.3 Neuronal model

This is where you specify the connection architecture for the weakly coupled oscillator model. If using the GUI, the Phase Interaction Functions are given by $a_{ij}sin(\phi_i - \phi_j)$ where a_{ij} are the connection weights that appear in the A-matrix and ϕ_i and ϕ_j are the phases in regions i and j. DCM for phase coupling can also be run from a MATLAB script. This provides greater flexibility in that the Phase Interaction Functions can be approximated using arbitrary order Fourier series. Have a look in the `example_scripts` to see how.

16.12.4 Hilbert transform

Pressing this button does two things. First, source data are bandpass filtered into the specified range. Second, a Hilbert transform is applied from which time series of phase variables are obtained.

16.12.5 Results

Sin(Data) - Region i

This plots the sin of the data (ie. sin of phase variable) and the corresponding model fit for the ith region.

Coupling (A),(B)

This will display the intrinsic and modulatory coupling matrices. The i, jth entry in A specifies how quickly region i changes its phase to align with region j. The corresponding entry in B shows how these values are changed by experimental manipulation.

Part V

Utilities

Chapter 17

Display Image

Contents

17.1 Image to Display 136

This is an interactive facility that allows orthogonal sections from an image volume to be displayed. Clicking the cursor on either of the three images moves the point around which the orthogonal sections are viewed. The co-ordinates of the cursor are shown both in voxel co-ordinates and millimetres within some fixed framework. The intensity at that point in the image (sampled using the current interpolation scheme) is also given. The position of the cross-hairs can also be moved by specifying the co-ordinates in millimetres to which they should be moved. Clicking on the horizontal bar above these boxes will move the cursor back to the origin (analogous to setting the cross-hair position (in mm) to [0 0 0]).

The images can be re-oriented by entering appropriate translations, rotations and zooms into the panel on the left. The transformations can then be saved by hitting the "Reorient images..." button. The transformations that were applied to the image are saved to the header information of the selected images. The transformations are considered to be relative to any existing transformations that may be stored. Note that the order that the transformations are applied in is the same as in spm_matrix.m.

The "Reset..." button next to it is for setting the orientation of images back to transverse. It retains the current voxel sizes, but sets the origin of the images to be the centre of the volumes and all rotations back to zero.

The right panel shows miscellaneous information about the image. This includes:

Dimensions - the x, y and z dimensions of the image.

Datatype - the computer representation of each voxel.

Intensity - scale-factors and possibly a DC offset.

Miscellaneous other information about the image.

Vox size - the distance (in mm) between the centres of neighbouring voxels.

Origin - the voxel at the origin of the co-ordinate system

DIr Cos - Direction cosines. This is a widely used representation of the orientation of an image.

There are also a few options for different resampling modes, zooms etc. You can also flip between voxel space (as would be displayed by Analyze) or world space (the orientation that SPM considers the image to be in). If you are re-orienting the images, make sure that world space is specified. Blobs (from activation studies) can be superimposed on the images and the intensity windowing can also be changed.

If you have put your images in the correct file format, then (possibly after specifying some rigid-body rotations):

The top-left image is coronal with the top (superior) of the head displayed at the top and the left shown on the left. This is as if the subject is viewed from behind.

The bottom-left image is axial with the front (anterior) of the head at the top and the left shown on the left. This is as if the subject is viewed from above.

The top-right image is sagittal with the front (anterior) of the head at the left and the top of the head shown at the top. This is as if the subject is viewed from the left.

17.1 Image to Display

Image to display.

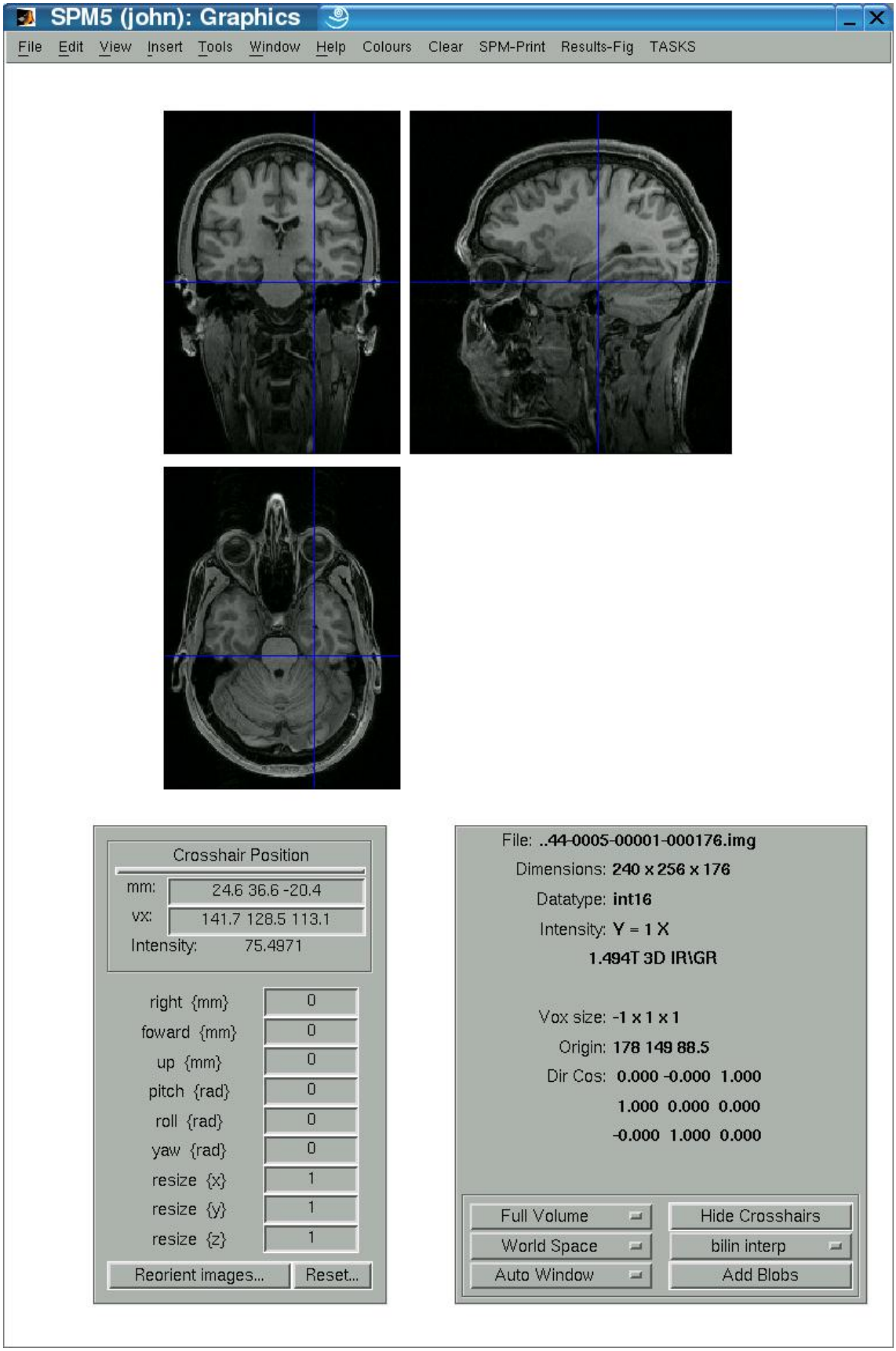


Figure 17.1: The Display routine.

Chapter 18

Check Registration

Contents

18.1 Images to Display 139

Orthogonal views of one or more images are displayed. Clicking in any image moves the centre of the orthogonal views. Images are shown in orientations relative to that of the first selected image. The first specified image is shown at the top-left, and the last at the bottom right. The fastest increment is in the left-to-right direction (the same as you are reading this).

If you have put your images in the correct file format, then (possibly after specifying some rigid-body rotations):

The top-left image is coronal with the top (superior) of the head displayed at the top and the left shown on the left. This is as if the subject is viewed from behind.

The bottom-left image is axial with the front (anterior) of the head at the top and the left shown on the left. This is as if the subject is viewed from above.

The top-right image is sagittal with the front (anterior) of the head at the left and the top of the head shown at the top. This is as if the subject is viewed from the left.

18.1 Images to Display

Images to display.

Chapter 19

Image Calculator

Contents

19.1 Input Images 141
19.2 Output Filename 141
19.3 Output Directory 141
19.4 Expression 142
19.5 Options 142
19.5.1 Data Matrix 142
19.5.2 Masking 142
19.5.3 Interpolation 142
19.5.4 Data Type 142

The image calculator is for performing user-specified algebraic manipulations on a set of images, with the result being written out as an image. The user is prompted to supply images to work on, a filename for the output image, and the expression to evaluate. The expression should be a standard MATLAB expression, within which the images should be referred to as i1, i2, i3,... etc.

19.1 Input Images

These are the images that are used by the calculator. They are referred to as i1, i2, i3, etc in the order that they are specified.

19.2 Output Filename

The output image is written to current working directory unless a valid full pathname is given. If a path name is given here, the output directory setting will be ignored.

19.3 Output Directory

Files produced by this function will be written into this output directory. If no directory is given, images will be written to current working directory. If both output filename and output directory contain a directory, then output filename takes precedence.

19.4 Expression

Example expressions (f):
* Mean of six images (select six images)
f = '(i1+i2+i3+i4+i5+i6)/6'
* Make a binary mask image at threshold of 100
f = 'i1>100'
* Make a mask from one image and apply to another
f = 'i2.*(i1>100)'
- here the first image is used to make the mask, which is applied to the second image
* Sum of n images
f = 'i1 + i2 + i3 + i4 + i5 + ...'
* Sum of n images (when reading data into a data-matrix - use dmtx arg)
f = 'sum(X)'

19.5 Options

Options for image calculator

19.5.1 Data Matrix

If the dmtx flag is set, then images are read into a data matrix X (rather than into separate variables i1, i2, i3,...). The data matrix should be referred to as X, and contains images in rows. Computation is plane by plane, so in data-matrix mode, X is a NxK matrix, where N is the number of input images [prod(size(Vi))], and K is the number of voxels per plane [prod(Vi(1).dim(1:2))].

19.5.2 Masking

For data types without a representation of NaN, implicit zero masking assumes that all zero voxels are to be treated as missing, and treats them as NaN. NaN's are written as zero (by spm_write_plane), for data types without a representation of NaN.

19.5.3 Interpolation

With images of different sizes and orientations, the size and orientation of the first is used for the output image. A warning is given in this situation. Images are sampled into this orientation using the interpolation specified by the hold parameter.

The method by which the images are sampled when being written in a different space.
Nearest Neighbour
- Fastest, but not normally recommended.
Bilinear Interpolation
- OK for PET, or realigned fMRI.
Sinc Interpolation
- Better quality (but slower) interpolation, especially
with higher degrees.

19.5.4 Data Type

Data-type of output image

Chapter 20

DICOM Import

Contents

20.1 DICOM files 143
20.2 Directory structure for converted files 143
20.3 Output directory 143
20.4 Conversion options 144
20.4.1 Output image format 144
20.4.2 Use ICEDims in filename 144

DICOM Conversion. Most scanners produce data in DICOM format. This routine attempts to convert DICOM files into SPM compatible image volumes, which are written into the current directory by default. Note that not all flavours of DICOM can be handled, as DICOM is a very complicated format, and some scanner manufacturers use their own fields, which are not in the official documentation at http://medical.nema.org/

20.1 DICOM files

Select the DICOM files to convert.

20.2 Directory structure for converted files

Choose root directory of converted file tree. The options are:
* Output directory: ./<StudyDate-StudyTime>: Automatically determine the project name and try to convert into the output directory, starting with a StudyDate-StudyTime subdirectory. This option is useful if automatic project recognition fails and one wants to convert data into a project directory.
* Output directory: ./<PatientID>: Convert into the output directory, starting with a PatientID subdirectory.
* Output directory: ./<PatientName>: Convert into the output directory, starting with a PatientName subdirectory.
* No directory hierarchy: Convert all files into the output directory, without sequence/series subdirectories

20.3 Output directory

Select a directory where files are written.

20.4 Conversion options

20.4.1 Output image format

DICOM conversion can create separate img and hdr files or combine them in one file. The single file option will help you save space on your hard disk, but may be incompatible with programs that are not NIfTI-aware.

In any case, only 3D image files will be produced.

20.4.2 Use ICEDims in filename

If image sorting fails, one can try using the additional SIEMENS ICEDims information to create unique filenames. Use this only if there would be multiple volumes with exactly the same file names.

Chapter 21

MINC Import

Contents

21.1 MINC files 145
21.2 Options 145
21.2.1 Data Type 145
21.2.2 NIFTI Type 145

MINC Conversion. MINC is the image data format used for exchanging data within the ICBM community, and the format used by the MNI software tools. It is based on NetCDF, but due to be superceded by a new version relatively soon. MINC is no longer supported for reading images into SPM, so MINC files need to be converted to NIFTI format in order to use them. See http://www.bic.mni.mcgill.ca/software/ for more information.

21.1 MINC files

Select the MINC files to convert.

21.2 Options

Conversion options

21.2.1 Data Type

Data-type of output images. Note that the number of bits used determines the accuracy, and the amount of disk space needed.

21.2.2 NIFTI Type

Output files can be written as .img + .hdr, or the two can be combined into a .nii file.

Chapter 22

ECAT Import

Contents

22.1 ECAT files 147
22.2 Options 147
22.2.1 NIFTI Type 147

ECAT 7 Conversion. ECAT 7 is the image data format used by the more recent CTI PET scanners.

22.1 ECAT files

Select the ECAT files to convert.

22.2 Options

Conversion options

22.2.1 NIFTI Type

Output files can be written as .img + .hdr, or the two can be combined into a .nii file.

Chapter 23

Deformations

Contents

23.1 Composition 149
23.1.1 Imported _sn.mat 149
23.1.2 DARTEL flow 150
23.1.3 Deformation Field 150
23.1.4 Identity 150
23.1.5 Inverse 150
23.1.6 Composition 156
23.2 Save as 163
23.3 Apply to 163
23.4 Interpolation 163

This is a utility for working with deformation fields. They can be loaded, inverted, combined etc, and the results either saved to disk, or applied to some image.

23.1 Composition

Deformation fields can be thought of as mappings. These can be combined by the operation of "composition", which is usually denoted by a circle "o". Suppose x:A->B and y:B->C are two mappings, where A, B and C refer to domains in 3 dimensions. Each element a in A points to element x(a) in B. This in turn points to element y(x(a)) in C, so we have a mapping from A to C. The composition of these mappings is denoted by yox:A->C. Compositions can be combined in an associative way, such that zo(yox) = (zoy)ox.

In this utility, the left-to-right order of the compositions is from top to bottom (note that the rightmost deformation would actually be applied first). i.e. ...((first o second) o third)...o last. The resulting deformation field will have the same domain as the first deformation specified, and will map to voxels in the codomain of the last specified deformation field.

23.1.1 Imported _sn.mat

Spatial normalisation, and the unified segmentation model of SPM5 save a parameterisation of deformation fields. These consist of a combination of an affine transform, and nonlinear warps that are parameterised by a linear combination of cosine transform basis functions. These are saved in *_sn.mat files, which can be converted to deformation fields.

Parameter File

Specify the _sn.mat to be used.

Voxel sizes

Specify the voxel sizes of the deformation field to be produced. Non-finite values will default to the voxel sizes of the template imagethat was originally used to estimate the deformation.

Bounding box

Specify the bounding box of the deformation field to be produced. Non-finite values will default to the bounding box of the template imagethat was originally used to estimate the deformation.

23.1.2 DARTEL flow

Imported DARTEL flow field.

Flow field

The flow field stores the deformation information. The same field can be used for both forward or backward deformations (or even, in principle, half way or exaggerated deformations).

Forward/Backwards

The direction of the DARTEL flow. Note that a backward transform will warp an individual subject's to match the template (ie maps from template to individual). A forward transform will warp the template image to the individual.

Time Steps

The number of time points used for solving the partial differential equations. A single time point would be equivalent to a small deformation model. Smaller values allow faster computations, but are less accurate in terms of inverse consistency and may result in the one-to-one mapping breaking down.

23.1.3 Deformation Field

Deformations can be thought of as vector fields. These can be represented by three-volume images.

23.1.4 Identity

This option generates an identity transform, but this can be useful for changing the dimensions of the resulting deformation (and any images that are generated from it). Dimensions, orientation etc are derived from an image.

Image to base Id on

Specify the image file on which to base the dimensions, orientation etc.

23.1.5 Inverse

Creates the inverse of a deformation field. Deformations are assumed to be one-to-one, in which case they have a unique inverse. If y':A->B is the inverse of y:B->A, then y' o y = y o y' = Id, where Id is the identity transform.

Deformations are inverted using the method described in the appendix of:

* Ashburner J, Andersson JLR & Friston KJ (2000) "Image Registration using a Symmetric Prior - in Three-Dimensions." Human Brain Mapping 9(4):212-225

Composition

Deformation fields can be thought of as mappings. These can be combined by the operation of "composition", which is usually denoted by a circle "o". Suppose x:A->B and y:B->C are two mappings, where A, B and C refer to domains in 3 dimensions. Each element a in A points to element x(a) in B. This in turn points to element y(x(a)) in C, so we have a mapping from A to C. The composition of these mappings is denoted by yox:A->C. Compositions can be combined in an associative way, such that zo(yox) = (zoy)ox.

In this utility, the left-to-right order of the compositions is from top to bottom (note that the rightmost deformation would actually be applied first). i.e. ...((first o second) o third)...o last. The resulting deformation field will have the same domain as the first deformation specified, and will map to voxels in the codomain of the last specified deformation field.

Imported _sn.mat Spatial normalisation, and the unified segmentation model of SPM5 save a parameterisation of deformation fields. These consist of a combination of an affine transform, and nonlinear warps that are parameterised by a linear combination of cosine transform basis functions. These are saved in *_sn.mat files, which can be converted to deformation fields.

Parameter File Specify the _sn.mat to be used.

Voxel sizes Specify the voxel sizes of the deformation field to be produced. Non-finite values will default to the voxel sizes of the template imagethat was originally used to estimate the deformation.

Bounding box Specify the bounding box of the deformation field to be produced. Non-finite values will default to the bounding box of the template imagethat was originally used to estimate the deformation.

DARTEL flow Imported DARTEL flow field.

Flow field The flow field stores the deformation information. The same field can be used for both forward or backward deformations (or even, in principle, half way or exaggerated deformations).

Forward/Backwards The direction of the DARTEL flow. Note that a backward transform will warp an individual subject's to match the template (ie maps from template to individual). A forward transform will warp the template image to the individual.

Time Steps The number of time points used for solving the partial differential equations. A single time point would be equivalent to a small deformation model. Smaller values allow faster computations, but are less accurate in terms of inverse consistency and may result in the one-to-one mapping breaking down.

Deformation Field Deformations can be thought of as vector fields. These can be represented by three-volume images.

Identity This option generates an identity transform, but this can be useful for changing the dimensions of the resulting deformation (and any images that are generated from it). Dimensions, orientation etc are derived from an image.

Image to base Id on Specify the image file on which to base the dimensions, orientation etc.

Inverse Creates the inverse of a deformation field. Deformations are assumed to be one-to-one, in which case they have a unique inverse. If y':A->B is the inverse of y:B->A, then y' o y = y o y' = Id, where Id is the identity transform.

Deformations are inverted using the method described in the appendix of:

* Ashburner J, Andersson JLR & Friston KJ (2000) "Image Registration using a Symmetric Prior - in Three-Dimensions." Human Brain Mapping 9(4):212-225

Composition Deformation fields can be thought of as mappings. These can be combined by the operation of "composition", which is usually denoted by a circle "o". Suppose x:A->B and y:B->C are two mappings, where A, B and C refer to domains in 3 dimensions. Each element a in A points to element x(a) in B. This in turn points to element y(x(a)) in C, so we have a mapping from A to C. The composition of these mappings is denoted by yox:A->C. Compositions can be combined in an associative way, such that zo(yox) = (zoy)ox.

In this utility, the left-to-right order of the compositions is from top to bottom (note that the rightmost deformation would actually be applied first). i.e. ...((first o second) o third)...o last. The resulting deformation field will have the same domain as the first deformation specified, and will map to voxels in the codomain of the last specified deformation field.

Imported _sn.mat Spatial normalisation, and the unified segmentation model of SPM5 save a parameterisation of deformation fields. These consist of a combination of an affine transform, and nonlinear warps that are parameterised by a linear combination of cosine transform basis functions. These are saved in *_sn.mat files, which can be converted to deformation fields.

Parameter File Specify the _sn.mat to be used.

Voxel sizes Specify the voxel sizes of the deformation field to be produced. Non-finite values will default to the voxel sizes of the template imagethat was originally used to estimate the deformation.

Bounding box Specify the bounding box of the deformation field to be produced. Non-finite values will default to the bounding box of the template imagethat was originally used to estimate the deformation.

DARTEL flow Imported DARTEL flow field.

Flow field The flow field stores the deformation information. The same field can be used for both forward or backward deformations (or even, in principle, half way or exaggerated deformations).

Forward/Backwards The direction of the DARTEL flow. Note that a backward transform will warp an individual subject's to match the template (ie maps from template to individual). A forward transform will warp the template image to the individual.

Time Steps The number of time points used for solving the partial differential equations. A single time point would be equivalent to a small deformation model. Smaller values allow faster computations, but are less accurate in terms of inverse consistency and may result in the one-to-one mapping breaking down.

Deformation Field Deformations can be thought of as vector fields. These can be represented by three-volume images.

Identity This option generates an identity transform, but this can be useful for changing the dimensions of the resulting deformation (and any images that are generated from it). Dimensions, orientation etc are derived from an image.

Image to base Id on Specify the image file on which to base the dimensions, orientation etc.

Inverse Creates the inverse of a deformation field. Deformations are assumed to be one-to-one, in which case they have a unique inverse. If y':A->B is the inverse of y:B->A, then y' o y = y o y' = Id, where Id is the identity transform.

Deformations are inverted using the method described in the appendix of:

* Ashburner J, Andersson JLR & Friston KJ (2000) "Image Registration using a Symmetric Prior - in Three-Dimensions." Human Brain Mapping 9(4):212-225

Composition Deformation fields can be thought of as mappings. These can be combined by the operation of "composition", which is usually denoted by a circle "o". Suppose x:A->B and y:B->C are two mappings, where A, B and C refer to domains in 3 dimensions. Each element a in A points to element x(a) in B. This in turn points to element y(x(a)) in C, so we have a mapping

from A to C. The composition of these mappings is denoted by yox:A->C. Compositions can be combined in an associative way, such that zo(yox) = (zoy)ox.

In this utility, the left-to-right order of the compositions is from top to bottom (note that the rightmost deformation would actually be applied first). i.e. ...((first o second) o third)...o last. The resulting deformation field will have the same domain as the first deformation specified, and will map to voxels in the codomain of the last specified deformation field.

Imported _sn.mat Spatial normalisation, and the unified segmentation model of SPM5 save a parameterisation of deformation fields. These consist of a combination of an affine transform, and nonlinear warps that are parameterised by a linear combination of cosine transform basis functions. These are saved in *_sn.mat files, which can be converted to deformation fields.

Parameter File Specify the _sn.mat to be used.

Voxel sizes Specify the voxel sizes of the deformation field to be produced. Non-finite values will default to the voxel sizes of the template imagethat was originally used to estimate the deformation.

Bounding box Specify the bounding box of the deformation field to be produced. Non-finite values will default to the bounding box of the template imagethat was originally used to estimate the deformation.

DARTEL flow Imported DARTEL flow field.

Flow field The flow field stores the deformation information. The same field can be used for both forward or backward deformations (or even, in principle, half way or exaggerated deformations).

Forward/Backwards The direction of the DARTEL flow. Note that a backward transform will warp an individual subject's to match the template (ie maps from template to individual). A forward transform will warp the template image to the individual.

Time Steps The number of time points used for solving the partial differential equations. A single time point would be equivalent to a small deformation model. Smaller values allow faster computations, but are less accurate in terms of inverse consistency and may result in the one-to-one mapping breaking down.

Deformation Field Deformations can be thought of as vector fields. These can be represented by three-volume images.

Identity This option generates an identity transform, but this can be useful for changing the dimensions of the resulting deformation (and any images that are generated from it). Dimensions, orientation etc are derived from an image.

Image to base Id on Specify the image file on which to base the dimensions, orientation etc.

IMAGE TO BASE INVERSE ON Specify the image file on which to base the dimensions, orientation etc.

Composition Deformation fields can be thought of as mappings. These can be combined by the operation of "composition", which is usually denoted by a circle "o". Suppose x:A->B and y:B->C are two mappings, where A, B and C refer to domains in 3 dimensions. Each element a in A points to element x(a) in B. This in turn points to element y(x(a)) in C, so we have a mapping from A to C. The composition of these mappings is denoted by yox:A->C. Compositions can be combined in an associative way, such that zo(yox) = (zoy)ox.

In this utility, the left-to-right order of the compositions is from top to bottom (note that the rightmost deformation would actually be applied first). i.e. ...((first o second) o third)...o last. The resulting deformation field will have the same domain as the first deformation specified, and will map to voxels in the codomain of the last specified deformation field.

IMPORTED _SN.MAT Spatial normalisation, and the unified segmentation model of SPM5 save a parameterisation of deformation fields. These consist of a combination of an affine transform, and nonlinear warps that are parameterised by a linear combination of cosine transform basis functions. These are saved in *_sn.mat files, which can be converted to deformation fields.

Parameter File Specify the _sn.mat to be used.

Voxel sizes Specify the voxel sizes of the deformation field to be produced. Non-finite values will default to the voxel sizes of the template imagethat was originally used to estimate the deformation.

Bounding box Specify the bounding box of the deformation field to be produced. Non-finite values will default to the bounding box of the template imagethat was originally used to estimate the deformation.

DARTEL FLOW Imported DARTEL flow field.

Flow field The flow field stores the deformation information. The same field can be used for both forward or backward deformations (or even, in principle, half way or exaggerated deformations).

Forward/Backwards The direction of the DARTEL flow. Note that a backward transform will warp an individual subject's to match the template (ie maps from template to individual). A forward transform will warp the template image to the individual.

Time Steps The number of time points used for solving the partial differential equations. A single time point would be equivalent to a small deformation model. Smaller values allow faster computations, but are less accurate in terms of inverse consistency and may result in the one-to-one mapping breaking down.

DEFORMATION FIELD Deformations can be thought of as vector fields. These can be represented by three-volume images.

IDENTITY This option generates an identity transform, but this can be useful for changing the dimensions of the resulting deformation (and any images that are generated from it). Dimensions, orientation etc are derived from an image.

Image to base Id on Specify the image file on which to base the dimensions, orientation etc.

Image to base inverse on Specify the image file on which to base the dimensions, orientation etc.

Composition Deformation fields can be thought of as mappings. These can be combined by the operation of "composition", which is usually denoted by a circle "o". Suppose x:A->B and y:B->C are two mappings, where A, B and C refer to domains in 3 dimensions. Each element a in A points to element x(a) in B. This in turn points to element y(x(a)) in C, so we have a mapping from A to C. The composition of these mappings is denoted by yox:A->C. Compositions can be combined in an associative way, such that zo(yox) = (zoy)ox.

In this utility, the left-to-right order of the compositions is from top to bottom (note that the rightmost deformation would actually be applied first). i.e. ...((first o second) o third)...o last. The resulting deformation field will have the same domain as the first deformation specified, and will map to voxels in the codomain of the last specified deformation field.

Imported _sn.mat Spatial normalisation, and the unified segmentation model of SPM5 save a parameterisation of deformation fields. These consist of a combination of an affine transform, and nonlinear warps that are parameterised by a linear combination of cosine transform basis functions. These are saved in *_sn.mat files, which can be converted to deformation fields.

Parameter File Specify the _sn.mat to be used.

Voxel sizes Specify the voxel sizes of the deformation field to be produced. Non-finite values will default to the voxel sizes of the template imagethat was originally used to estimate the deformation.

Bounding box Specify the bounding box of the deformation field to be produced. Non-finite values will default to the bounding box of the template imagethat was originally used to estimate the deformation.

DARTEL flow Imported DARTEL flow field.

Flow field The flow field stores the deformation information. The same field can be used for both forward or backward deformations (or even, in principle, half way or exaggerated deformations).

Forward/Backwards The direction of the DARTEL flow. Note that a backward transform will warp an individual subject's to match the template (ie maps from template to individual). A forward transform will warp the template image to the individual.

Time Steps The number of time points used for solving the partial differential equations. A single time point would be equivalent to a small deformation model. Smaller values allow faster computations, but are less accurate in terms of inverse consistency and may result in the one-to-one mapping breaking down.

Deformation Field Deformations can be thought of as vector fields. These can be represented by three-volume images.

Identity This option generates an identity transform, but this can be useful for changing the dimensions of the resulting deformation (and any images that are generated from it). Dimensions, orientation etc are derived from an image.

Image to base Id on Specify the image file on which to base the dimensions, orientation etc.

Inverse Creates the inverse of a deformation field. Deformations are assumed to be one-to-one, in which case they have a unique inverse. If y':A->B is the inverse of y:B->A, then y' o y = y o y' = Id, where Id is the identity transform.

Deformations are inverted using the method described in the appendix of:

* Ashburner J, Andersson JLR & Friston KJ (2000) "Image Registration using a Symmetric Prior - in Three-Dimensions." Human Brain Mapping 9(4):212-225

Composition Deformation fields can be thought of as mappings. These can be combined by the operation of "composition", which is usually denoted by a circle "o". Suppose x:A->B and y:B->C are two mappings, where A, B and C refer to domains in 3 dimensions. Each element a in A points to element x(a) in B. This in turn points to element y(x(a)) in C, so we have a mapping from A to C. The composition of these mappings is denoted by yox:A->C. Compositions can be combined in an associative way, such that zo(yox) = (zoy)ox.

In this utility, the left-to-right order of the compositions is from top to bottom (note that the rightmost deformation would actually be applied first). i.e. ...((first o second) o third)...o last. The resulting deformation field will have the same domain as the first deformation specified, and will map to voxels in the codomain of the last specified deformation field.

IMPORTED _SN.MAT Spatial normalisation, and the unified segmentation model of SPM5 save a parameterisation of deformation fields. These consist of a combination of an affine transform, and nonlinear warps that are parameterised by a linear combination of cosine transform basis functions. These are saved in *_sn.mat files, which can be converted to deformation fields.

Parameter File Specify the _sn.mat to be used.

Voxel sizes Specify the voxel sizes of the deformation field to be produced. Non-finite values will default to the voxel sizes of the template imagethat was originally used to estimate the deformation.

Bounding box Specify the bounding box of the deformation field to be produced. Non-finite values will default to the bounding box of the template imagethat was originally used to estimate the deformation.

DARTEL FLOW Imported DARTEL flow field.

Flow field The flow field stores the deformation information. The same field can be used for both forward or backward deformations (or even, in principle, half way or exaggerated deformations).

Forward/Backwards The direction of the DARTEL flow. Note that a backward transform will warp an individual subject's to match the template (ie maps from template to individual). A forward transform will warp the template image to the individual.

Time Steps The number of time points used for solving the partial differential equations. A single time point would be equivalent to a small deformation model. Smaller values allow faster computations, but are less accurate in terms of inverse consistency and may result in the one-to-one mapping breaking down.

DEFORMATION FIELD Deformations can be thought of as vector fields. These can be represented by three-volume images.

IDENTITY This option generates an identity transform, but this can be useful for changing the dimensions of the resulting deformation (and any images that are generated from it). Dimensions, orientation etc are derived from an image.

Image to base Id on Specify the image file on which to base the dimensions, orientation etc.

Image to base inverse on Specify the image file on which to base the dimensions, orientation etc.

Composition Deformation fields can be thought of as mappings. These can be combined by the operation of "composition", which is usually denoted by a circle "o". Suppose x:A->B and y:B->C are two mappings, where A, B and C refer to domains in 3 dimensions. Each element a in A points to element x(a) in B. This in turn points to element y(x(a)) in C, so we have a mapping from A to C. The composition of these mappings is denoted by yox:A->C. Compositions can be combined in an associative way, such that zo(yox) = (zoy)ox.

In this utility, the left-to-right order of the compositions is from top to bottom (note that the rightmost deformation would actually be applied first). i.e. ...((first o second) o third)...o last. The resulting deformation field will have the same domain as the first deformation specified, and will map to voxels in the codomain of the last specified deformation field.

Imported _sn.mat Spatial normalisation, and the unified segmentation model of SPM5 save a parameterisation of deformation fields. These consist of a combination of an affine transform, and nonlinear warps that are parameterised by a linear combination of cosine transform basis functions. These are saved in *_sn.mat files, which can be converted to deformation fields.

Parameter File Specify the _sn.mat to be used.

Voxel sizes Specify the voxel sizes of the deformation field to be produced. Non-finite values will default to the voxel sizes of the template imagethat was originally used to estimate the deformation.

Bounding box Specify the bounding box of the deformation field to be produced. Non-finite values will default to the bounding box of the template imagethat was originally used to estimate the deformation.

DARTEL flow Imported DARTEL flow field.

Flow field The flow field stores the deformation information. The same field can be used for both forward or backward deformations (or even, in principle, half way or exaggerated deformations).

Forward/Backwards The direction of the DARTEL flow. Note that a backward transform will warp an individual subject's to match the template (ie maps from template to individual). A forward transform will warp the template image to the individual.

Time Steps The number of time points used for solving the partial differential equations. A single time point would be equivalent to a small deformation model. Smaller values allow faster computations, but are less accurate in terms of inverse consistency and may result in the one-to-one mapping breaking down.

Deformation Field Deformations can be thought of as vector fields. These can be represented by three-volume images.

Identity This option generates an identity transform, but this can be useful for changing the dimensions of the resulting deformation (and any images that are generated from it). Dimensions, orientation etc are derived from an image.

Image to base Id on Specify the image file on which to base the dimensions, orientation etc.

Image to base inverse on

Specify the image file on which to base the dimensions, orientation etc.

23.1.6 Composition

Deformation fields can be thought of as mappings. These can be combined by the operation of "composition", which is usually denoted by a circle "o". Suppose x:A->B and y:B->C are two mappings, where A, B and C refer to domains in 3 dimensions. Each element a in A points to element x(a) in B. This in turn points to element y(x(a)) in C, so we have a mapping from A to C. The composition of these mappings is denoted by yox:A->C. Compositions can be combined in an associative way, such that zo(yox) = (zoy)ox.

In this utility, the left-to-right order of the compositions is from top to bottom (note that the rightmost deformation would actually be applied first). i.e. ...((first o second) o third)...o last. The resulting deformation field will have the same domain as the first deformation specified, and will map to voxels in the codomain of the last specified deformation field.

Imported _sn.mat

Spatial normalisation, and the unified segmentation model of SPM5 save a parameterisation of deformation fields. These consist of a combination of an affine transform, and nonlinear warps that are parameterised by a linear combination of cosine transform basis functions. These are saved in *_sn.mat files, which can be converted to deformation fields.

Parameter File Specify the _sn.mat to be used.

Voxel sizes Specify the voxel sizes of the deformation field to be produced. Non-finite values will default to the voxel sizes of the template imagethat was originally used to estimate the deformation.

Bounding box Specify the bounding box of the deformation field to be produced. Non-finite values will default to the bounding box of the template imagethat was originally used to estimate the deformation.

DARTEL flow

Imported DARTEL flow field.

Flow field The flow field stores the deformation information. The same field can be used for both forward or backward deformations (or even, in principle, half way or exaggerated deformations).

Forward/Backwards The direction of the DARTEL flow. Note that a backward transform will warp an individual subject's to match the template (ie maps from template to individual). A forward transform will warp the template image to the individual.

Time Steps The number of time points used for solving the partial differential equations. A single time point would be equivalent to a small deformation model. Smaller values allow faster computations, but are less accurate in terms of inverse consistency and may result in the one-to-one mapping breaking down.

Deformation Field

Deformations can be thought of as vector fields. These can be represented by three-volume images.

Identity

This option generates an identity transform, but this can be useful for changing the dimensions of the resulting deformation (and any images that are generated from it). Dimensions, orientation etc are derived from an image.

Image to base Id on Specify the image file on which to base the dimensions, orientation etc.

Inverse

Creates the inverse of a deformation field. Deformations are assumed to be one-to-one, in which case they have a unique inverse. If y':A->B is the inverse of y:B->A, then y' o y = y o y' = Id, where Id is the identity transform.

Deformations are inverted using the method described in the appendix of:

* Ashburner J, Andersson JLR & Friston KJ (2000) "Image Registration using a Symmetric Prior - in Three-Dimensions." Human Brain Mapping 9(4):212-225

Composition Deformation fields can be thought of as mappings. These can be combined by the operation of "composition", which is usually denoted by a circle "o". Suppose x:A->B and y:B->C are two mappings, where A, B and C refer to domains in 3 dimensions. Each element a in A points to element x(a) in B. This in turn points to element y(x(a)) in C, so we have a mapping from A to C. The composition of these mappings is denoted by yox:A->C. Compositions can be combined in an associative way, such that zo(yox) = (zoy)ox.

In this utility, the left-to-right order of the compositions is from top to bottom (note that the rightmost deformation would actually be applied first). i.e. ...((first o second) o third)...o last. The resulting deformation field will have the same domain as the first deformation specified, and will map to voxels in the codomain of the last specified deformation field.

Imported _sn.mat Spatial normalisation, and the unified segmentation model of SPM5 save a parameterisation of deformation fields. These consist of a combination of an affine transform, and nonlinear warps that are parameterised by a linear combination of cosine transform basis functions. These are saved in *_sn.mat files, which can be converted to deformation fields.

Parameter File Specify the _sn.mat to be used.

Voxel sizes Specify the voxel sizes of the deformation field to be produced. Non-finite values will default to the voxel sizes of the template imagethat was originally used to estimate the deformation.

Bounding box Specify the bounding box of the deformation field to be produced. Non-finite values will default to the bounding box of the template imagethat was originally used to estimate the deformation.

DARTEL flow Imported DARTEL flow field.

Flow field The flow field stores the deformation information. The same field can be used for both forward or backward deformations (or even, in principle, half way or exaggerated deformations).

Forward/Backwards The direction of the DARTEL flow. Note that a backward transform will warp an individual subject's to match the template (ie maps from template to individual). A forward transform will warp the template image to the individual.

Time Steps The number of time points used for solving the partial differential equations. A single time point would be equivalent to a small deformation model. Smaller values allow faster computations, but are less accurate in terms of inverse consistency and may result in the one-to-one mapping breaking down.

Deformation Field Deformations can be thought of as vector fields. These can be represented by three-volume images.

Identity This option generates an identity transform, but this can be useful for changing the dimensions of the resulting deformation (and any images that are generated from it). Dimensions, orientation etc are derived from an image.

Image to base Id on Specify the image file on which to base the dimensions, orientation etc.

Inverse Creates the inverse of a deformation field. Deformations are assumed to be one-to-one, in which case they have a unique inverse. If y':A->B is the inverse of y:B->A, then y' o y = y o y' = Id, where Id is the identity transform.

Deformations are inverted using the method described in the appendix of:

* Ashburner J, Andersson JLR & Friston KJ (2000) "Image Registration using a Symmetric Prior - in Three-Dimensions." Human Brain Mapping 9(4):212-225

Composition Deformation fields can be thought of as mappings. These can be combined by the operation of "composition", which is usually denoted by a circle "o". Suppose x:A->B and y:B->C are two mappings, where A, B and C refer to domains in 3 dimensions. Each element a in A points to element x(a) in B. This in turn points to element y(x(a)) in C, so we have a mapping from A to C. The composition of these mappings is denoted by yox:A->C. Compositions can be combined in an associative way, such that zo(yox) = (zoy)ox.

In this utility, the left-to-right order of the compositions is from top to bottom (note that the rightmost deformation would actually be applied first). i.e. ...((first o second) o third)...o last. The resulting deformation field will have the same domain as the first deformation specified, and will map to voxels in the codomain of the last specified deformation field.

IMPORTED _SN.MAT Spatial normalisation, and the unified segmentation model of SPM5 save a parameterisation of deformation fields. These consist of a combination of an affine transform, and nonlinear warps that are parameterised by a linear combination of cosine transform basis functions. These are saved in *_sn.mat files, which can be converted to deformation fields.

Parameter File Specify the _sn.mat to be used.

Voxel sizes Specify the voxel sizes of the deformation field to be produced. Non-finite values will default to the voxel sizes of the template imagethat was originally used to estimate the deformation.

Bounding box Specify the bounding box of the deformation field to be produced. Non-finite values will default to the bounding box of the template imagethat was originally used to estimate the deformation.

DARTEL FLOW Imported DARTEL flow field.

Flow field The flow field stores the deformation information. The same field can be used for both forward or backward deformations (or even, in principle, half way or exaggerated deformations).

Forward/Backwards The direction of the DARTEL flow. Note that a backward transform will warp an individual subject's to match the template (ie maps from template to individual). A forward transform will warp the template image to the individual.

Time Steps The number of time points used for solving the partial differential equations. A single time point would be equivalent to a small deformation model. Smaller values allow faster computations, but are less accurate in terms of inverse consistency and may result in the one-to-one mapping breaking down.

DEFORMATION FIELD Deformations can be thought of as vector fields. These can be represented by three-volume images.

IDENTITY This option generates an identity transform, but this can be useful for changing the dimensions of the resulting deformation (and any images that are generated from it). Dimensions, orientation etc are derived from an image.

Image to base Id on Specify the image file on which to base the dimensions, orientation etc.

Image to base inverse on Specify the image file on which to base the dimensions, orientation etc.

Composition Deformation fields can be thought of as mappings. These can be combined by the operation of "composition", which is usually denoted by a circle "o". Suppose x:A->B and y:B->C are two mappings, where A, B and C refer to domains in 3 dimensions. Each element a in A points to element x(a) in B. This in turn points to element y(x(a)) in C, so we have a mapping from A to C. The composition of these mappings is denoted by yox:A->C. Compositions can be combined in an associative way, such that zo(yox) = (zoy)ox.

In this utility, the left-to-right order of the compositions is from top to bottom (note that the rightmost deformation would actually be applied first). i.e. ...((first o second) o third)...o last. The resulting deformation field will have the same domain as the first deformation specified, and will map to voxels in the codomain of the last specified deformation field.

Imported _sn.mat Spatial normalisation, and the unified segmentation model of SPM5 save a parameterisation of deformation fields. These consist of a combination of an affine transform, and nonlinear warps that are parameterised by a linear combination of cosine transform basis functions. These are saved in *_sn.mat files, which can be converted to deformation fields.

PARAMETER FILE Specify the _sn.mat to be used.

VOXEL SIZES Specify the voxel sizes of the deformation field to be produced. Non-finite values will default to the voxel sizes of the template imagethat was originally used to estimate the deformation.

BOUNDING BOX Specify the bounding box of the deformation field to be produced. Non-finite values will default to the bounding box of the template imagethat was originally used to estimate the deformation.

DARTEL flow Imported DARTEL flow field.

FLOW FIELD The flow field stores the deformation information. The same field can be used for both forward or backward deformations (or even, in principle, half way or exaggerated deformations).

FORWARD/BACKWARDS The direction of the DARTEL flow. Note that a backward transform will warp an individual subject's to match the template (ie maps from template to individual). A forward transform will warp the template image to the individual.

TIME STEPS The number of time points used for solving the partial differential equations. A single time point would be equivalent to a small deformation model. Smaller values allow faster computations, but are less accurate in terms of inverse consistency and may result in the one-to-one mapping breaking down.

Deformation Field Deformations can be thought of as vector fields. These can be represented by three-volume images.

Identity This option generates an identity transform, but this can be useful for changing the dimensions of the resulting deformation (and any images that are generated from it). Dimensions, orientation etc are derived from an image.

IMAGE TO BASE ID ON Specify the image file on which to base the dimensions, orientation etc.

Image to base inverse on Specify the image file on which to base the dimensions, orientation etc.

Composition

Deformation fields can be thought of as mappings. These can be combined by the operation of "composition", which is usually denoted by a circle "o". Suppose x:A->B and y:B->C are two mappings, where A, B and C refer to domains in 3 dimensions. Each element a in A points to element x(a) in B. This in turn points to element y(x(a)) in C, so we have a mapping from A to C. The composition of these mappings is denoted by yox:A->C. Compositions can be combined in an associative way, such that zo(yox) = (zoy)ox.

In this utility, the left-to-right order of the compositions is from top to bottom (note that the rightmost deformation would actually be applied first). i.e. ...((first o second) o third)...o last. The resulting deformation field will have the same domain as the first deformation specified, and will map to voxels in the codomain of the last specified deformation field.

Imported _sn.mat Spatial normalisation, and the unified segmentation model of SPM5 save a parameterisation of deformation fields. These consist of a combination of an affine transform, and nonlinear warps that are parameterised by a linear combination of cosine transform basis functions. These are saved in *_sn.mat files, which can be converted to deformation fields.

Parameter File Specify the _sn.mat to be used.

Voxel sizes Specify the voxel sizes of the deformation field to be produced. Non-finite values will default to the voxel sizes of the template imagethat was originally used to estimate the deformation.

Bounding box Specify the bounding box of the deformation field to be produced. Non-finite values will default to the bounding box of the template imagethat was originally used to estimate the deformation.

DARTEL flow Imported DARTEL flow field.

Flow field The flow field stores the deformation information. The same field can be used for both forward or backward deformations (or even, in principle, half way or exaggerated deformations).

Forward/Backwards The direction of the DARTEL flow. Note that a backward transform will warp an individual subject's to match the template (ie maps from template to individual). A forward transform will warp the template image to the individual.

Time Steps The number of time points used for solving the partial differential equations. A single time point would be equivalent to a small deformation model. Smaller values allow faster computations, but are less accurate in terms of inverse consistency and may result in the one-to-one mapping breaking down.

Deformation Field Deformations can be thought of as vector fields. These can be represented by three-volume images.

Identity This option generates an identity transform, but this can be useful for changing the dimensions of the resulting deformation (and any images that are generated from it). Dimensions, orientation etc are derived from an image.

Image to base Id on Specify the image file on which to base the dimensions, orientation etc.

Inverse Creates the inverse of a deformation field. Deformations are assumed to be one-to-one, in which case they have a unique inverse. If y':A->B is the inverse of y:B->A, then y' o y = y o y' = Id, where Id is the identity transform.

Deformations are inverted using the method described in the appendix of:

* Ashburner J, Andersson JLR & Friston KJ (2000) "Image Registration using a Symmetric Prior - in Three-Dimensions." Human Brain Mapping 9(4):212-225

Composition Deformation fields can be thought of as mappings. These can be combined by the operation of "composition", which is usually denoted by a circle "o". Suppose x:A->B and y:B->C are two mappings, where A, B and C refer to domains in 3 dimensions. Each element a in A points to element x(a) in B. This in turn points to element y(x(a)) in C, so we have a mapping from A to C. The composition of these mappings is denoted by yox:A->C. Compositions can be combined in an associative way, such that zo(yox) = (zoy)ox.

In this utility, the left-to-right order of the compositions is from top to bottom (note that the rightmost deformation would actually be applied first). i.e. ...((first o second) o third)...o last. The resulting deformation field will have the same domain as the first deformation specified, and will map to voxels in the codomain of the last specified deformation field.

Imported _sn.mat Spatial normalisation, and the unified segmentation model of SPM5 save a parameterisation of deformation fields. These consist of a combination of an affine transform, and nonlinear warps that are parameterised by a linear combination of cosine transform basis functions. These are saved in *_sn.mat files, which can be converted to deformation fields.

Parameter File Specify the _sn.mat to be used.

Voxel sizes Specify the voxel sizes of the deformation field to be produced. Non-finite values will default to the voxel sizes of the template imagethat was originally used to estimate the deformation.

Bounding box Specify the bounding box of the deformation field to be produced. Non-finite values will default to the bounding box of the template imagethat was originally used to estimate the deformation.

DARTEL flow Imported DARTEL flow field.

Flow field The flow field stores the deformation information. The same field can be used for both forward or backward deformations (or even, in principle, half way or exaggerated deformations).

Forward/Backwards The direction of the DARTEL flow. Note that a backward transform will warp an individual subject's to match the template (ie maps from template to individual). A forward transform will warp the template image to the individual.

Time Steps The number of time points used for solving the partial differential equations. A single time point would be equivalent to a small deformation model. Smaller values allow

faster computations, but are less accurate in terms of inverse consistency and may result in the one-to-one mapping breaking down.

Deformation Field Deformations can be thought of as vector fields. These can be represented by three-volume images.

Identity This option generates an identity transform, but this can be useful for changing the dimensions of the resulting deformation (and any images that are generated from it). Dimensions, orientation etc are derived from an image.

IMAGE TO BASE ID ON Specify the image file on which to base the dimensions, orientation etc.

Image to base inverse on Specify the image file on which to base the dimensions, orientation etc.

Composition Deformation fields can be thought of as mappings. These can be combined by the operation of "composition", which is usually denoted by a circle "o". Suppose x:A->B and y:B->C are two mappings, where A, B and C refer to domains in 3 dimensions. Each element a in A points to element x(a) in B. This in turn points to element y(x(a)) in C, so we have a mapping from A to C. The composition of these mappings is denoted by yox:A->C. Compositions can be combined in an associative way, such that zo(yox) = (zoy)ox.

In this utility, the left-to-right order of the compositions is from top to bottom (note that the rightmost deformation would actually be applied first). i.e. ...((first o second) o third)...o last. The resulting deformation field will have the same domain as the first deformation specified, and will map to voxels in the codomain of the last specified deformation field.

Imported _sn.mat Spatial normalisation, and the unified segmentation model of SPM5 save a parameterisation of deformation fields. These consist of a combination of an affine transform, and nonlinear warps that are parameterised by a linear combination of cosine transform basis functions. These are saved in *_sn.mat files, which can be converted to deformation fields.

Parameter File Specify the _sn.mat to be used.

Voxel sizes Specify the voxel sizes of the deformation field to be produced. Non-finite values will default to the voxel sizes of the template imagethat was originally used to estimate the deformation.

Bounding box Specify the bounding box of the deformation field to be produced. Non-finite values will default to the bounding box of the template imagethat was originally used to estimate the deformation.

DARTEL flow Imported DARTEL flow field.

Flow field The flow field stores the deformation information. The same field can be used for both forward or backward deformations (or even, in principle, half way or exaggerated deformations).

Forward/Backwards The direction of the DARTEL flow. Note that a backward transform will warp an individual subject's to match the template (ie maps from template to individual). A forward transform will warp the template image to the individual.

Time Steps The number of time points used for solving the partial differential equations. A single time point would be equivalent to a small deformation model. Smaller values allow faster computations, but are less accurate in terms of inverse consistency and may result in the one-to-one mapping breaking down.

Deformation Field Deformations can be thought of as vector fields. These can be represented by three-volume images.

Identity This option generates an identity transform, but this can be useful for changing the dimensions of the resulting deformation (and any images that are generated from it). Dimensions, orientation etc are derived from an image.

Image to base Id on Specify the image file on which to base the dimensions, orientation etc.

23.2 Save as

Save the result as a three-volume image. "y_" will be prepended to the filename. The result will be written to the current directory.

23.3 Apply to

Apply the resulting deformation field to some images. The warped images will be written to the current directory, and the filenames prepended by "w". Note that trilinear interpolation is used to resample the data, so the original values in the images will not be preserved.

23.4 Interpolation

The method by which the images are sampled when being written in a different space.

Nearest Neighbour: - Fastest, but not normally recommended.

Bilinear Interpolation: - OK for PET, or realigned fMRI.

B-spline Interpolation: - Better quality (but slower) interpolation [69], especially with higher degree splines. Do not use B-splines when there is any region of NaN or Inf in the images.

Part VI

Tools

Chapter 24

FieldMap Toolbox

24.1 Introduction

This chapter describes how to use the FieldMap toolbox version 2.1[1] for creating unwrapped field maps that can be used to do geometric distortion correction of EPI images [45, 44, 42, 43, 2]. The toolbox can be used in an interactive mode so that the user can see the effect of applying different field maps and unwarping parameters to EPI images. However, once a set of parameters has been established for a specific scanning protocol, the toolbox routines can be scripted or batched using the SPM8 user-interface. The toolbox creates a voxel-displacement map that can be used with Realign & Unwarp for doing a combined static and dynamic distortion correction. FieldMap and Unwarp are designed only to work with images collected with the phase-encode direction aligned with the anterior-posterior direction.

24.2 Creating Field Maps Using the FieldMap GUI

The FieldMap Toolbox GUI is shown on the left Figure 24.1. It is divided into two parts. The top part deals with creating the field map in Hz and the bottom part deals with creating the voxel displacement map (VDM) and unwarping the EPI. The toolbox can be used by working through the different inputs in the following order:

24.2.1 Create field map in Hz

Load defaults file

Select the defaults file from which to load default parameters. If necessary, the parameters used to create the field map can be temporarily modified using the GUI. To change the default parameters, edit `pm_defaults.m` or create a new file called `pm_defaults_NAME.m` (as described in Section 24.3).

Data Input Format

PM The acquired field map images are in phase and magnitude format. There may be a single pair of phase and magnitude images (i.e. 2 images) in which case the phase image has been created by the vendor sequence from two echo times acquisitions. Alternatively there may be two pairs of phase and magnitude images, one for each echo time(ie 4 images). The units for the phase images MUST BE RADIANS BETWEEN +pi and -pi. The user will be asked if this is required when the images are selected.

[1] FieldMap Version 2.0 can be downloaded as part SPM5: http://www.fil.ion.ucl.ac.uk/spm/software/spm5/
FieldMap Version 1.1 for SPM2 can be downloaded from http://www.fil.ion.ucl.ac.uk/spm/toolbox/fieldmap/

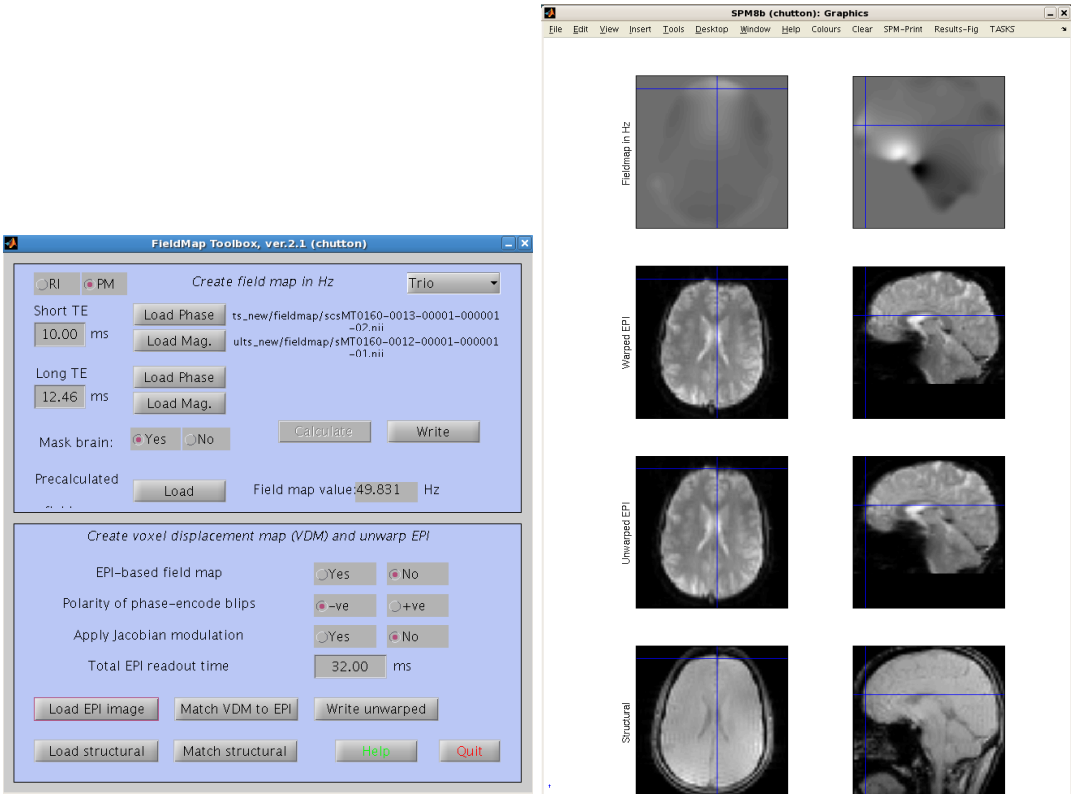


Figure 24.1: FieldMap GUI and Results.

RI The acquired field map images are in real and imaginary format. Two pairs of real and imaginary image volumes, one for a shorter and one for a longer echo time (ie 4 images)[2].

File Selection

Select NIfTI format images. Generally, the acquired scanner files will be in dicom format which can be correctly converted using the DICOM converter in the corresponding version of SPM. DICOM and other image formats can also be converted to using MRIcro[3].

If the data input format is PM, load Phase and Magnitude images:

1. Single phase image OR phase of short echo-time image.
2. Single magnitude image OR magnitude of short echo-time image.
3. LEAVE EMPTY if input consists of a single phase and magnitude pair OR phase of long echo-time image.
4. LEAVE EMPTY if input consists of a single phase and magnitude pair OR magnitude of long echo-time image.

OR If the data input format is RI, load Real and Magnitude images:

1. Real part of short echo-time image.
2. Imaginary part of short echo-time image.
3. Real part of long echo-time image.
4. Imaginary part of long echo-time image.

[2] NB If using SPM2, the data input format can only be changed by editing the spm_defaults.m file. This is described in Section 24.3.

[3] MRIcro is freely available from http://www.cla.sc.edu/psyc/faculty/rorden/mricro.html.

Short TE/Long TE (ms)

Specify the short and long echo times in ms associated with the field map acquisition. Both of these values are required even if a single phase and magnitude image is used as input.

Mask brain

Specify yes to generate a brain mask using the magnitude data which will be used to exclude regions of the field map outside of the brain.

Calculate

Calculate an unwrapped field map in Hz which is stored in memory. This represents the map of phase changes associated with the measured field map data. The processing is described in more detail in Section 24.8 and involves some or all of the following steps (as specified in spm_defaults.m):

1. Calculation of a Hz fieldmap from input data
2. Segmentation to exclude regions outside of the brain
3. Phase unwrapping
4. Smoothing and dilation of the processed fieldmap

The processed field map (in Hz) is displayed in the graphics window (top row, right Figure 24.1) and the field at different points can be explored. The field map in Hz is converted to a VDM (voxel displacement map) using the parameters shown in the FieldMap GUI and saved with the filename vdm5_NAME-OF-FIRST-INPUT-IMAGE.img in the same directory as the acquired field map images. The VDM file is overwritten whenever the field map is recalculated or when any parameters are changed. The resulting VDM file can be used for unwarping the EPI using Realign & Unwarp in SPM8 (see Section 24.7).

Write

Write out the processed field map (in Hz) as a Nifti format image. The image will be saved with the filename fpm_NAME-OF-FIRST-INPUT-IMAGE.img in the same directory as the acquired field map images.

Load Pre-calculated

Load a precalculated unwrapped field map (`fpm_.img`). This should be a single image volume with units of Hz in NIfTI format. The precalculated field map may have been created previously using the FieldMap toolbox or by other means. Once loaded, the field map is displayed in the graphics window (top row, right, Figure 24.1) and the field at different points can be explored.

Field map value (Hz)

Interrogate the value of the field map in Hz at the location specified by the mouse pointer in the graphics window.

24.2.2 Create voxel displacement map (VDM) and unwarp EPI

When any of the parameters below are changed, a new VDM is created and written out as vdm5_NAME-OF-FIRST-INPUT-IMAGE.img. The vdm5_NAME-OF-FIRST-INPUT-IMAGE.mat file is not updated unless 'Match VDM to EPI' is selected as described in Section 24.2.2.

EPI-based field map - Yes/No

Select Yes if the field map is based on EPI data or No otherwise. Most scanner vendor field map sequences are non-EPI.

Polarity of phase-encode blips - +ve/-ve

Select +ve or -ve blip direction.When images are acquired K-space can be traversed using positive or negative phase-encode blips. This direction will influence the geometric distortions in terms of whether the affected regions of the image are stretched or compressed.

The convention used to describe the direction of the k-space traversal is based on the coordinate system used by SPM. In this coordinate system, the phase encode direction corresponds with the y-direction and is defined as positive from the posterior to the anterior of the head. The x-direction is defined as positive from left to right and the z-direction is defined as positive from foot to head. The polarity of the phase-encode blips describes in which direction k-space is traversed along the y-axis with respect to the coordinate system described here.

Apply Jacobian modulation - Yes/No

Select Yes to do Jacobian Modulation to adjust the intensities of voxels that have been stretched or compressed. In general this is not recommended for unwarping EPI data at this stage.

Total EPI readout time (ms)

Enter the total time in ms for the readout of the EPI echo train which is typically 10s of ms. This is the time taken to acquire all of the phase encode steps required to cover k-space (ie one image slice). For example, if the EPI sequence has 64 phase encode steps, the total readout time is the time taken to acquire 64 echoes: total readout time = number of echoes $\times$ echo spacing. This time does not include i) the duration of the excitation, ii) the delay between the excitation and the start of the acquisition or iii) time for fat saturation.

Load EPI image

Select a sample EPI image in NIfTI format. This image is automatically unwarped using the VDM calculated with the current parameters. The warped and the unwarped image are displayed in the graphics window underneath the field map (middle rows, right, Figure 24.1).

Match VDM to EPI

Select this option to match the field map magnitude data to the EPI image before it is used to unwarp the EPI. In general, the field map data should be acquired so that it is as closely registered with the EPI data as possible but matching can be selected if required. If a precalculated field map was loaded then the user is prompted to select a magnitude image in the same space as the field map. If real and imaginary images were selected, the toolbox automatically creates a magnitude image from these images and saves it with the name mag_NAME-OF-FIRST-INPUT-IMAGE.img.

Write unwarped

Write unwarped EPI image with the filename `uNAME_OF_EPI.img`.

Load structural

Load a structural image for comparison with unwarped EPI. This is displayed in the graphics window below the other images (bottom row, right fig 1).

MatchStructural

Coregister the structural image to the unwarped EPI and write the resulting transformation matrix to the header of the selected structural image.

Help

Call `spm_help` to display `FieldMap.man`.

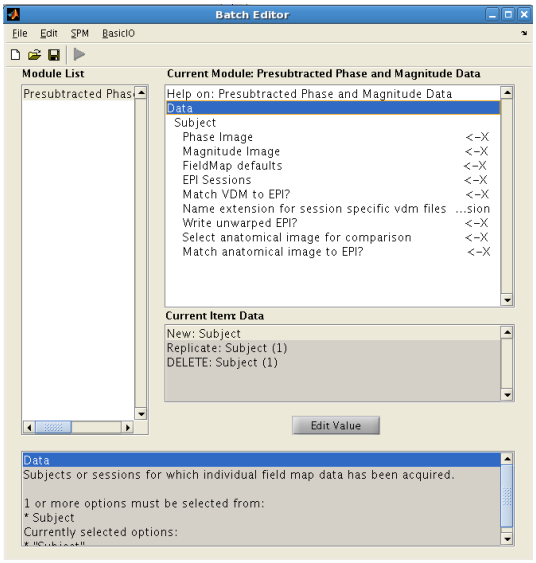


Figure 24.2: FieldMap using the SPM8 User Interface.

Quit

Quit the toolbox and closes all windows associated with it.

24.3 Using the pm_defaults file

Input parameters and the mode in which the toolbox works can be customised in the defaults file called pm_defaults.m. Different defaults files can be selected. This means that it is possible to have more than one set of default parameters, e.g. to accommodate different scanners or sequences. To be recognised by the FieldMap toolbox, these files must be named pm_defaults_NAME.m and have the same format as the existing pm_defaults.m. These defaults files can be loaded using the FieldMap GUI (as described in section 24.2.1) or selected when filling in the job fields using the SPM User-Interface (as described in section 6).

24.4 Using the SPM8 User Interface

FieldMap version 2.1 for SPM8 can be run using the SPM8 UI (Figure 24.2). To do this, from the top menu on the SPM8 Graphics window, select TASKS → SPM(interactive) → Tools → FieldMap. Select the type of FieldMap job to run from 1) Presubtracted Phase and Magntiude Data (this means a single phase and magnitude pair), 2) Real and Imaginary Data, 3) Phase and Magnitude Data (this means a double phase and magnitude pair) and 4) Precalculated FieldMap. In the Batch Editor, select New "Subject" in the lower panel, then select each item in the upper panel to fill in the different job fields. All fields marked with an X must be filled in. The other fields can be optionally filled or contain default values which can be changed if required.

24.5 Using FieldMap with multiple sessions or runs

If multiple EPI sessions or runs have been acquired, but with only a single set of fieldmap data, it is possible to automatically create a vdm file for each session. To do this, for each new each session under the input field EPI sessions, select the first EPI in each session. A copy of the VDM file will be matched to the first EPI in each session and saved with a separate name.

24.6 Using the FieldMap in Batch scripts

`FieldMap_preprocess.m` which calls `FieldMap_create.m` gives an example of how to run the FieldMap toolbox without using the GUI. To run the script, make sure your MATLAB path includes the directory where the FieldMap toolbox is installed. This can be done using the Set Path option under File in the MATLAB windows manager or using the command:

```
addpath /whatever/spm/toolbox/FieldMap
```

To run the FieldMap batch script, in MATLAB enter the following command:

```
VDM = FieldMap_preprocess(fm_dir,epi_dir, [te1, te2, epifm, tert, kdir, mask, match] );
```

where

fm_dir - name of directory containing fieldmap images.(e.g. fm_dir = '/path/study1/subj1/fieldmap')
epi_dir - name of directory containing epi images. (e.g. epi_dir = '/path/study1/subj1/images')
te1 - short echo time (in ms)
te2 - long echo time (in ms)
epifm - epi-based fieldmap - yes or no (1/0)
tert - total echo readout time (in ms)
kdir - blip direction (1/-1)
mask do brain segmentation to mask field map (1/0)
match match vdm file to first EPI in run (1/0).

NB: FieldMap will match the field map to the first epi image in the time series (after removing the dummy scans). Therefore, epi_dir must be the directory that contains the epi run that all other images will be realigned to.

The script will create an fpm* file, a vdm5_* file and an unwarped version of the EPI saved with the prescript "u".

24.7 Using the VDM file with Unwarp

In SPM, select the Realign & Unwarp option. For the input data called Phase map (vdm* file), select the vdm5_ or vdm5_ file for the subject and/or session. If you acquired more than one session (or run) of EPI images, you need to select a different vdm5_* file for each one. For more information about Unwarp see http://www.fil.ion.ucl.ac.uk/spm/toolbox/unwarp.

24.8 Appendices

24.8.1 Processing Hz field maps

Processing field maps involves a series of steps for which certain parameters in the spm_defaults file must be set.

1. If the acquired field map data comprises two complex images, the phase difference between them is calculated.

2. The phase map is unwrapped using the method specified by spm_def.UNWRAPPING_METHOD = 'Mark3D' or 'Mark2D' or 'Huttonish'. For a description of these different methods see spm_unwrap.m or FieldMap_principles.man. The default option is 'Mark3D'.

3. A mask is created so that unwrapping only occurs in regions where there is signal. If necessary, this mask can be expanded so that any voxel that hasn't been unwrapped and is less than spm_def.PAD/2 voxels away from an unwrapped one will be replaced by an average of the surrounding unwrapped voxels. This can be done by setting the parameter spm_def.PAD to a value greater than 0. The default value is 0 but a value ¿ 0 (eg 10) may be necessary if normal smoothing is chosen instead of weighted smoothing (as explained in the next step).

4. If required a mask can be generated to exclude regions of the fieldmap outside of the brain (in addition to the unwrapping mask described above). This step uses SPM segmentation for which the parameters in spm_def.MFLAGS can be set. For example, if the segmentation fails, (maybe because the fieldmap magnitude image doesn't have enough contrast), spm_def.MFLAGS.REG can be increased to say 0.05). The other parameters control morphological operations to generate a smooth brain mask and have been set empirically.

5. The unwrapped phase map is scaled by 1/(2*PI*difference in echo time) to convert it to Hz.

6. A weighted gaussian smoothing (weighted by the inverse of the noise) is performed on the unwrapped phase-map if the parameter spm_def.WS = 1. If spm_def.WS = 0, a normal smoothing is done. The weighted smoothing is particularly slow on large data sets ie high resolution. If field maps are acquired at high resolution then it is recommended to use spm_def.WS = 0 and do some padding of the intensity mask eg spm_def.PAD = 10. The size of the Gaussian filter used to implement either weighted or normal smoothing of the unwrapped maps is usually set to spm_def.FWHM = 10.

24.8.2 Converting Hz field map to VDM

1. The field map in Hz is multiplied by the total EPI readout time (in ms,) of the EPI image to be unwarped, resulting in a VDM. The readout time is specified by spm_def.TOTAL_EPI_READOUT_TIM (eg typically 10s of ms).The total EPI readout time is the time taken to acquire all of the phase encode steps required to cover k-space (ie one image slice). For example, if the EPI sequence has 64 phase encode steps, the total readout time is the time taken to acquire 64 echoes, e.g. total readout time = number of echoes × echo spacing. This time does not include i) the duration of of the excitation, ii) the delay between the excitation and the start of the acquisition or iii) time for fat saturation etc.

2. The VDM is multiplied by +/-1 to indicate whether the K-space traversal for the data acquisition has a +ve or -ve blip direction. This will ensure that the unwarping is performed in the correct direction and is specified by spm_def.K_SPACE_TRAVERSAL_BLIP_DIR = +/- 1.

3. The toolbox must know if the field map is based on an EPI or non-EPI acquisition. If using an EPI-based field map, the VDM must be inverted since the field map was acquired in warped space. This is specified by spm_def.EPI_BASED_FIELDMAPS = 1 or 0.

4. Jacobian Modulation can be applied to the unwarped EPI image. This modulates the intensity of the unwarped image so that in regions where voxels were compressed, the intensity is decresed and where voxels were stretched, the intensities are increased slightly. The modulation involves multiplying the unwarped EPI by 1 + the 1-d derivative of the VDM in the phase direction. An intensity adjustment of this nature may improve the coregistration results between an unwarped EPI and an undistorted image. This is specified by spm_def.DO_JACOBIAN_MODULATION = 0 or 1.

5. When any of the above conversion parameters are changed or a new EPI is selected, a new VDM is created and saved with the filename vdm5_NAME-OF-FIRST-INPUT-IMAGE.img. Any previous copy of the .img file is overwritten, but the corresponding .mat file is retained. It is done this way because the VDM may have already been coregiseterd to the EPI (as described below). Then, for an EPI-based VDM, the match between the VDM and the EPI will still be valid even if any of the above parameters have been changed. If the VDM is non-EPI-based and any of the above parameters are changed, the match between the VDM and the EPI may no longer be valid. In this case a warning is given to the user that it may be necessary to perform the coregistration again.

24.8.3 Matching field map data to EPI data

1. If required, the fieldmap can be matched to the EPI. This is done slightly differently depending on whether the field map is based on EPI or non-EPI data. If using an EPI field

map, the magnitude image is coregistered to the EPI. The resulting transformation matrix is used to sample the VDM file in the space of the EPI before unwarping.

2. If using a non-EPI field map, the VDM is used to forward warp the magnitude image which is then coregistered to the EPI. The forward warped image is saved with the filename wfmag_NAME-OF-FIRST-INPUT-IMAGE.img.

Chapter 25

New Segment

Contents

25.1 Data 176
25.1.1 Channel 176
25.2 Tissues 177
25.2.1 Tissue 177
25.3 Warping 178
25.3.1 Warping Regularisation 178
25.3.2 Affine Regularisation 179
25.3.3 Sampling distance 179
25.3.4 Deformation Fields 179

This toolbox is currently only work in progress, and is an extension of the default unified segmentation. The algorithm is essentially the same as that described in the Unified Segmentation paper, except for (i) a slightly different treatment of the mixing proportions, (ii) the use of an improved registration model, (iii) the ability to use multi-spectral data, (iv) an extended set of tissue probability maps, which allows a different treatment of voxels outside the brain. Some of the options in the toolbox do not yet work, and it has not yet been seamlessly integrated into the SPM8 software. Also, the extended tissue probability maps need further refinement. The current versions were crudely generated (by JA) using data that was kindly provided by Cynthia Jongen of the Imaging Sciences Institute at Utrecht, NL.

This function segments, bias corrects and spatially normalises - all in the same model [9]. Many investigators use tools within older versions of SPM for a technique that has become known as "optimised" voxel-based morphometry (VBM). VBM performs region-wise volumetric comparisons among populations of subjects. It requires the images to be spatially normalised, segmented into different tissue classes, and smoothed, prior to performing statistical tests [73, 56, 7, 8]. The "optimised" pre-processing strategy involved spatially normalising subjects' brain images to a standard space, by matching grey matter in these images, to a grey matter reference. The historical motivation behind this approach was to reduce the confounding effects of non-brain (e.g. scalp) structural variability on the registration. Tissue classification in older versions of SPM required the images to be registered with tissue probability maps. After registration, these maps represented the prior probability of different tissue classes being found at each location in an image. Bayes rule can then be used to combine these priors with tissue type probabilities derived from voxel intensities, to provide the posterior probability.

This procedure was inherently circular, because the registration required an initial tissue classification, and the tissue classification requires an initial registration. This circularity is resolved here by combining both components into a single generative model. This model also includes parameters that account for image intensity non-uniformity. Estimating the model parameters

(for a maximum a posteriori solution) involves alternating among classification, bias correction and registration steps. This approach provides better results than simple serial applications of each component.

25.1 Data

Specify the number of different channels (for multi-spectral classification). If you have scans of different contrasts for each of the subjects, then it is possible to combine the information from them in order to improve the segmentation accuracy. Note that only the first channel of data is used for the initial affine registration with the tissue probability maps.

25.1.1 Channel

Specify a channel for processing. If multiple channels are used (eg PD & T2), then the same order of subjects must be specified for each channel and they must be in register (same position, size, voxel dims etc..). The different channels can be treated differently in terms of inhomogeneity correction etc. You may wish to correct some channels and save the corrected images, whereas you may wish not to do this for other channels.

Volumes

Select scans from this channel for processing. If multiple channels are used (eg T1 & T2), then the same order of subjects must be specified for each channel and they must be in register (same position, size, voxel dims etc..).

Bias regularisation

MR images are usually corrupted by a smooth, spatially varying artifact that modulates the intensity of the image (bias). These artifacts, although not usually a problem for visual inspection, can impede automated processing of the images.

An important issue relates to the distinction between intensity variations that arise because of bias artifact due to the physics of MR scanning, and those that arise due to different tissue properties. The objective is to model the latter by different tissue classes, while modelling the former with a bias field. We know a priori that intensity variations due to MR physics tend to be spatially smooth, whereas those due to different tissue types tend to contain more high frequency information. A more accurate estimate of a bias field can be obtained by including prior knowledge about the distribution of the fields likely to be encountered by the correction algorithm. For example, if it is known that there is little or no intensity non-uniformity, then it would be wise to penalise large values for the intensity non-uniformity parameters. This regularisation can be placed within a Bayesian context, whereby the penalty incurred is the negative logarithm of a prior probability for any particular pattern of non-uniformity.

Knowing what works best should be a matter of empirical exploration. For example, if your data has very little intensity non-uniformity artifact, then the bias regularisation should be increased. This effectively tells the algorithm that there is very little bias in your data, so it does not try to model it.

Bias FWHM

FWHM of Gaussian smoothness of bias. If your intensity non-uniformity is very smooth, then choose a large FWHM. This will prevent the algorithm from trying to model out intensity variation due to different tissue types. The model for intensity non-uniformity is one of i.i.d. Gaussian noise that has been smoothed by some amount, before taking the exponential. Note also that smoother bias fields need fewer parameters to describe them. This means that the algorithm is faster for smoother intensity non-uniformities.

Save Bias Corrected

This is the option to save a bias corrected version of your images from this channel, or/and the estimated bias field. MR images are usually corrupted by a smooth, spatially varying artifact that modulates the intensity of the image (bias). These artifacts, although not usually a problem for visual inspection, can impede automated processing of the images. The bias corrected version should have more uniform intensities within the different types of tissues.

25.2 Tissues

The data for each subject are classified into a number of different tissue types. The tissue types are defined according to tissue probability maps, which define the prior probability of finding a tissue type at a particular location. Typically, the order of tissues is grey matter, white matter, CSF, bone, soft tissue and air/background (if using toolbox/Seg/TPM.nii).

25.2.1 Tissue

A number of options are available for each of the tissues. You may wish to save images of some tissues, but not others. If planning to use DARTEL, then make sure you generate "imported" tissue class images of grey and white matter (and possibly others). Different numbers of Gaussians may be needed to model the intensity distributions of the various tissues.

Tissue probability map

Select the tissue probability image for this class. These should be maps of eg grey matter, white matter or cerebro-spinal fluid probability. A nonlinear deformation field is estimated that best overlays the tissue probability maps on the individual subjects' image. The default tissue probability maps are modified versions of the ICBM Tissue Probabilistic Atlases.These tissue probability maps are kindly provided by the International Consortium for Brain Mapping, John C. Mazziotta and Arthur W. Toga. http://www.loni.ucla.edu/ICBM/ICBM_TissueProb.html. The original data are derived from 452 T1-weighted scans, which were aligned with an atlas space, corrected for scan inhomogeneities, and classified into grey matter, white matter and cerebrospinal fluid. These data were then affine registered to the MNI space and down-sampled to 2mm resolution.

Rather than assuming stationary prior probabilities based upon mixing proportions, additional information is used, based on other subjects' brain images. Priors are usually generated by registering a large number of subjects together, assigning voxels to different tissue types and averaging tissue classes over subjects. Three tissue classes are used: grey matter, white matter and cerebro-spinal fluid. A fourth class is also used, which is simply one minus the sum of the first three. These maps give the prior probability of any voxel in a registered image being of any of the tissue classes - irrespective of its intensity.

The model is refined further by allowing the tissue probability maps to be deformed according to a set of estimated parameters. This allows spatial normalisation and segmentation to be combined into the same model.

Num. Gaussians

The number of Gaussians used to represent the intensity distribution for each tissue class can be greater than one. In other words, a tissue probability map may be shared by several clusters. The assumption of a single Gaussian distribution for each class does not hold for a number of reasons. In particular, a voxel may not be purely of one tissue type, and instead contain signal from a number of different tissues (partial volume effects). Some partial volume voxels could fall at the interface between different classes, or they may fall in the middle of structures such as the thalamus, which may be considered as being either grey or white matter. Various other image segmentation approaches use additional clusters to model such partial volume effects. These generally assume that a pure tissue class has a Gaussian intensity distribution, whereas intensity distributions for partial volume voxels are broader, falling between the intensities of the pure

classes. Unlike these partial volume segmentation approaches, the model adopted here simply assumes that the intensity distribution of each class may not be Gaussian, and assigns belonging probabilities according to these non-Gaussian distributions. Typical numbers of Gaussians could be two for grey matter, two for white matter, two for CSF, three for bone, four for other soft tissues and two for air (background).

Note that if any of the Num. Gaussians is set to non-parametric, then a non-parametric approach will be used to model the tissue intensities. This may work for some images (eg CT), but not others - and it has not been optimised for multi-channel data. Note that it is likely to be especially problematic for images with poorly behaved intensity histograms due to aliasing effects that arise from having discrete values on the images.

Native Tissue

The native space option allows you to produce a tissue class image (c*) that is in alignment with the original (see Figure 5.1). It can also be used for "importing" into a form that can be used with the DARTEL toolbox (rc*).

Warped Tissue

You can produce spatially normalised versions of the tissue class - both with (mwc*) and without (wc*) modulation (see below). These can be used for voxel-based morphometry. All you need to do is smooth them and do the stats.

"Modulation" is to compensate for the effect of spatial normalisation. When warping a series of images to match a template, it is inevitable that volumetric differences will be introduced into the warped images. For example, if one subject's temporal lobe has half the volume of that of the template, then its volume will be doubled during spatial normalisation. This will also result in a doubling of the voxels labelled grey matter. In order to remove this confound, the spatially normalised grey matter (or other tissue class) is adjusted by multiplying by its relative volume before and after warping. If warping results in a region doubling its volume, then the correction will halve the intensity of the tissue label. This whole procedure has the effect of preserving the total amount of grey matter signal in the normalised partitions. Actually, in this version of SPM the warped data are not scaled by the Jacobian determinants when generating the "modulated" data. Instead, the original voxels are projected into their new location in the warped images. This exactly preserves the tissue count, but has the effect of introducing aliasing artifacts - especially if the original data are at a lower resolution than the warped images. Smoothing should reduce this artifact though.

Note also that the "unmodulated" data are generated slightly differently in this version of SPM. In this version, the projected data are corrected using a kind of smoothing procedure. This is not done exactly as it should be done (to save computational time), but it does a reasonable job. It also has the effect of extrapolating the warped tissue class images beyond the range of the original data. This extrapolation is not perfect, as it is only an estimate, but it may still be a good thing to do.

25.3 Warping

A number of warping options are provided, but the main one that you could consider changing is the one for specifying whether deformation fields or inverse deformation fields should be generated.

25.3.1 Warping Regularisation

The objective function for registering the tissue probability maps to the image to process, involves minimising the sum of two terms. One term gives a function of how probable the data is given the warping parameters. The other is a function of how probable the parameters are, and provides a penalty for unlikely deformations. Smoother deformations are deemed to be more probable. The amount of regularisation determines the tradeoff between the terms. Pick a value around one. However, if your normalised images appear distorted, then it may be an idea to increase the amount of regularisation (by an order of magnitude). More regularisation gives

smoother deformations, where the smoothness measure is determined by the bending energy of the deformations.

25.3.2 Affine Regularisation

The procedure is a local optimisation, so it needs reasonable initial starting estimates. Images should be placed in approximate alignment using the Display function of SPM before beginning. A Mutual Information affine registration with the tissue probability maps (D'Agostino et al, 2004) is used to achieve approximate alignment. Note that this step does not include any model for intensity non-uniformity. This means that if the procedure is to be initialised with the affine registration, then the data should not be too corrupted with this artifact.If there is a lot of intensity non-uniformity, then manually position your image in order to achieve closer starting estimates, and turn off the affine registration.

Affine registration into a standard space can be made more robust by regularisation (penalising excessive stretching or shrinking). The best solutions can be obtained by knowing the approximate amount of stretching that is needed (e.g. ICBM templates are slightly bigger than typical brains, so greater zooms are likely to be needed). For example, if registering to an image in ICBM/MNI space, then choose this option. If registering to a template that is close in size, then select the appropriate option for this.

25.3.3 Sampling distance

This encodes the approximate distance between sampled points when estimating the model parameters. Smaller values use more of the data, but the procedure is slower and needs more memory. Determining the "best" setting involves a compromise between speed and accuracy.

25.3.4 Deformation Fields

Deformation fields can be saved to disk, and used by the Deformations Utility. For spatially normalising images to MNI space, you will need the forward deformation, whereas for spatially normalising (eg) GIFTI surface files, you'll need the inverse. It is also possible to transform data in MNI space on to the individual subject, which also requires the inverse transform. Deformations are saved as .nii files, which contain three volumes to encode the x, y and z coordinates.

Chapter 26

DARTEL Tools

Contents

26.1 Initial Import . . . 182
26.1.1 Parameter Files . . . 182
26.1.2 Output Directory . . . 183
26.1.3 Bounding box . . . 183
26.1.4 Voxel size . . . 183
26.1.5 Image option . . . 183
26.1.6 Grey Matter . . . 183
26.1.7 White Matter . . . 183
26.1.8 CSF . . . 183
26.2 Run DARTEL (create Templates) . . . 183
26.2.1 Images . . . 183
26.2.2 Settings . . . 183
26.3 Run DARTEL (existing Templates) . . . 185
26.3.1 Images . . . 185
26.3.2 Settings . . . 185
26.4 Normalise to MNI Space . . . 186
26.4.1 DARTEL Template . . . 186
26.4.2 Select according to . . . 186
26.4.3 Voxel sizes . . . 187
26.4.4 Bounding box . . . 187
26.4.5 Preserve . . . 187
26.4.6 Gaussian FWHM . . . 187
26.5 Create Warped . . . 187
26.5.1 Flow fields . . . 188
26.5.2 Images . . . 188
26.5.3 Modulation . . . 188
26.5.4 Time Steps . . . 188
26.5.5 Interpolation . . . 188
26.6 Jacobian determinants . . . 188
26.6.1 Flow fields . . . 188
26.6.2 Time Steps . . . 188
26.7 Create Inverse Warped . . . 188
26.7.1 Flow fields . . . 189
26.7.2 Images . . . 189
26.7.3 Time Steps . . . 189
26.7.4 Interpolation . . . 189
26.8 Kernel Utilities . . . 189

26.8.1 Kernel from Images 189
26.8.2 Kernel from Flows 190

This toolbox is based around the "A Fast Diffeomorphic Registration Algorithm" paper [3] . The idea is to register images by computing a "flow field", which can then be "exponentiated" to generate both forward and backward deformations. Currently, the software only works with images that have isotropic voxels, identical dimensions and which are in approximate alignment with each other. One of the reasons for this is that the approach assumes circulant boundary conditions, which makes modelling global rotations impossible. Another reason why the images should be approximately aligned is because there are interactions among the transformations that are minimised by beginning with images that are already almost in register. This problem could be alleviated by a time varying flow field, but this is currently computationally impractical.

Because of these limitations, images should first be imported. This involves taking the "*_seg_sn.mat" files produced by the segmentation code of SPM5, and writing out rigidly transformed versions of the tissue class images, such that they are in as close alignment as possible with the tissue probability maps. Rigidly transformed original images can also be generated, with the option to have skull-stripped versions.

The next step is the registration itself. This can involve matching single images together, or it can involve the simultaneous registration of e.g. GM with GM, WM with WM and 1-(GM+WM) with 1-(GM+WM) (when needed, the 1-(GM+WM) class is generated implicitly, so there is no need to include this class yourself). This procedure begins by creating a mean of all the images, which is used as an initial template. Deformations from this template to each of the individual images are computed, and the template is then re-generated by applying the inverses of the deformations to the images and averaging. This procedure is repeated a number of times.

Finally, warped versions of the images (or other images that are in alignment with them) can be generated.

This toolbox is not yet seamlessly integrated into the SPM package. Eventually, the plan is to use many of the ideas here as the default strategy for spatial normalisation. The toolbox may change with future updates. There will also be a number of other (as yet unspecified) extensions, which may include a variable velocity version (related to LDDMM). Note that the Fast Diffeomorphism paper only describes a sum of squares objective function. The multinomial objective function is an extension, based on a more appropriate model for aligning binary data to a template.

26.1 Initial Import

Images first need to be imported into a form that DARTEL can work with. If the default segmentation is used (ie the Segment button), then this involves taking the results of the segmentation (*_seg_sn.mat) [9] , in order to have rigidly aligned tissue class images. Typically, there would be imported grey matter and white matter images, but CSF images can also be included. The subsequent DARTEL alignment will then attempt to nonlinearly register these tissue class images together. If the new segmentation routine is used (from the toolbox), then this includes the option to generate "imported" tissue class images. This means that a seperate importing step is not needed for it.

26.1.1 Parameter Files

Select '_sn.mat' files containing the spatial transformation and segmentation parameters. Rigidly aligned versions of the image that was segmented will be generated. The image files used by the segmentation may have moved. If they have, then (so the import can find them) ensure that they are either in the output directory, or the current working directory.

26.1.2 Output Directory

Select the directory where the resliced files should be written.

26.1.3 Bounding box

The bounding box (in mm) of the volume that is to be written (relative to the anterior commissure). Non-finite values will be replaced by the bounding box of the tissue probability maps used in the segmentation.

26.1.4 Voxel size

The (isotropic) voxel sizes of the written images. A non-finite value will be replaced by the average voxel size of the tissue probability maps used by the segmentation.

26.1.5 Image option

A resliced version of the original image can be produced, which may have various procedures applied to it. All options will rescale the images so that the mean of the white matter intensity is set to one. The "skull stripped" versions are the images simply scaled by the sum of the grey and white matter probabilities.

26.1.6 Grey Matter

Produce a resliced version of this tissue class?

26.1.7 White Matter

Produce a resliced version of this tissue class?

26.1.8 CSF

Produce a resliced version of this tissue class?

26.2 Run DARTEL (create Templates)

Run the DARTEL nonlinear image registration procedure. This involves iteratively matching all the selected images to a template generated from their own mean. A series of Template*.nii files are generated, which become increasingly crisp as the registration proceeds.

26.2.1 Images

Select the images to be warped together. Multiple sets of images can be simultaneously registered. For example, the first set may be a bunch of grey matter images, and the second set may be the white matter images of the same subjects.

Images

Select a set of imported images of the same type to be registered by minimising a measure of difference from the template.

26.2.2 Settings

Various settings for the optimisation. The default values should work reasonably well for aligning tissue class images together.

Template basename

Enter the base for the template name. Templates generated at each outer iteration of the procedure will be basename_1.nii, basename_2.nii etc. If empty, then no template will be saved. Similarly, the estimated flow-fields will have the basename appended to them.

Regularisation Form

The registration is penalised by some "energy" term. Here, the form of this energy term is specified. Three different forms of regularisation can currently be used.

Outer Iterations

The images are averaged, and each individual image is warped to match this average. This is repeated a number of times.

Outer Iteration Different parameters can be specified for each outer iteration. Each of them warps the images to the template, and then regenerates the template from the average of the warped images. Multiple outer iterations should be used for more accurate results, beginning with a more coarse registration (more regularisation) then ending with the more detailed registration (less regularisation).

Inner Iterations The number of Gauss-Newton iterations to be done within this outer iteration. After this, new average(s) are created, which the individual images are warped to match.

Reg params For linear elasticity, the parameters are mu, lambda and id. For membrane energy, the parameters are lambda, unused and id.id is a term for penalising absolute displacements, and should therefore be small. For bending energy, the parameters are lambda, id1 and id2, and the regularisation is by (-lambda*Laplacian + id1)$\hat{2}$ + id2.

Use more regularisation for the early iterations so that the deformations are smooth, and then use less for the later ones so that the details can be better matched.

Time Steps The number of time points used for solving the partial differential equations. A single time point would be equivalent to a small deformation model. Smaller values allow faster computations, but are less accurate in terms of inverse consistency and may result in the one-to-one mapping breaking down. Earlier iteration could use fewer time points, but later ones should use about 64 (or fewer if the deformations are very smooth).

Smoothing Parameter A LogOdds parameterisation of the template is smoothed using a multi-grid scheme. The amount of smoothing is determined by this parameter.

Optimisation Settings

Settings for the optimisation. If you are unsure about them, then leave them at the default values. Optimisation is by repeating a number of Levenberg-Marquardt iterations, in which the equations are solved using a full multi-grid (FMG) scheme. FMG and Levenberg-Marquardt are both described in Numerical Recipes (2nd edition).

LM Regularisation Levenberg-Marquardt regularisation. Larger values increase the the stability of the optimisation, but slow it down. A value of zero results in a Gauss-Newton strategy, but this is not recommended as it may result in instabilities in the FMG.

Cycles Number of cycles used by the full multi-grid matrix solver. More cycles result in higher accuracy, but slow down the algorithm. See Numerical Recipes for more information on multi-grid methods.

Iterations Number of relaxation iterations performed in each multi-grid cycle. More iterations are needed if using "bending energy" regularisation, because the relaxation scheme only runs very slowly. See the chapter on solving partial differential equations in Numerical Recipes for more information about relaxation methods.

26.3 Run DARTEL (existing Templates)

Run the DARTEL nonlinear image registration procedure to match individual images to pre-existing template data. Start out with smooth templates, and select crisp templates for the later iterations.

26.3.1 Images

Select the images to be warped together. Multiple sets of images can be simultaneously registered. For example, the first set may be a bunch of grey matter images, and the second set may be the white matter images of the same subjects.

Images

Select a set of imported images of the same type to be registered by minimising a measure of difference from the template.

26.3.2 Settings

Various settings for the optimisation. The default values should work reasonably well for aligning tissue class images together.

Regularisation Form

The registration is penalised by some "energy" term. Here, the form of this energy term is specified. Three different forms of regularisation can currently be used.

Outer Iterations

The images are warped to match a sequence of templates. Early iterations should ideally use smoother templates and more regularisation than later iterations.

Outer Iteration Different parameters and templates can be specified for each outer iteration.

Inner Iterations The number of Gauss-Newton iterations to be done within this outer iteration.

Reg params For linear elasticity, the parameters are mu, lambda and id. For membrane energy, the parameters are lambda, unused and id.id is a term for penalising absolute displacements, and should therefore be small. For bending energy, the parameters are lambda, id1 and id2, and the regularisation is by (-lambda*Laplacian + id1)$\hat{2}$ + id2.

Use more regularisation for the early iterations so that the deformations are smooth, and then use less for the later ones so that the details can be better matched.

Time Steps The number of time points used for solving the partial differential equations. A single time point would be equivalent to a small deformation model. Smaller values allow faster computations, but are less accurate in terms of inverse consistency and may result in the one-to-one mapping breaking down. Earlier iteration could use fewer time points, but later ones should use about 64 (or fewer if the deformations are very smooth).

Template Select template. Smoother templates should be used for the early iterations. Note that the template should be a 4D file, with the 4th dimension equal to the number of sets of images.

Optimisation Settings

Settings for the optimisation. If you are unsure about them, then leave them at the default values. Optimisation is by repeating a number of Levenberg-Marquardt iterations, in which the equations are solved using a full multi-grid (FMG) scheme. FMG and Levenberg-Marquardt are both described in Numerical Recipes (2nd edition).

LM Regularisation Levenberg-Marquardt regularisation. Larger values increase the the stability of the optimisation, but slow it down. A value of zero results in a Gauss-Newton strategy, but this is not recommended as it may result in instabilities in the FMG.

Cycles Number of cycles used by the full multi-grid matrix solver. More cycles result in higher accuracy, but slow down the algorithm. See Numerical Recipes for more information on multi-grid methods.

Iterations Number of relaxation iterations performed in each multi-grid cycle. More iterations are needed if using "bending energy" regularisation, because the relaxation scheme only runs very slowly. See the chapter on solving partial differential equations in Numerical Recipes for more information about relaxation methods.

26.4 Normalise to MNI Space

Normally, DARTEL generates warped images that align with the average-shaped template. This routine includes an initial affine regisration of the template (the final one generated by DARTEL), with the TPM data released with SPM.

"Smoothed" (blurred) spatially normalised images are generated in such a way that the original signal is preserved. Normalised images are generated by a "pushing" rather than a "pulling" (the usual) procedure. Note that trilinear interpolation is used, and no masking is done. It is therefore essential that the images are realigned and resliced before they are spatially normalised. Alternatively, contrast images generated from unsmoothed native-space fMRI/PET data can be spatially normalised for a 2nd level analysis.

Two "preserve" options are provided. One of them should do the equavalent of generating smoothed "modulated" spatially normalised images. The other does the equivalent of smoothing the modulated normalised fMRI/PET, and dividing by the smoothed Jacobian determinants.

26.4.1 DARTEL Template

Select the final Template file generated by DARTEL. This will be affine registered with a TPM file, such that the resulting spatially normalised images are closer aligned to MNI space. Leave empty if you do not wish to incorporate a transform to MNI space (ie just click "done' on the file selector, without selecting any images).

26.4.2 Select according to

You may wish to spatially normalise only a few subjects, but have many scans per subject (eg for fMRI), or you may have lots of subjects, but with a small and fixed number of scans for each of them (eg for VBM). The idea is to chose the way of selecting files that is easier.

Few Subjects

Select this option if there are only a few subjects, each with many or a variable number of scans each. You will then need to specify a series of subjects, and the flow field and images of each of them.

Subject Subject to be spatially normalized.

Flow Field DARTEL flow field for this subject.

Images Images for this subject to spatially normalise.

Many Subjects

Select this option if you have many subjects to spatially normalise, but there are a small and fixed number of scans for each subject.

Flow fields The flow fields store the deformation information. The same fields can be used for both forward or backward deformations (or even, in principle, half way or exaggerated deformations).

Images The flow field deformations can be applied to multiple images. At this point, you are choosing how many images each flow field should be applied to.

Images Select images to be warped. Note that there should be the same number of images as there are flow fields, such that each flow field warps one image.

26.4.3 Voxel sizes

Specify the voxel sizes of the deformation field to be produced. Non-finite values will default to the voxel sizes of the template imagethat was originally used to estimate the deformation.

26.4.4 Bounding box

Specify the bounding box of the deformation field to be produced. Non-finite values will default to the bounding box of the template imagethat was originally used to estimate the deformation.

26.4.5 Preserve

Preserve Concentrations: Smoothed spatially normalised images (sw*) represent weighted averages of the signal under the smoothing kernel, approximately preserving the intensities of the original images. This option is currently suggested for eg fMRI.

Preserve Total: Smoothed and spatially normalised images preserve the total amount of signal from each region in the images (smw*). Areas that are expanded during warping are correspondingly reduced in intensity. This option is suggested for VBM.

26.4.6 Gaussian FWHM

Specify the full-width at half maximum (FWHM) of the Gaussian blurring kernel in mm. Three values should be entered, denoting the FWHM in the x, y and z directions. Note that you can also specify [0 0 0], but any "modulated' data will show aliasing (see eg Wikipedia), which occurs because of the way the warped images are generated.

26.5 Create Warped

This allows spatially normalised images to be generated. Note that voxel sizes and bounding boxes can not be adjusted, and that there may be strange effects due to the boundary conditions used by the warping. Also note that the warped images are not in Talairach or MNI space. The coordinate system is that of the average shape and size of the subjects to which DARTEL was applied. In order to have MNI-space normalised images, then the Deformations Utility can be used to compose the individual DARTEL warps, with a deformation field that matches (e.g.) the Template grey matter generated by DARTEL, with one of the grey matter volumes released with SPM.

26.5.1 Flow fields

The flow fields store the deformation information. The same fields can be used for both forward or backward deformations (or even, in principle, half way or exaggerated deformations).

26.5.2 Images

The flow field deformations can be applied to multiple images. At this point, you are choosing how many images each flow field should be applied to.

Images

Select images to be warped. Note that there should be the same number of images as there are flow fields, such that each flow field warps one image.

26.5.3 Modulation

This allows the spatially normalised images to be rescaled by the Jacobian determinants of the deformations. Note that the rescaling is only approximate for deformations generated using smaller numbers of time steps.

26.5.4 Time Steps

The number of time points used for solving the partial differential equations. Note that Jacobian determinants are not very accurate for very small numbers of time steps (less than about 16).

26.5.5 Interpolation

The method by which the images are sampled when being written in a different space.

Nearest Neighbour: - Fastest, but not normally recommended.

Bilinear Interpolation: - OK for PET, or realigned fMRI.

B-spline Interpolation: - Better quality (but slower) interpolation [69], especially with higher degree splines. Do not use B-splines when there is any region of NaN or Inf in the images.

26.6 Jacobian determinants

Create Jacobian determinant fields from flowfields.

26.6.1 Flow fields

The flow fields store the deformation information. The same fields can be used for both forward or backward deformations (or even, in principle, half way or exaggerated deformations).

26.6.2 Time Steps

The number of time points used for solving the partial differential equations. Note that Jacobian determinants are not very accurate for very small numbers of time steps (less than about 16).

26.7 Create Inverse Warped

Create inverse normalised versions of some image(s). The image that is inverse-normalised should be in alignment with the template (generated during the warping procedure). Note that the results have the same dimensions as the "flow fields", but are mapped to the original images via the affine transformations in their headers.

26.7.1 Flow fields

The flow fields store the deformation information. The same fields can be used for both forward or backward deformations (or even, in principle, half way or exaggerated deformations).

26.7.2 Images

Select the image(s) to be inverse normalised. These should be in alignment with the template image generated by the warping procedure.

26.7.3 Time Steps

The number of time points used for solving the partial differential equations. Note that Jacobian determinants are not very accurate for very small numbers of time steps (less than about 16).

26.7.4 Interpolation

The method by which the images are sampled when being written in a different space.

Nearest Neighbour: - Fastest, but not normally recommended.

Bilinear Interpolation: - OK for PET, or realigned fMRI.

B-spline Interpolation: - Better quality (but slower) interpolation [69], especially with higher degree splines. Do not use B-splines when there is any region of NaN or Inf in the images.

26.8 Kernel Utilities

DARTEL can be used for generating matrices of dot-products for various kernel pattern-recognition procedures.

The idea of applying pattern-recognition procedures is to obtain a multi-variate characterisation of the anatomical differences among groups of subjects. These characterisations can then be used to separate (eg) healthy individuals from particular patient populations. There is still a great deal of methodological work to be done, so the types of kernel that can be generated here are unlikely to be the definitive ways of proceeding. They are only just a few ideas that may be worth trying out. The idea is simply to attempt a vaguely principled way to combine generative models with discriminative models (see the "Pattern Recognition and Machine Learning" book by Chris Bishop for more ideas). Better ways (higher predictive accuracy) will eventually emerge.

Various pattern recognition algorithms are available freely over the Internet. Possible approaches include Support-Vector Machines, Relevance-Vector machines and Gaussian Process Models. Gaussian Process Models probably give the most accurate probabilistic predictions, and allow kernels generated from different pieces of data to be most easily combined.

26.8.1 Kernel from Images

Generate a kernel matrix from images. In principle, this same function could be used for generating kernels from any image data (e.g. "modulated" grey matter). If there is prior knowledge about some region providing more predictive information (e.g. the hippocampi for AD), then it is possible to weight the generation of the kernel accordingly. The matrix of dot-products is saved in a variable "Phi", which can be loaded from the dp_*.mat file. The "kernel trick" can be used to convert these dot-products into distance measures for e.g. radial basis-function approaches.

Data

Select images to generate dot-products from.

Weighting image

The kernel can be generated so that some voxels contribute to the similarity measures more than others. This is achieved by supplying a weighting image, which each of the component images are multiplied before the dot-products are computed. This image needs to have the same dimensions as the component images, but orientation information (encoded by matrices in the headers) is ignored. If left empty, then all voxels are weighted equally.

Dot-product Filename

Enter a filename for results (it will be prefixed by "dp_" and saved in the current directory).

26.8.2 Kernel from Flows

Generate a kernel from flow fields. The dot-products are saved in a variable "Phi" in the resulting dp_*.mat file.

Flow fields

Select the flow fields for each subject.

Regularisation Form

The registration is penalised by some "energy" term. Here, the form of this energy term is specified. Three different forms of regularisation can currently be used.

Reg params

For linear elasticity, the parameters are 'mu', 'lambda' and 'id'. For membrane and bending energy, the parameters are 'lambda', unused and 'id'. The term 'id' is for penalising absolute displacements, and should therefore be small.

Dot-product Filename

Enter a filename for results (it will be prefixed by "dp_" and saved in the current directory.

Chapter 27

High-Dimensional Warping

Contents

27.1 Subjects 191
27.1.1 Subject 191
27.2 Bias Correction Options 192
27.2.1 Iterations 192
27.2.2 Bias FWHM 192
27.2.3 Bias regularisation 192
27.2.4 Levenberg-Marquardt regularisation 192
27.3 Warping Options 192
27.3.1 Iterations 192
27.3.2 Warping regularisation 192

This toolbox is a Bayesian method for three dimensional registration of brain images [4] . A finite element approach is used to obtain a maximum a posteriori (MAP) estimate of the deformation field at every voxel of a template volume. The priors used by the MAP estimate penalize unlikely deformations and enforce a continuous one-to-one mapping. The deformations are assumed to have some form of symmetry, in that priors describing the probability distribution of the deformations should be identical to those for the inverses (i.e., warping brain A to brain B should not be different probablistically from warping B to A). A gradient descent algorithm is used to estimate the optimum deformations.

Deformation fields are written with the same name as the moved image, but with "y_" prefixed on to the filename. Jacobian determinant images are also written (prefixed by "jy_").

27.1 Subjects

Specify pairs of images to match together.

27.1.1 Subject

Two images of the same subject, which are to be registered together. Prior to nonlinear high-dimensional warping, the images should be rigidly registered with each other.

Reference Image

This is the reference image, which remains stationary.

Moved Image

This is the moved image, which is warped to match the reference.

27.2 Bias Correction Options

MR images are usually corrupted by a smooth, spatially varying artifact that modulates the intensity of the image (bias). These artifacts, although not usually a problem for visual inspection, can impede automated processing of the images.

Before registering the images, an approximate bias correction is estimated for the moved image. This is based on minimising the difference between the images an a symmetric way. Prior to registering the images, they should be rigidly aligned together. The bias correction is estimated once for these aligned images.

27.2.1 Iterations

Number of iterations for the bias correction

27.2.2 Bias FWHM

FWHM of Gaussian smoothness of bias. If your intensity nonuniformity is very smooth, then choose a large FWHM. This will prevent the algorithm from trying to model out intensity variation due to different tissue types. The model for intensity nonuniformity is one of i.i.d. Gaussian noise that has been smoothed by some amount, before taking the exponential. Note also that smoother bias fields need fewer parameters to describe them. This means that the algorithm is faster for smoother intensity nonuniformities.

27.2.3 Bias regularisation

We know a priori that intensity variations due to MR physics tend to be spatially smooth, whereas those due to different tissue types tend to contain more high frequency information. A more accurate estimate of a bias field can be obtained by including prior knowledge about the distribution of the fields likely to be encountered by the correction algorithm. For example, if it is known that there is little or no intensity non-uniformity, then it would be wise to penalise large values for the intensity nonuniformity parameters. This regularisation can be placed within a Bayesian context, whereby the penalty incurred is the negative logarithm of a prior probability for any particular pattern of nonuniformity.

27.2.4 Levenberg-Marquardt regularisation

Levenberg-Marquardt regularisation keeps the bias correction part stable. Higher values means more stability, but slower convergence.

27.3 Warping Options

There are a couple of user-customisable warping options.

27.3.1 Iterations

Number of iterations for the warping.

27.3.2 Warping regularisation

There is a tradeoff between the smoothness of the estimated warps, and the difference between the registered images. Higher values mean smoother warps, at the expense of a lower mean squared difference between the images.

Part VII

Data sets and examples

Chapter 28

Auditory fMRI data

This data set comprises whole brain BOLD/EPI images acquired on a modified 2T Siemens MAGNETOM Vision system. Each acquisition consisted of 64 contiguous slices (64×64×64 3×3mm×3 mm^3 voxels). Acquisition took 6.05s, with the scan to scan repeat time (TR) set arbitrarily to 7s.

96 acquisitions were made (TR=7s) from a single subject, in blocks of 6, giving 16 42s blocks. The condition for successive blocks alternated between rest and auditory stimulation, starting with rest. Auditory stimulation was bi-syllabic words presented binaurally at a rate of 60 per minute. The functional data starts at acquisition 4, image `fM00223_004`. Due to T1 effects it is advisable to discard the first few scans (there were no "dummy" lead-in scans). A structural image was also acquired: `sM00223_002`. These images are stored in Analyze format and are available from the SPM site [1]. This data set was the first ever collected and analysed in the Functional Imaging Laboratory (FIL) and is known locally as the mother of all experiments (MoAE).

To analyse the data, first create a new directory `DIR`, eg. `c:\data\auditory`, in which to place the results of your analysis. Then create 3 subdirectories (i) `jobs`, (ii) `classical` and (iii) `bayesian`. As the analysis proceeds these directories will be filled with job-specification files, design matrices and models estimated using classical or Bayesian methods.

Start up MATLAB enter your jobs directory and type `spm fmri` at the MATLAB prompt. SPM will then open in fMRI mode with three windows (1) the top-left or "Menu" window, (2) the bottom-left or "Interactive" window and (3) the right-hand or "Graphics" window. Analysis then takes place in three major stages (i) spatial pre-processing, (ii) model specification, review and estimation and (iii) inference. These stages organise the buttons in SPM's Menu window.

28.1 Spatial pre-processing

28.1.1 Realignment

Under the spatial pre-processing section of the SPM base window select textscRealign (Est & Res) from the REALIGN pulldown menu. This will call up a realignment job specification in the batch editor. Then

- Highlight data, select "New Session", then highlight the newly created "Session" option.
- Select "Specify Files" and use the SPM file selector to choose all of your functional images eg. "`fM000*.img`". There should be 96 files.
- Save the job file as eg. `DIR\jobs\realign.mat`.
- Press the `RUN` button in the batch editor (green arrow).

This will run the realign job which will write realigned images into the directory where the functional images are. These new images will be prefixed with the letter "`r`". SPM will then plot the estimated time series of translations and rotations shown in Figure 28.2. These data are also

[1] Auditory fMRI dataset: http://www.fil.ion.ucl.ac.uk/spm/data/auditory/

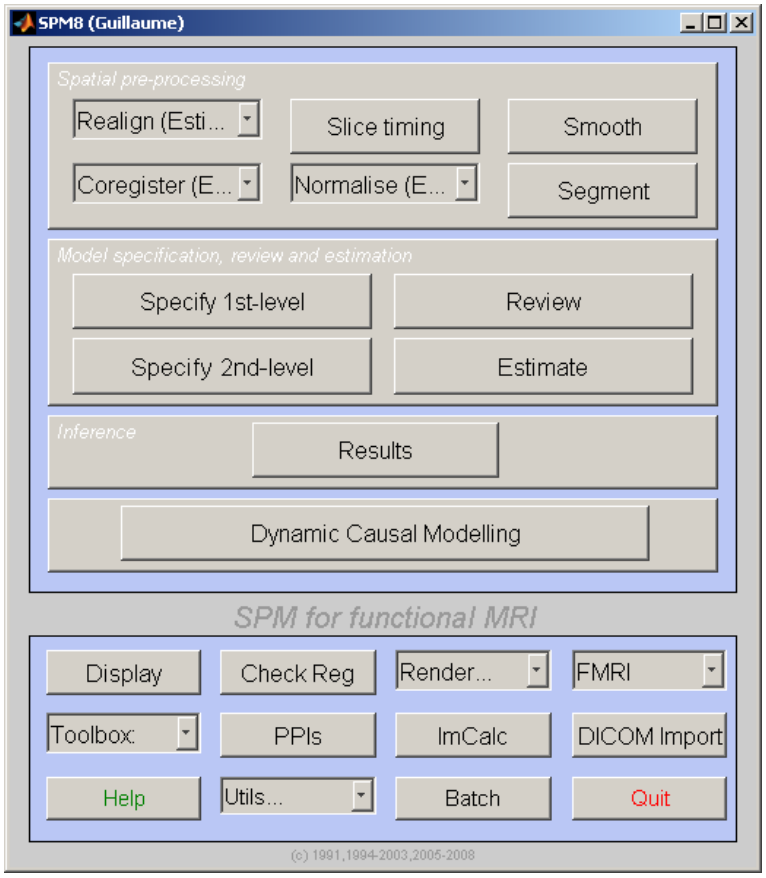


Figure 28.1: *The SPM base window comprises three sections i) spatial pre-processing, (ii) model specification, review and estimation and (iii) inference.*

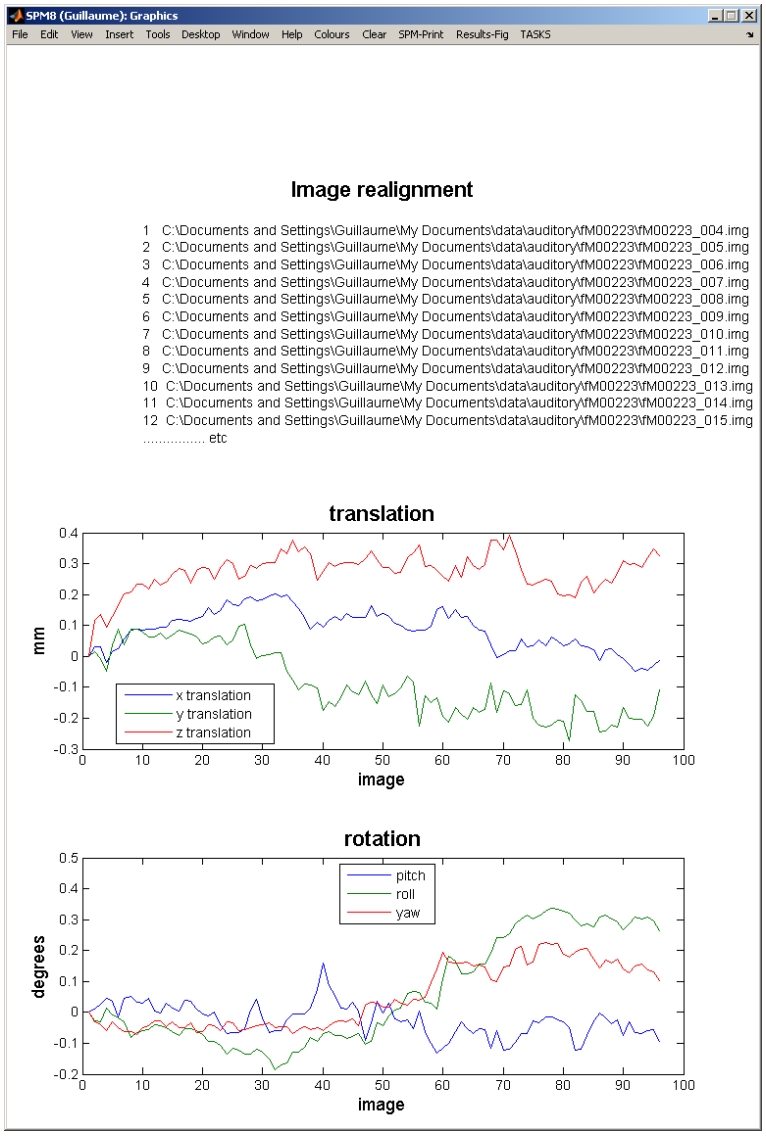


Figure 28.2: *Realignment of Auditory data.*

saved to a file eg. `rp_fM00223_004.txt`, so that these variables can be used as regressors when fitting GLMs. This allows movements effects to be discounted when looking for brain activations.

SPM will also create a mean image eg. `meanfM00223_004.img` which will be used in the next step of spatial processing - coregistration.

28.1.2 Coregistration

Select Coregister (Estimate) from the Coregister pulldown. This will call up the specification of a coregistration job in the batch editor.

- Highlight "Reference Image" and then select the mean fMRI scan from realignment eg. `meanfM00223_004.img`.
- Highlight "Source Image" and then select the structural image eg. `sM00223_002.img`.
- Press the Save button and save the job as `coreg.job`.
- Then press the `RUN` button.

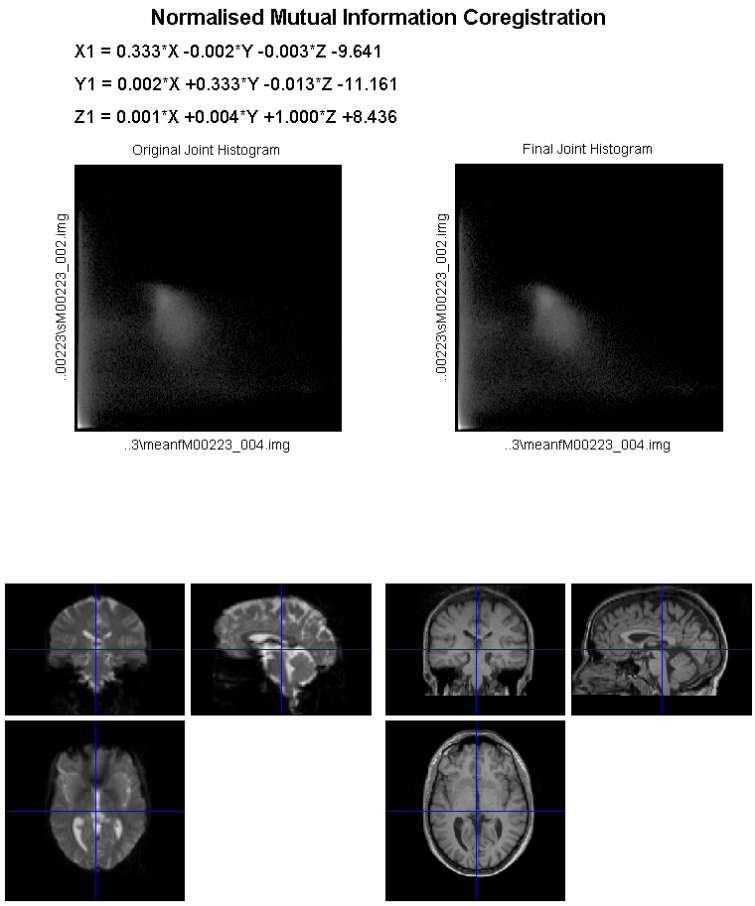


Figure 28.3: *Mutual Information Coregistration of Auditory data.*

SPM will then implement a coregistration between the structural and functional data that maximises the mutual information. The image in figure 28.3 should then appear in the graphics window. SPM will have changed the header of the source file which in this case is the structural image `sM00223_002.hdr`.

The CHECK REG facility is useful here, to check the results of coregistration. Press the CHECK REG button in the lower section of the base window and then the select the "Reference" and "Source" Images specified above ie `meanfM00223_004.img` and `sM00223_002.img`. SPM will then produce an image like that shown in Figure 28.4 in the Graphics window. You can then use your mouse to navigate these images to confirm that there is an anatomical correspondence.

28.1.3 Segmentation

Press the SEGMENT button. This will call up the specification of a segmentation job in the batch editor. Highlight the Data field and then select the subjects registered anatomical image eg. `sM00223_002.img`. Save the job file as `segment.mat` and then press `RUN`. SPM will segment the structural image using the default tissue probability maps as priors.

Faster, though perhaps less optimal results can be obtained by eg. reducing the number of Gaussians per class from [2 2 2 4] to eg. [1 1 1 4], increasing the sampling distance from eg. 3 to 4mm. These options can be edited under the "Custom" sub-menu and saved before the job is run. The results obtained in figure 28.5 were obtained using the default values.

SPM will create, by default, gray and white matter images and bias-field corrected structural image. These can be viewed using the CheckReg facility as described in the previous section. Figure 28.5 shows the gray matter image, `c1sM0023_002.img` along with the original structural.

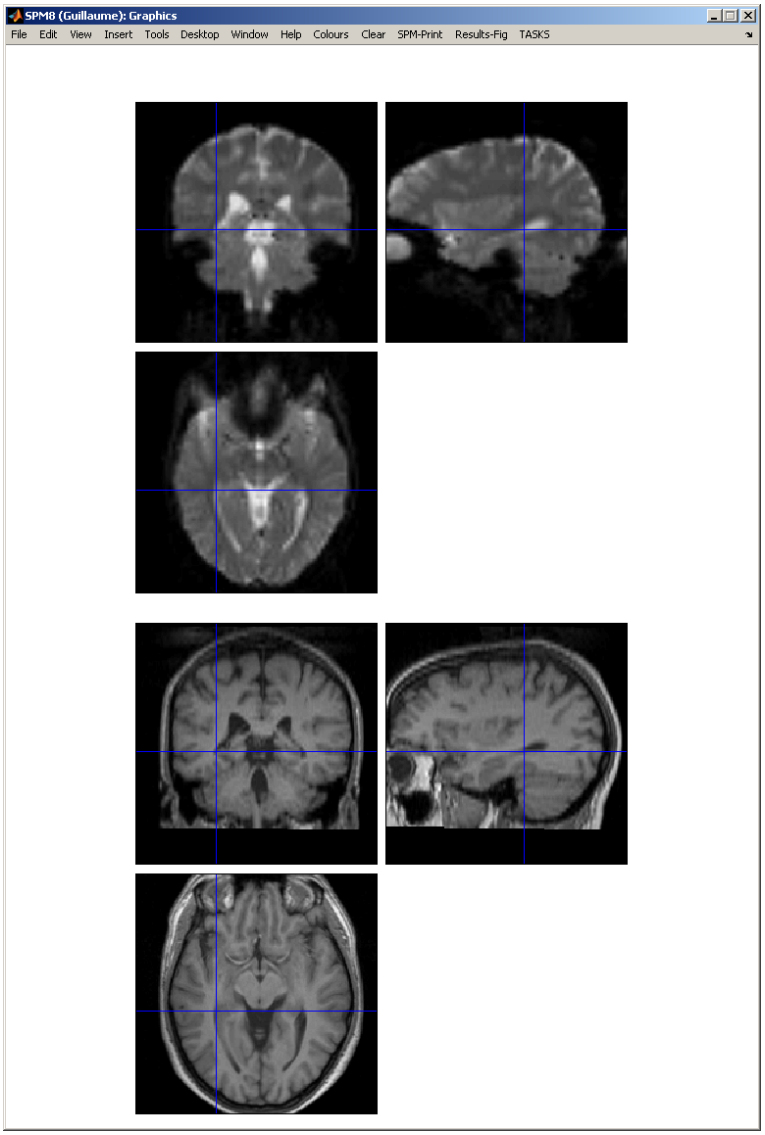


Figure 28.4: *Checking registration of functional and "registered" structural data.*

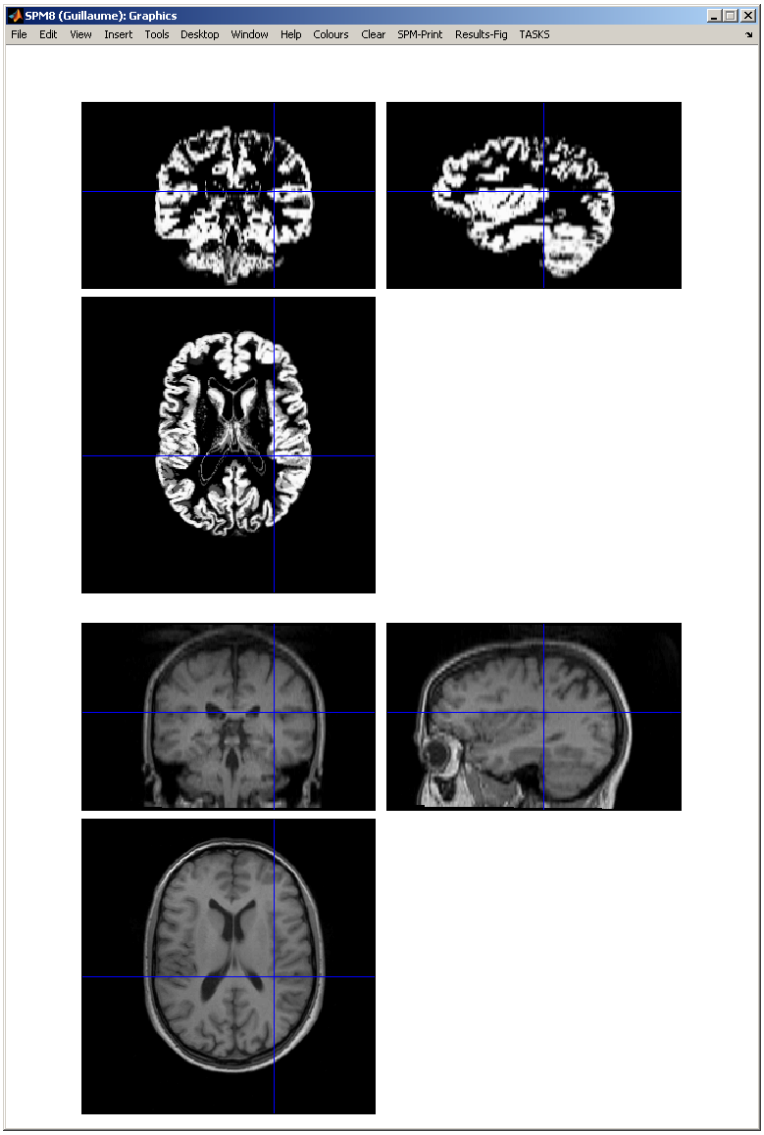


Figure 28.5: *Gray matter image and "registered" structural image.*

Figure 28.6 shows the structural and bias-corrected image, `msM0023_002.img`.

SPM will also write a spatial normalisation eg. `sM00223_0020_seg_sn.mat` and inverse spatial normalisation parameters `sM00223_0020_seg_inv_sn.mat` to files in the original structural directory. These can be used to normalise the functional data.

28.1.4 Normalise

Select NORMALISE (WRITE) from the NORMALISE pulldown menu. This will call up the specification of a normalise job in the batch editor.

- Highlight "Data", select New "Subject",
- Highlight "Parameter File" and select the `sM00223_0020_seg_sn.mat` file that you created in the previous section,
- Highlight images to write and select all of the realigned functional images `rfM000*.img`. Note: This can be done efficiently by changing the filter in the SPM file selector to `^r.*`. SPM will then only list those files beginning with the letter `r` ie. those that have been

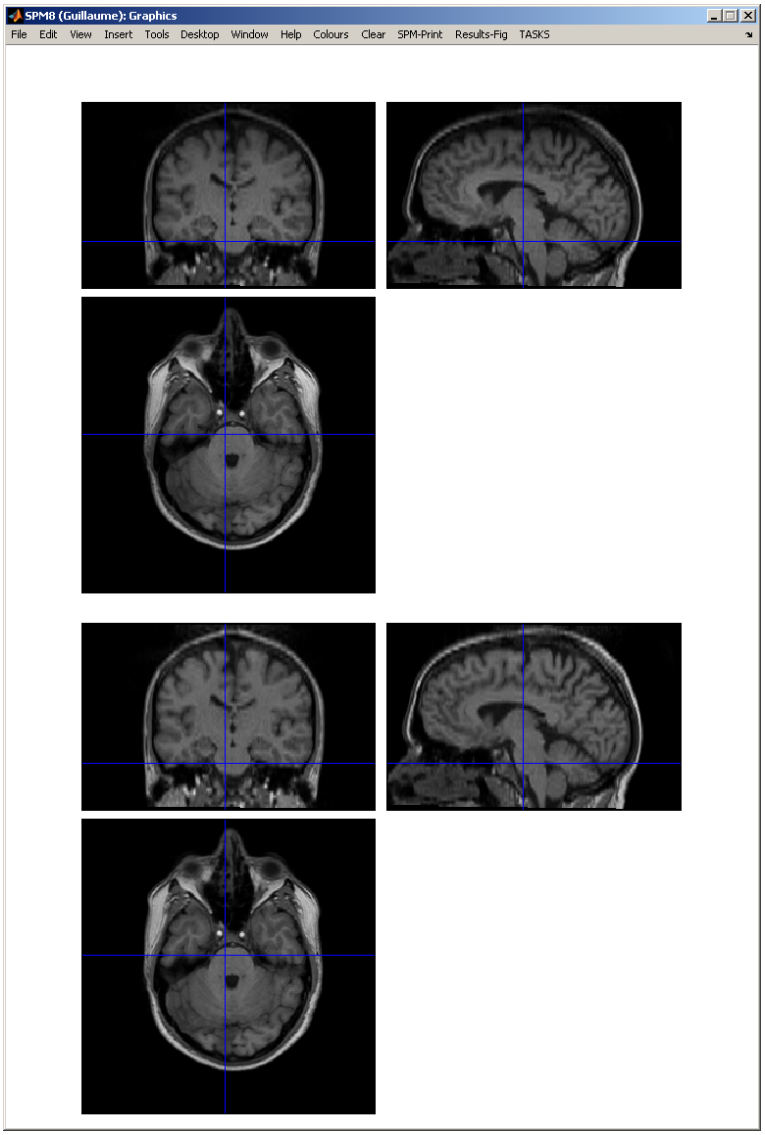


Figure 28.6: *Structural image (top) and bias-corrected structural image (bottom). Notice that the original structural is darker at the top than at the bottom. This non-uniformity has been removed in the bias-corrected image.*

realigned. You can then right click over the listed files, choose "Select all" and press "Done".

- Open "Writing Options", and change "Voxel sizes" from [2 2 2] to [3 3 3].[2]
- Press "Save", save the job as `normalise.mat` and then press the `RUN` button.

SPM will then write spatially normalised files to the functional data directory. These files have the prefix `w`.

If you wish to superimpose a subject's functional activations on their own anatomy[3] you will also need to apply the spatial normalisation parameters to their (bias-corrected) anatomical image. To do this

- Select NORMALISE (WRITE), highlight "Data", select "New Subject".
- Highlight "Parameter File", select the `sM00223_0020_seg_sn.mat` file that you created in the previous section, press "Done".
- Highlight "Images to Write", select the bias-corrected structural eg. `msM00223_002.img`, press "Done".
- Open "Writing Options", select voxel sizes and change the default [2 2 2] to [1 1 3] which corresponds to the original resolution of the images.
- Save the job as `norm_struct.mat` and press the `Run` button.

28.1.5 Smoothing

Press the SMOOTH button[4]. This will call up the specification of a smooth job in the batch editor.

- Select "Images to Smooth" and then select the spatially normalised files created in the last section eg. `^wrf.*`.
- Highlight "FWHM" and change [8 8 8] to [6 6 6]. This will smooth the data by 6mm in each direction.
- Save the job as `smooth.mat` and press the `Run` button.

An example of functional image and its smoothed version is displayed on Figure 28.7.

28.2 Model specification, review and estimation

To avoid T1 effects in the initial scans of an fMRI time series we recommend discarding the first few scans. To make this example simple, we'll discard the first complete cycle (12 scans, 04-15), leaving 84 scans, image files 16-99. This is best done by moving these files to a different directory.

Press the "Specify 1st-level" button. This will call up the specification of an fMRI specification job in the batch editor. Then

- Open the "Timing parameters" option.
- Highlight "Units for design" and select "Scans".
- Highlight "Interscan interval" and enter 7.
- Highlight "Data and Design" and select "New Subject/Session". Then open the newly created "Subject/Session" option.

[2] This step is not strictly necessary. It will write images out at a resolution closer to that at which they were acquired. This will speed up subsequent analysis and is necessary, for example, to make Bayesian fMRI analysis computationally efficient.

[3] Beginners may wish to skip this step, and instead just superimpose functional activations on an "average structural image".

[4] The smoothing step is unnecessary if you are only interested in Bayesian analysis of your functional data.

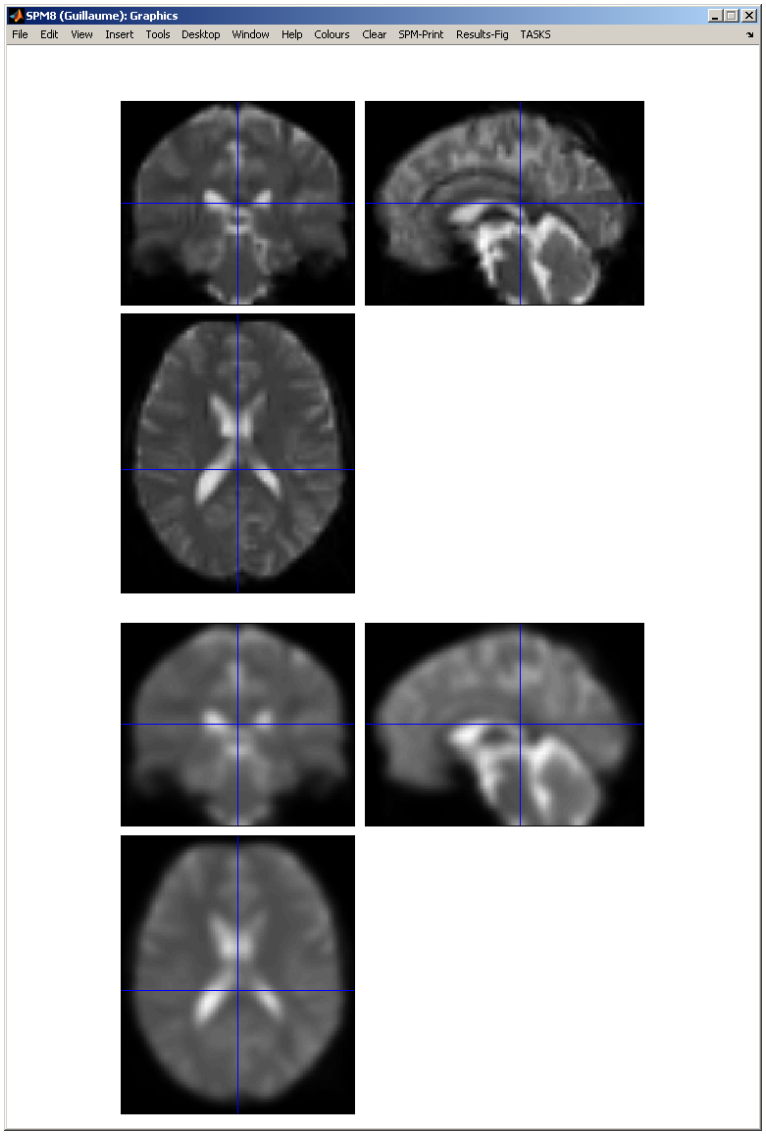


Figure 28.7: *Functional image (top) and 6mm-smoothed functional image (bottom). These images were obtained using SPM's "CheckReg" facility.*

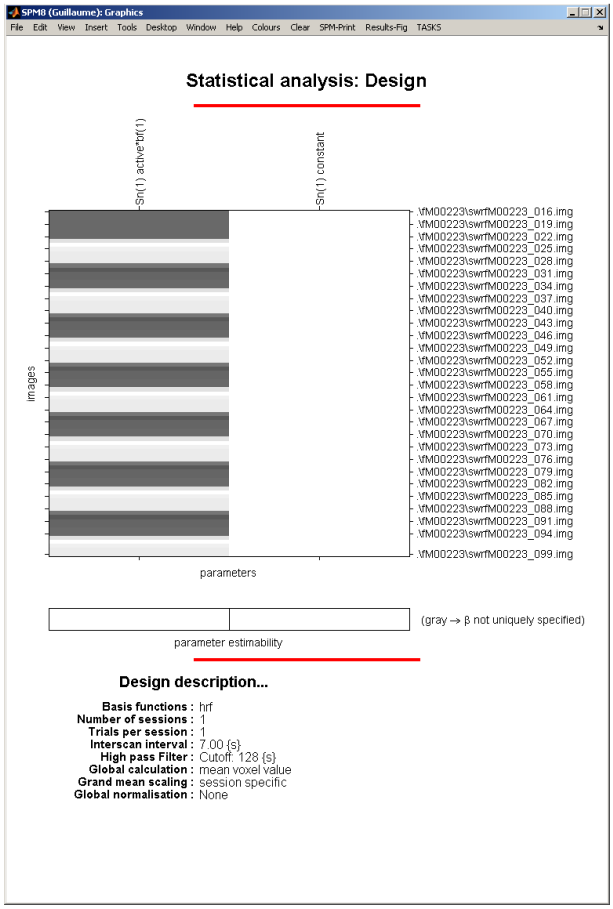


Figure 28.8: *`Design matrix`: The filenames on the right-hand side of the design matrix indicate the scan associated with each row.*

- Highlight "Scans" and use SPM's file selector to choose the 84 smoothed, normalised functional images ie `swrfM00223_016.img` to `swrfM00223_099.img`. These can be selected easily using the `^s.*'` filter, and select all (provided you have moved the scans 4 to 15 into a different directory). Then press "Done".
- Highlight "Condition" and select "New condition".
- Open the newly created "Condition" option. Highlight "Name" and enter "active". Highlight "Onsets" and enter "6:12:84". Highlight "Durations" and enter "6".
- Highlight "Directory" and select the `DIR/classical` directory you created earlier.
- Save the job as `specify.mat` and press the `Run` button.

SPM will then write an `SPM.mat` file to the `DIR/classical` directory. It will also plot the design matrix, as shown in Figure 28.8.

At this stage it is advisable to check your model specification using SPM's review facility which is accessed via the "Review" button. This brings up a "design" tab on the interactive window clicking on which produces a pulldown menu. If you select the first item "Design Matrix" SPM will produce the image shown in Figure 28.8. If you select "Explore" then "Session 1" then "active", SPM will produce the plots shown in Figure 28.9.

If you select the second item on the "Design" tab, "Design Orthogonality", SPM will produce the plot shown in Figure 28.10. Columns x_1 and x_2 are orthogonal if the inner product $x_1^T x_2 = 0$. The inner product can also be written $x_1^T x_2 = |x_1||x_2|cos\theta$ where $|x|$ denotes the length of x and θ is the angle between the two vectors. So, the vectors will be orthogonal if $cos\theta = 0$. The

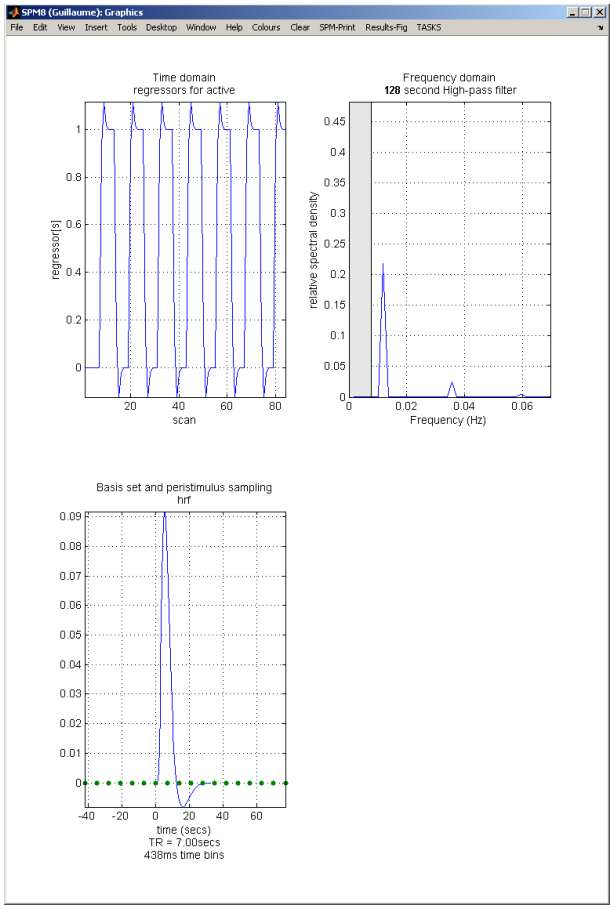


Figure 28.9: ***Exploring the design matrix in Figure 28.8:*** *This shows the time series of the "active" regressor (top left), a frequency domain plot of the active regressor (top right) and the basis function used to convert assumed neuronal activity into hemodynamic activity. In this model we used the default option - the canonical basis function. The frequency domain plot shows that the frequency content of the "active" regressor is above the set frequencies that are removed by the High Pass Filter (HPF) (these are shown in gray - in this model we accepted the default HPF cut-off of 128s or 0.008Hz).*

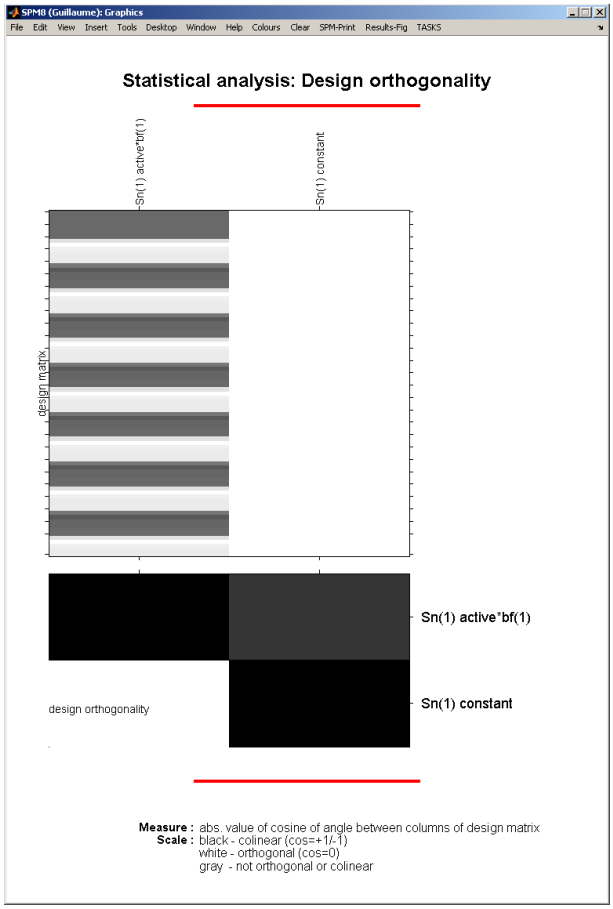


Figure 28.10: *`Design Orthogonality`: The description above the first column in the design matrix* Sn(1)Active*bf(1) *means that this column refers to the first session of data (in this analysis there is only 1 session), the name of this condition/trial is 'Active' and the trial information has been convolved with the first basis function (the canonical hemodynamic response). The constant regressor for session 1 is referred to as* Sn(1)Constant. *The orthogonality matrix at the bottom indicates a degree of collinearity between regressors.*

upper-diagonal elements in the matrix at the bottom of figure 28.10 plot $cos\theta$ for each pair of columns in the design matrix. Here we have a single entry. A degree of non-orthogonality or collinearity is indicated by the gray shading.

28.2.1 Estimate

Press the ESTIMATE button. This will call up the specification of an fMRI estimation job in the batch editor. Then

- Highlight the "Select SPM.mat" option and then choose the `SPM.mat` file saved in the classical subdirectory.
- Save the job as `estimate.job` and press the `Run` button.

SPM will write a number of files into the selected directory including an `SPM.mat` file.

28.3 Inference

After estimation:



Figure 28.11: *The contrast manager*

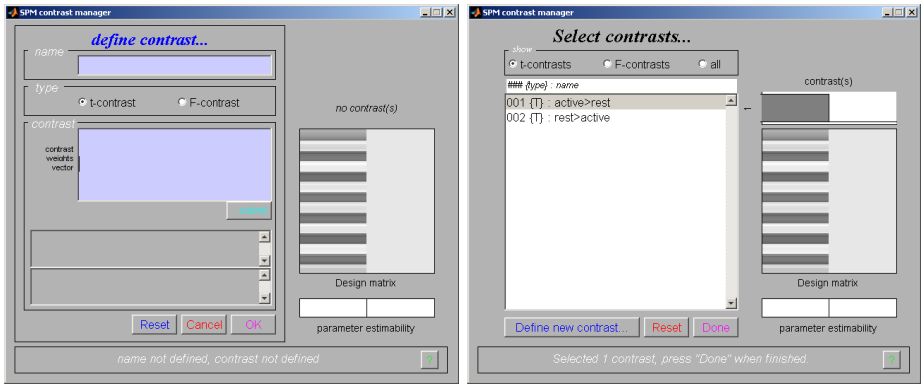


Figure 28.12: *Left: A contrast is entered by specifying the numeric values in the lower window and the name in the upper window. Right: After contrasts have been specified they can be selected.*

- Press "Results".
- Select the `SPM.mat` file created in the last section.

This will invoke the contrast manager.

28.3.1 Contrast manager

The contrast manager displays the design matrix (surfable) in the right panel and lists specified contrasts in the left panel. Either "t-contrast" or "F-contrast" can be selected. To examine statistical results for condition effects

- Select "Define new contrast"

One sided main effects for the active condition (i.e., a one-sided t-test) can be specified (in this example) as "1" (active > rest) and "-1" (rest > active). SPM will accept correct contrasts only. Accepted contrasts are displayed at the bottom of the contrast manager window in green, incorrect ones are displayed in red. To view a contrast

- Select the contrast name e.g., "active > rest".
- Press "Done".

28.3.2 Masking

You will then be prompted with

- *Mask with other contrast ? [Yes/No].*
- "Specify No".

Masking implies selecting voxels specified by other contrasts. If "yes", SPM will prompt for (one or more) masking contrasts, the significance level of the mask (default p = 0.05 uncorrected), and will ask whether an inclusive or exclusive mask should be used. Exclusive will remove all voxels which reach the default level of significance in the masking contrast, inclusive will remove all voxels which do not reach the default level of significance in the masking contrast. Masking does not affect p-values of the "target" contrast, it only includes or excludes voxels.

28.3.3 Thresholds

You will then be prompted with

- *Title for comparison ?*
- Enter eg. "active > rest".
- *p value adjustment to control: [FWE/none].*
- Select "FWE".
- *p value(family-wise error).*
- Accept the default value, 0.05.

A Family Wise Error (FWE) is a false positive anywhere in the SPM. Now, imagine repeating your experiment many times and producing SPMs. The proportion of SPMs containing FWEs is the FWE rate. A value of 0.05 implies that 1 in 20 SPMs contains a false positive somewhere in the image.

If you choose the "none" option above this corresponds to making statistical inferences at the "voxel level". These use "uncorrected" p values, whereas FWE thresholds are said to use "corrected" p values. SPM's default uncorrected p value is p=0.001. This means that the probability of a false positive at each voxel is 0.001. So if, you have 50,000 voxels you can expect $50,000 \times 0.001 = 50$ false positives in each SPM.

You will then be prompted with

- *Extent Threshold {voxels} [0].*
- Accept the default value, "0".

Entering a value v here will produce SPMs with clusters containing at least v voxels. SPM will then produce the SPM shown in Figure 28.13.

28.3.4 Files

A number of files are written to the working directory at this time. Images containing weighted parameter estimates are saved as `con_0002.hdr/img`, `con_0003.hdr/img`, etc. in the working directory. Images of T-statistics are saved as `spmT_0002.hdr/img`, `spmT_0003.hdr/img` etc., also in the working directory.

28.3.5 Maximum Intensity Projections

SPM displays a Maximum Intensity Projection (MIP) of the statistical map in the graphics window. The MIP is projected on a glass brain in three orthogonal planes. The MIP is surfable: right-clicking in the MIP will activate a pulldown menu, left-clicking on the red cursor will allow it to be dragged to a new position.

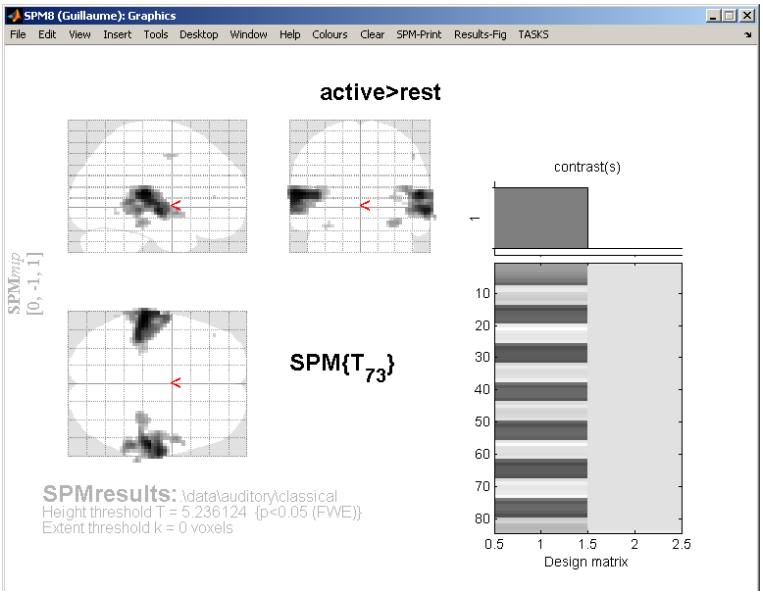


Figure 28.13: *SPM showing bilateral activation of auditory cortex.*

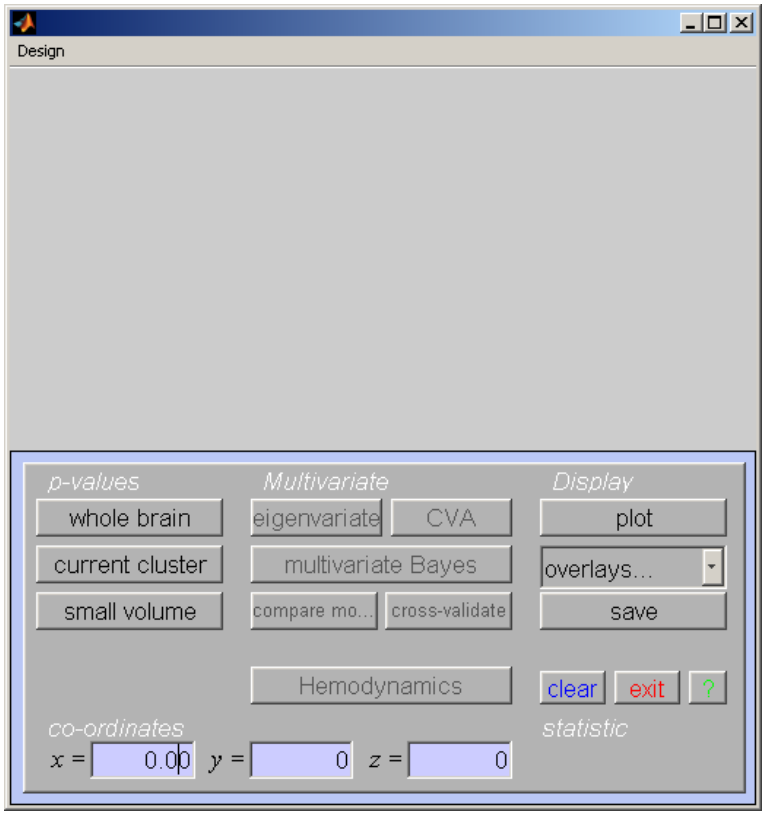
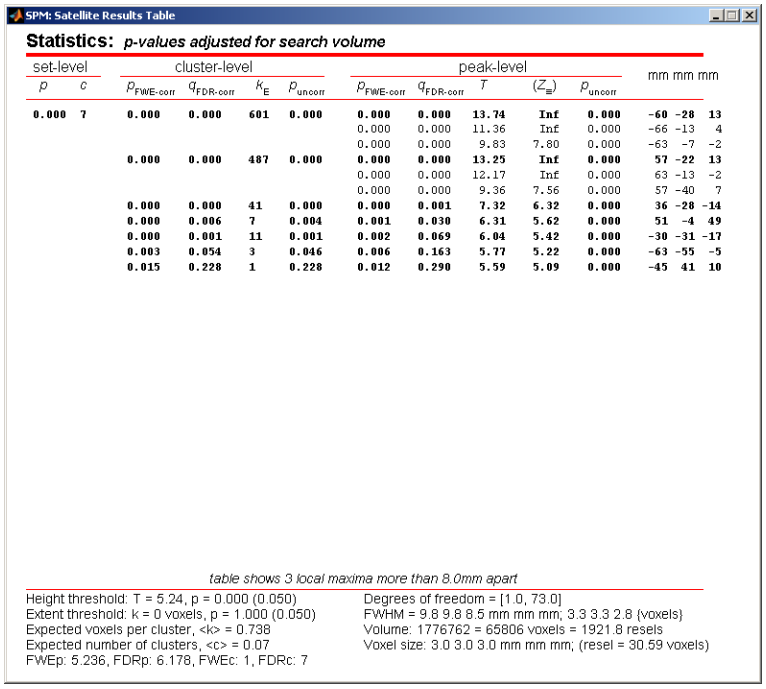


Figure 28.14: *SPM's Interactive window during results assessment. The "p-values" section is used to produce tables of statistical information. The visualisation section is used to plot responses at a voxel or to visual activations overlaid on anatomical images. The "Multivariate" section, ie. the "eigenvariate" button, is used to extract data for subsequent analyses such as assessment of PsychoPhysiological Interactions (PPIs) or Dynamic Causal Models (DCMs).*


SPM: Satellite Results Table

Statistics: *p-values adjusted for search volume*

set-level		cluster-level				peak-level					mm mm mm
p	c	$p_{FWE\text{-}corr}$	$q_{FDR\text{-}corr}$	k_E	p_{uncorr}	$p_{FWE\text{-}corr}$	$q_{FDR\text{-}corr}$	T	$(Z_{\equiv})$	p_{uncorr}	
0.000	**7**	**0.000**	**0.000**	**601**	**0.000**	**0.000**	**0.000**	**13.74**	**Inf**	**0.000**	**-60 -28 13**
						0.000	0.000	11.36	Inf	0.000	-66 -13 4
						0.000	0.000	9.83	7.80	0.000	-63 -7 -2
		0.000	**0.000**	**487**	**0.000**	**0.000**	**0.000**	**13.25**	**Inf**	**0.000**	**57 -22 13**
						0.000	0.000	12.17	Inf	0.000	63 -13 -2
						0.000	0.000	9.36	7.56	0.000	57 -40 7
		0.000	**0.000**	**41**	**0.000**	**0.000**	**0.001**	**7.32**	**6.32**	**0.000**	**36 -28 -14**
		0.000	**0.006**	**7**	**0.004**	**0.001**	**0.030**	**6.31**	**5.62**	**0.000**	**51 -4 49**
		0.000	**0.001**	**11**	**0.001**	**0.002**	**0.069**	**6.04**	**5.42**	**0.000**	**-30 -31 -17**
		0.003	**0.054**	**3**	**0.046**	**0.006**	**0.163**	**5.77**	**5.22**	**0.000**	**-63 -55 -5**
		0.015	**0.228**	**1**	**0.228**	**0.012**	**0.290**	**5.59**	**5.09**	**0.000**	**-45 41 10**

table shows 3 local maxima more than 8.0mm apart

Height threshold: T = 5.24, p = 0.000 (0.050)
Extent threshold: k = 0 voxels, p = 1.000 (0.050)
Expected voxels per cluster, <k> = 0.738
Expected number of clusters, <c> = 0.07
FWEp: 5.236, FDRp: 6.178, FWEc: 1, FDRc: 7

Degrees of freedom = [1.0, 73.0]
FWHM = 9.8 9.8 8.5 mm mm mm; 3.3 3.3 2.8 {voxels}
Volume: 1776762 = 65806 voxels = 1921.8 resels
Voxel size: 3.0 3.0 3.0 mm mm mm; (resel = 30.59 voxels)

Figure 28.15: *Volume table for "active > rest" effect. This table of values was created by pressing the "Results-Fig" tab at the top of the graphics window and then pressing the "whole brain" button. This displays the table of results in a separate window.*

28.3.6 Design matrix

SPM also displays the design matrix with the selected contrast. The design matrix is also surfable: right-clicking will show parameter names, left-clicking will show design matrix values for each scan.

In the SPM Interactive window (lower left panel) a button box appears with various options for displaying statistical results (p-values panel) and creating plots/overlays (visualisation panel). Clicking "Design" (upper left) will activate a pulldown menu as in the "Explore design" option.

28.3.7 Statistical tables

To get a summary of local maxima, press the "whole brain" button in the p-values section of the interactive window. This will list all clusters above the chosen level of significance as well as separate (>8mm apart) maxima within a cluster, with details of significance thresholds and search volume underneath, as shown in Figure 28.15

The columns in volume table show, from right to left:

- **x, y, z (mm)**: coordinates in MNI space for each maximum.
- peak-level: the chance (p) of finding (under the null hypothesis) a peak with this or a greater height (T- or Z-statistic), corrected (FWE or FDR)/ uncorrected for search volume.
- **cluster-level**: the chance (p) of finding a cluster with this many (ke) or a greater number of voxels, corrected (FWE or FDR)/ uncorrected for search volume.
- **set-level**: the chance (p) of finding this (c) or a greater number of clusters in the search volume.

It is also worth noting that:

- The table is surfable: clicking a row of cluster coordinates will move the pointer in the MIP to that cluster, clicking other numbers will display the exact value in the MATLAB window (e.g. 0.000 = 6.1971e-07).

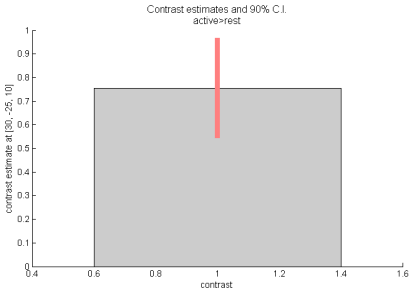


Figure 28.16: *Estimated effect size.*

- To inspect a specific cluster (e.g., in this example data set, the right auditory cortex), either move the cursor in the MIP (by left-clicking and dragging the cursor, or right-clicking the MIP background which will activate a pulldown menu).
- Alternatively, click the cluster coordinates in the volume table, or type the coordinates in the co-ordinates section of the interactive window.

It is also possible to produce tables of statistical information for a single cluster of interest rather than for the whole volume. Firstly, select the relevant cluster in the MIP and then press the "current cluster" button in the p-values section of the interactive window. This will show coordinates and voxel-level statistics for local maxima (>4mm apart) in the selected cluster. This table is also surfable.

28.3.8 Plotting responses at a voxel

A voxel can be chosen with co-ordinates corresponding to those in the interactive window. The responses at this voxel can then be plotted using the "Plot" button in the visualisation section of the interactive window. This will provide you with five further options:

1. Contrast estimates and 90% CI: SPM will prompt for a specific contrast (e.g., active>rest). The plot will show effect size and 90% confidence intervals. See eg. Figure 28.16.
2. Fitted responses: Plots adjusted data and fitted response across session/subject. SPM will prompt for a specific contrast and provides the option to choose different ordinates ("an explanatory variable", "scan or time", or "user specified"). If "scan or time", the plot will show adjusted or fitted data with errors added as shown in Figure 28.17.
3. Event-related responses: Plots adjusted data and fitted response across peri-stimulus time.
4. Parametric responses.
5. Volterra kernels.

For plotting event-related responses SPM provides three options

1. Fitted response and PSTH (peri-stimulus time histogram): plots mean regressor(s) (ie. averaged over session) and mean signal +/- SE for each peri-stimulus time bin.
2. Fitted response and 90% CI: plots mean regressor(s) along with a 90% confidence interval.
3. Fitted response and adjusted data: plots regressor(s) and individual data (note that in this example the data are shown in columns due to the fixed TR/ISI relationship).

Its worth noting that

- The values for the fitted response across session/subject for the selected plot can be displayed and accessed in the Matlab window by typing "Y". Typing "y" will display the adjusted data.
- "Adjusted" data = adjusted for confounds (e.g., global flow) and high- and low pass filtering.

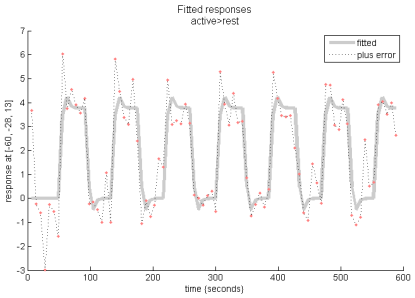


Figure 28.17: *Fitted responses.*

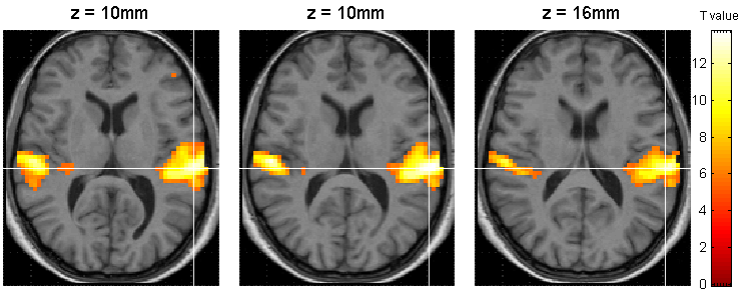


Figure 28.18: *Slices.*

28.3.9 Overlays

The visualisation section of the interactive window also provides an overlay facility for anatomical visualisation of clusters of activation. Pressing "Overlays" will activate a pulldown menu with three options

1. **Slices**: overlay on three adjacent (2mm) transaxial slices. SPM will prompt for an image for rendering. This could be a canonical image (see `spm_template.man`) or an individual T1/mean EPI image for single-subject analyses.

2. **Sections**: overlay on three intersecting (sagittal, coronal, transaxial) slices. These renderings are surfable: clicking the images will move the crosshair.

3. **Render**: overlay on a volume rendered brain, with options for using a smoothed brain, and old (left) and new (right) style rendering.

Renderings can be saved as `filename.img/hdr` in the working directory by using the *write filtered* option. In Figures 28.18, 28.19 and 28.20 the 'active > rest' activation has been superimposed on the spatially normalised, bias-corrected anatomical image `wmsM00223_002.img` created earlier.

For the "Render" option we first created a rendering for this subject. This was implemented by

- "Normalise (Write)" the two images `c1sM00223_002.img` and `c2sM00223_002.img` using the "Parameter File " `sM00223_002_seg_sn.mat` and a voxel size of [1 1 3].

- Selecting "Xtract Surface" from the "Render" pulldown menu.

- Selecting the gray and white matter images `wc1sM00223_002.img` and `wc2sM00223_002.img` created in the first step.

- Saving the results using the default options (Rendering and Surface).

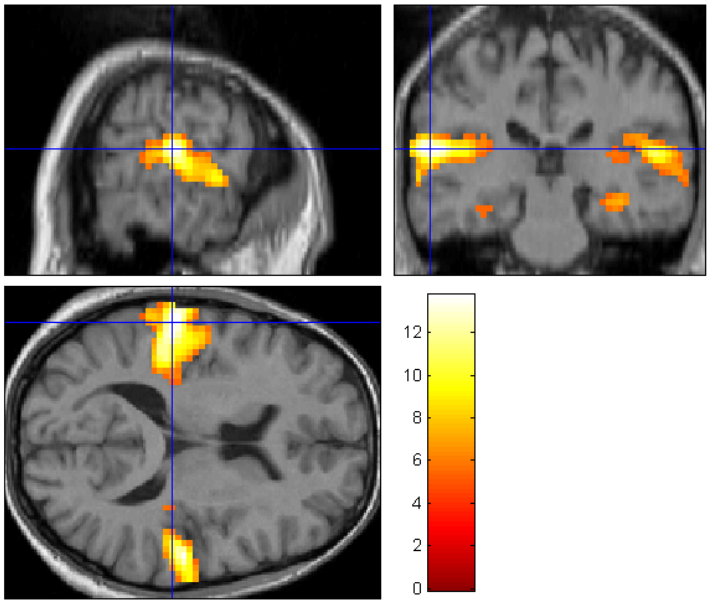


Figure 28.19: *Sections.*

SPM plots the rendered anatomical image in the graphics window and saves it as `render_wc1sM00223_002.ma`
The surface image is saved as `surf_wc1sM00223_002.mat`).

28.3.10 Miscellaneous

Other options (in the results controls panel):

- **clear**: clears lower subpanel of Graphics window,
- **exit**: exits the results section,
- **?**: launches help.

28.4 Bayesian analysis

28.4.1 Specification

Press the "Specify 1st-level" button. This will call up an fMRI specification job in the batch editor. Then

- Open the fMRI model specification option.
- Load the "specify.mat" job file created for the classical analysis.
- Open "Subject/Session", highlight "Scans".
- Deselect the smoothed functional images using the "unselect all" option available from a right mouse click in the SPM file selector (bottom window).
- Select the unsmoothed functional images using the `^w.*` filter and 'select all' option available from a right mouse click in the SPM file selector (top right window)[5]. The Bayesian analysis uses a spatial prior where the spatial regularity in the signal is estimated from the data. It

[5]Remember not to select the first 12 scans, scans 4 to 15, as these may contain T1 effects. This can be done during selection or by first moving the files to a different directory.

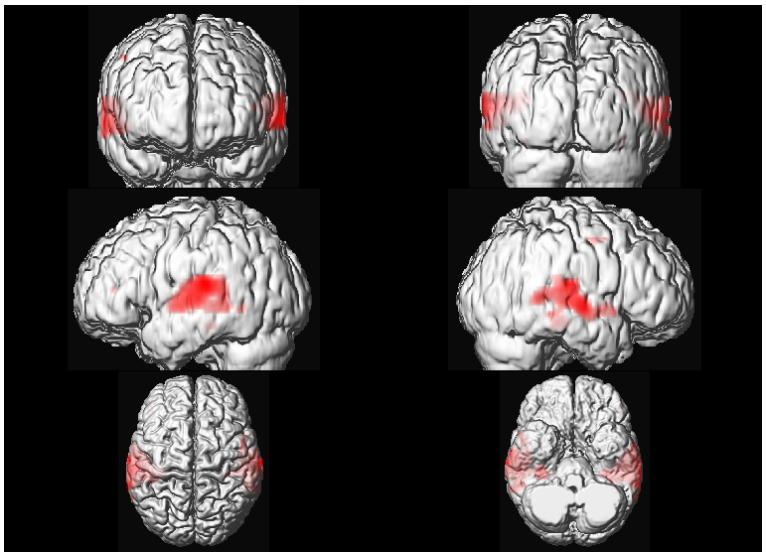


Figure 28.20: *Render.*

is therefore not necessary to create smoothed images if you are only going to do a Bayesian analysis.

- Press "Done".
- Highlight "Directory" and select the `DIR/bayesian` directory you created earlier (you will first need to deselect the `DIR/classical` directory).
- Save the job as `specify_bayesian.mat` and press the "Run" button.

28.4.2 Estimation

Press the "Estimate" button. This will call up the specification of an fMRI estimation job in the batch editor. Then

- Highlight the "Select SPM.mat" option and then choose the SPM.mat file saved in the `DIR/bayesian` directory.
- Highlight "Method" and select the "Choose Bayesian 1st-level" option.
- Open the newly created "Bayesian 1st-level" option, highlight "AR model order" and select 0. This data set has a TR=7s, so is unlikely to have temporally autocorrelated errors.
- Save the job as `estimate_bayesian.job` and press the "Run" butto.

SPM will write a number of files into the output directory including

- An `SPM.mat` file.
- Images of estimated regression coefficients `Cbeta_0001.img` and `Cbeta_0002.img`. These filenames are prefixed with a `C` indicating that these are the mean values of the "Conditional" or "Posterior" density.
- Images of error bars/standard deviations on the regression coefficients `SDbeta_0001.img` and `SDbeta_0002.img`.
- An image of the standard deviation of the error `Sess1_SDerror.img`.
- An image `mask.img` indicating which voxels were included in the analysis.

28.4.3 Inference

After estimation:

- Press "Results",
- Select the `SPM.mat` file created in the last section,
- Select "Define new contrast",
- Enter the name "active > rest",
- Enter the value "1", press "Submit", "OK", "Done",
- *Mask with other contrast ? [Yes/No]*,
- Specify No,
- Title for comparison, accept the default,
- *Effect size threshold for PPM*,
- Enter the value 2,
- *Posterior probability threshold for PPM*,
- Enter the value 0.99,
- *Extent threshold [0]*,
- Accept the default value,
- *Plot effect size [Yes/No]*,
- Select the default 'Yes'.

SPM will then plot a map of effect sizes at voxels where it is 99% sure that the effect size is greater than 2% of the global mean. This is a large activation. Then use overlays, sections, select the normalised structural image created earlier and move the cursor to the activation in the left hemisphere. This should create the plot shown in Figure 28.21.

It is also possible to look for regions where responses in the active condition are different to those at rest. Active responses could be greater or smaller.

- Press "Results",
- Select the `SPM.mat` file created in the last section,
- Select "Define new contrast" and highlight the "F" radio button,
- Enter the name "active != rest",
- Enter the value "1", press "Submit", "OK", "Done",
- *Mask with other contrast ? [Yes/No]*,
- Specify "No",
- Title for comparison, accept the default,
- *Posterior probability threshold for PPM*,
- Accept the default value[6],

[6]The default PPM threshold is set to $1 - 1/S$ where S is the number of voxels in the volume being analysed. The rationale for this is that inference is based on an approximate posterior distribution, Q, which factorises across voxels. The approximate posterior is chosen to best match the true posterior in the sense of KL-divergence. Given the factorisation in Q, the expected number of false positives in the PPM is 1.

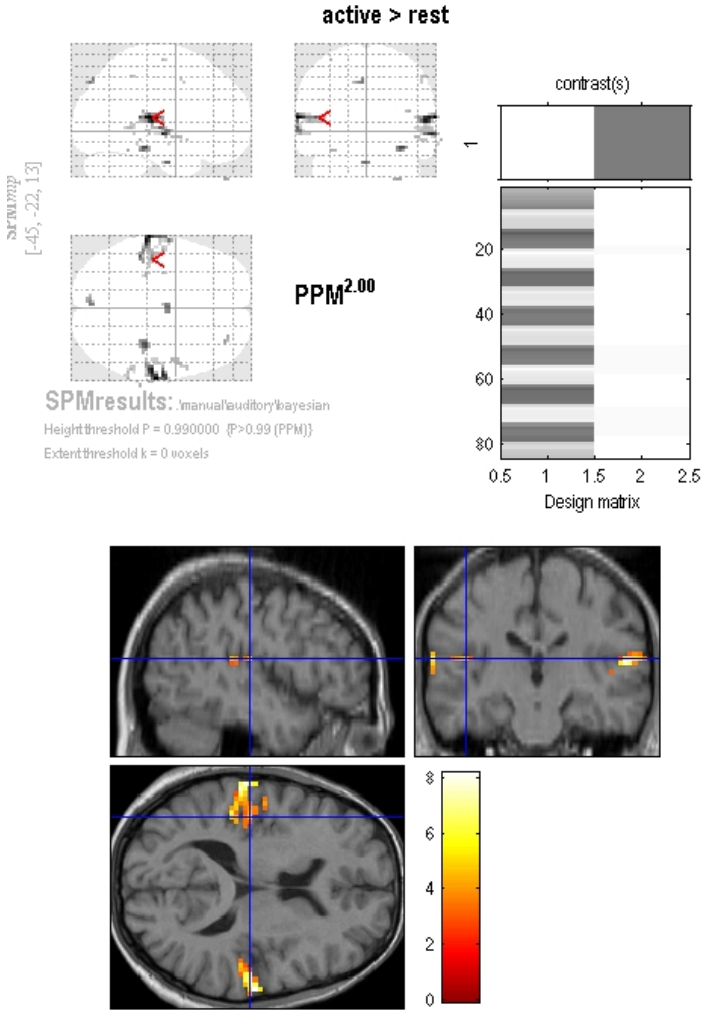


Figure 28.21: ***Bayesian analysis:*** *MIP and overlay of effect sizes at voxels where SPM is 99% sure that the effect size is greater than 2% of the global mean.*

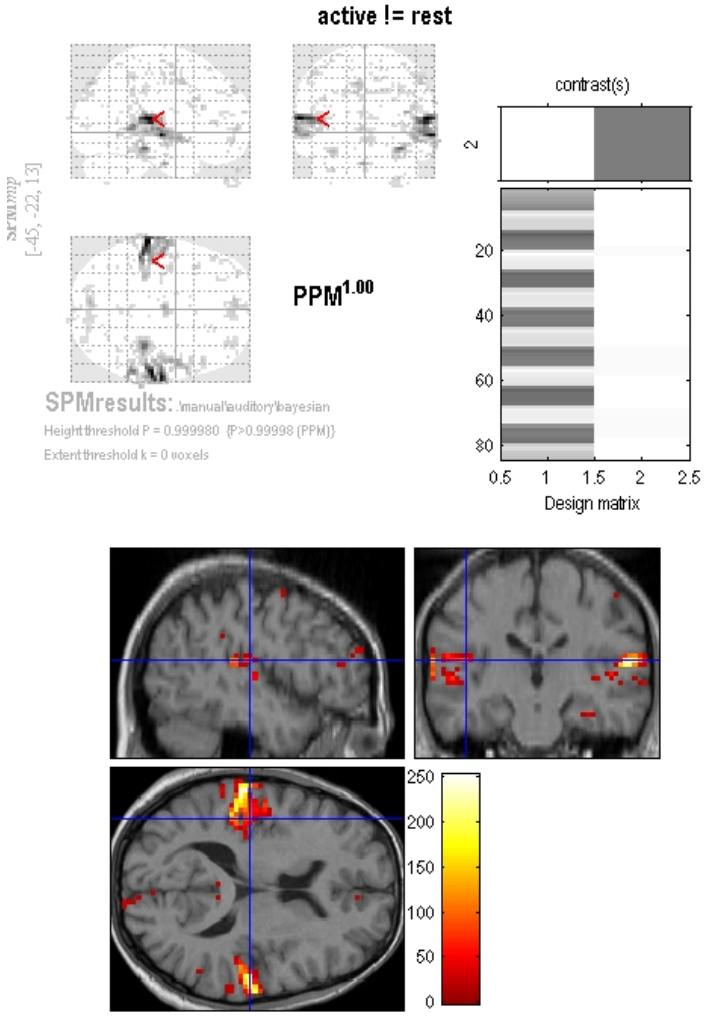


Figure 28.22: ***Two-sided Bayesian analysis:*** *MIP and overlay of χ^2 statistic values at above threshold voxels. This shows regions where activity is different between active and rest conditions, whether positive or negative.*

- *Extent threshold [0]*,
- Accept the default value, 0,
- *Plot effect size [Yes/No]*,
- Select the default "Yes".

SPM will then plot a map of χ^2 statistic values at above threshold voxels. Then use overlays, sections, select the normalised structural image created earlier and move the cursor to the activation in the left hemisphere. This should create the plot shown in Figure 28.22

When you revisit the contrast manager this contrast will be referred to as a "P" contrast, rather than an "F" contrast. This indicates that Bayes rule is used to make the inference. To indicate that we are testing a two-sided effect it is advisable to make this clear when naming the contrast (as we have done with the label "active != rest").

Chapter 29

Face data

As another, more sophisticated example, consider the data from a repetition priming experiment performed using event-related fMRI. Briefly, this is a 2×2 factorial study with factors "fame" and "repetition" where famous and non-famous faces were presented twice against a checkerboard baseline (for more details, see [41]). The subject was asked to make fame judgements by making key presses. There are thus four event-types of interest; first and second presentations of famous and non-famous faces, which we denote N1, N2, F1 and F2. The experimental stimuli and timings of events are shown in Figures 29.1 and 29.2.

Images were acquired using continuous Echo-Planar Imaging (EPI) with TE=40ms, TR=2s and 24 descending slices (64×64 3×3 mm^2), 3mm thick with a 1.5mm gap. The data archive is available from the SPM website[1]. This contains 351 Analyze format functional images `sM03953_0005_*.img` of dimension 64×64×24 with 3×3×4.5 mm^3 voxels. A structural image is also provided in Analyze format (`sM03953_0007.img`).

To analyse the data, first create a new directory `DIR` eg. `C:\data\face_rep`, in which to place the results of your analysis. Then create 4 subdirectories (i) `jobs`, (ii) `categorical`, (iii) `parametric` and (iv) `bayesian`. As the analysis proceeds these directories will be filled with job-specification files, design matrices and models estimated using classical or Bayesian methods.

As well as the classical/Bayesian distinction we will show how this data can be analysed from a parametric as well as a categorical perspective. We will look at the main effects of fame and repetition and in the parameteric analysis we will look at responses as a function of "lag", that is, the number of faces intervening between repetition of a specific face.

Start up matlab, enter your jobs directory and type `spm fmri` at the MATLAB prompt. SPM will then open in fMRI mode with three windows (1) the top-left or "Menu" window, (2) the bottom-left or "Interactive" window and (3) the right-hand or "Graphics" window. Analysis then takes place in three major stages (i) spatial pre-processing, (ii) model specification, review and estimation and (iii) inference. These stages organise the buttons in SPM's base window.

29.1 Spatial pre-processing

29.1.1 Display

Display eg. the first functional image using the "Display" button. Note orbitofrontal and inferior temporal drop-out and ghosting. This can be seen more clearly by selecting "brighten" from the "Effects" tab in the "Colours" at the top of the Graphics window.

29.1.2 Realignment

Under the spatial pre-processing section of the SPM base window select REALIGN (EST & RES) from the REALIGN pulldown menu. This will call up a realignment job specification in the batch editor window. Then

- Highlight data, select "New Session", then highlight the newly created "Session" option.

[1] Face Repetition dataset: http://www.fil.ion.ucl.ac.uk/spm/data/face_rep/

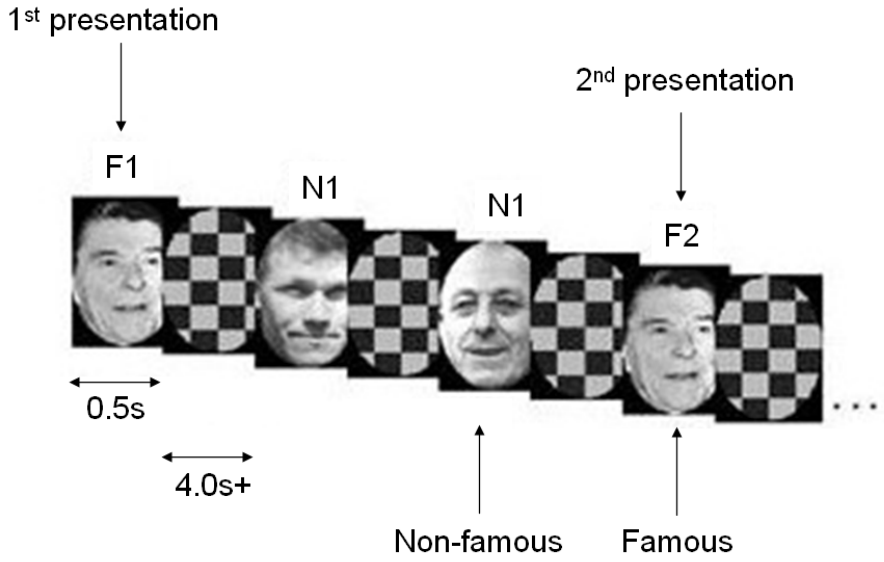


Figure 29.1: ***Face repetition paradigm****: There were 2 presentations of 26 Famous and 26 Nonfamous Greyscale photographs, for 0.5s each, randomly intermixed. The minimal Stimulus Onset Asynchrony (SOA)=4.5s, with probability 2/3 (ie 1/3 null events). The subject made one of two right finger key presses denoting whether or not the subject thought the face was famous.*

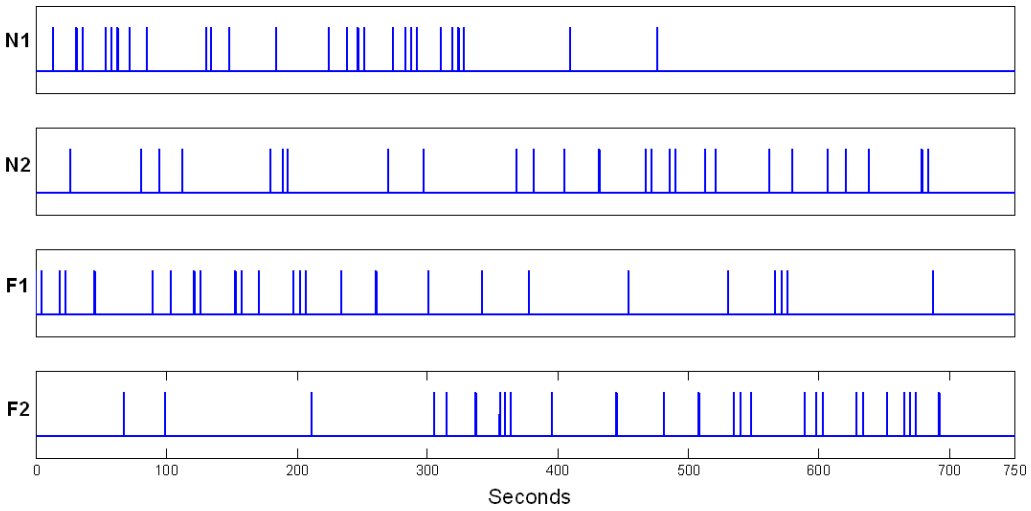


Figure 29.2: *Time series of events.*

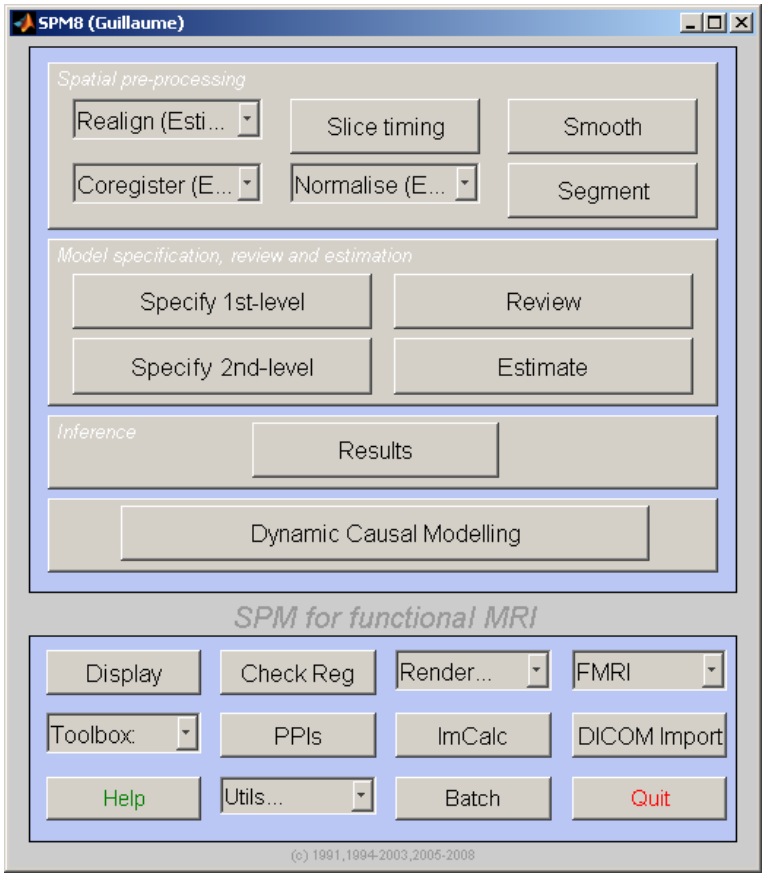


Figure 29.3: *The SPM base window comprises three sections (i) spatial pre-processing, (ii) model specification, review and estimation and (iii) inference.*

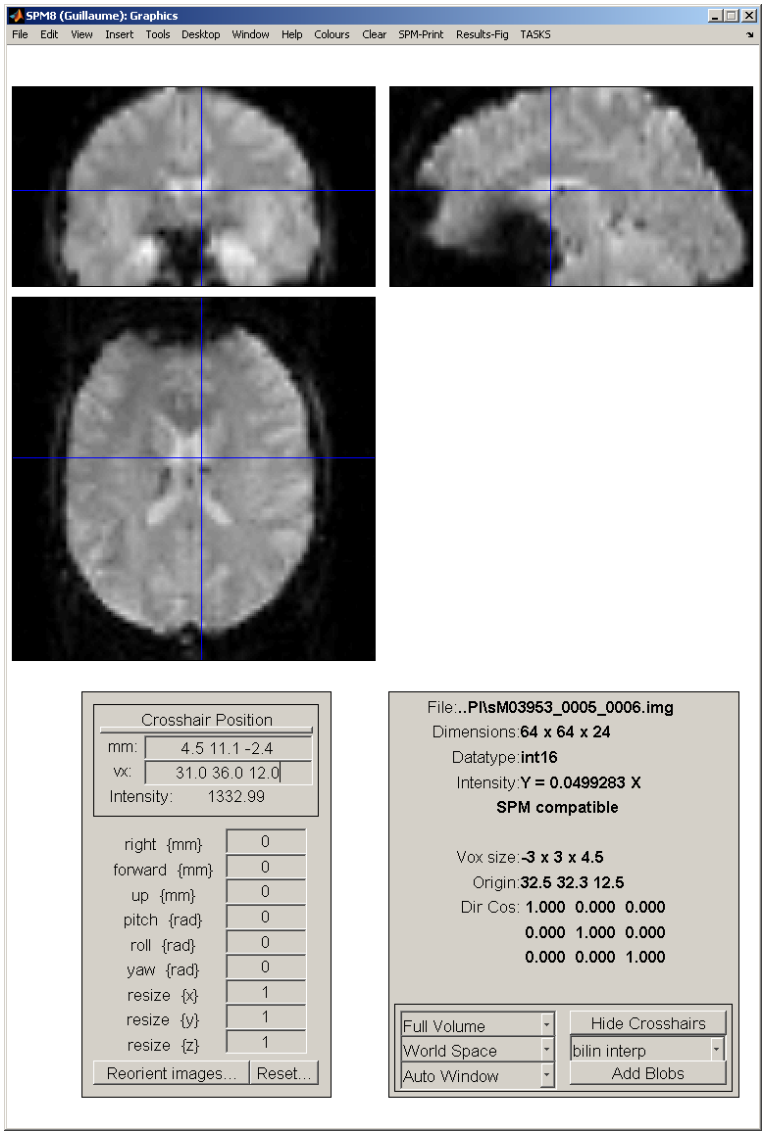


Figure 29.4: *Signal dropout in EPI images.*

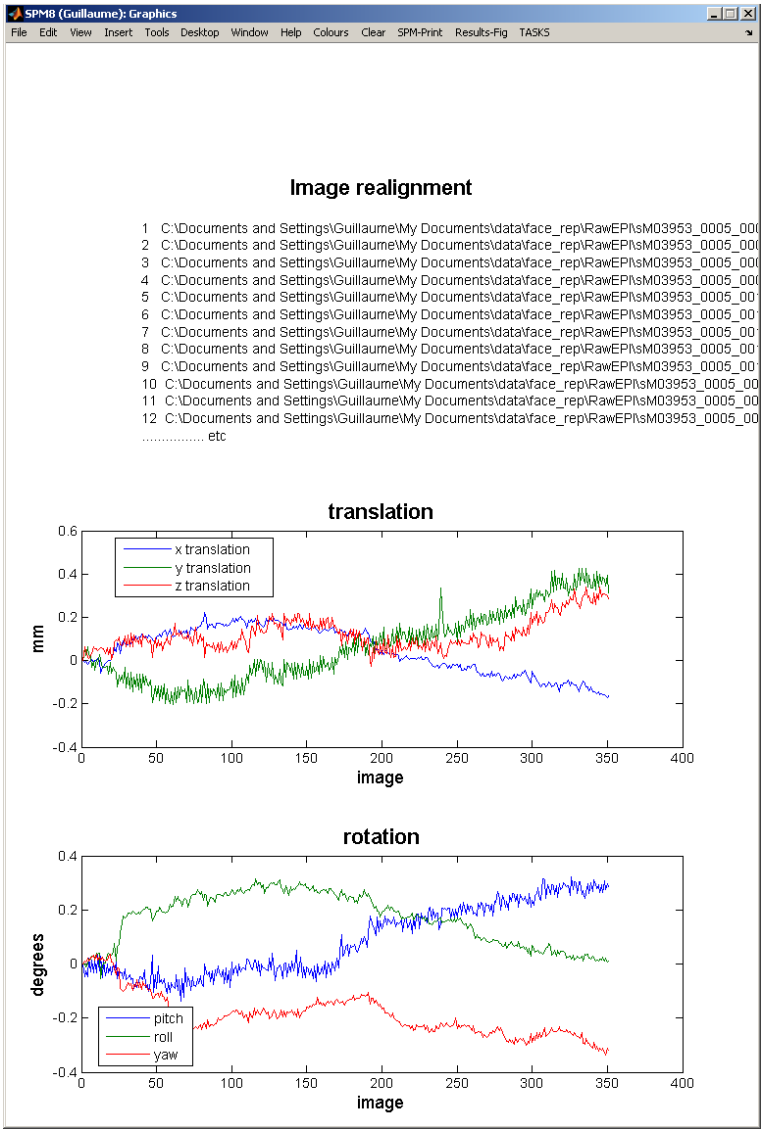


Figure 29.5: ***Realignment of face data****: Movement less than the size of a voxel, which for this data set is 3mm, is not considered problematic.*

- Select "Specify Files" and use the SPM file selector to choose all of your functional images eg. `sM03953_0005_*.img`.
- Save the job file as eg. `DIR/jobs/realign.mat`.
- Press the `Run` button in the batch editor window (green triangle).

This will run the realign job which will write realigned images into the directory where the functional images are. These new images will be prefixed with the letter "`r`". SPM will then plot the estimated time series of translations and rotations shown in Figure 29.5. These data, the realignment parameters, are also saved to a file eg. `rp_sM03953_0005_0006.txt`, so that these variables can be used as regressors when fitting GLMs. To prepare for this, copy the file into the `DIR/jobs` directory and rename it `movepars.txt`. This allows movements effects to be discounted when looking for brain activations.

SPM will also create a mean image eg. `meansM03953_0005_0006.img` which will be used in the next step of spatial processing - coregistration.

29.1.3 Slice timing correction

Press the SLICE TIMING button. This will call up the specification of a slice timing job in the batch editor window. Note that these data consist of N=24 axial slices acquired continuously with a TR=2s (ie TA = TR - TR/N, where TA is the time between the onset of the first and last slice of one volume, and the TR is the time between the onset of the first slice of one volume and the first slice of next volume) and in a descending order (ie, most superior slice was sampled first). The data however are ordered within the file such that the first slice (slice number 1) is the most inferior slice, making the slice acquisition order [24 23 22 ... 1].

- Highlight "Data" and select "New Sessions"
- Highlight the newly create "Sessions" option, "Specify Files" and select the 351 realigned functional images using the filter `^r.*`.
- Select "Number of Slices" and enter 24.
- Select TR and enter 2.
- Select TA and enter 1.92 (or 2 - 2/24).
- Select "Slice order" and enter 24:-1:1.
- Select "Reference Slice", and enter 12.
- Save the job as `slice_timing.mat` and press the "Run" button.

SPM will write slice-time corrected files with the prefix "`a`" in the functional data directory.

29.1.4 Coregistration

Select COREGISTER (ESTIMATE) from the `Coregister` pulldown menu. This will call up the specification of a coregistration job in the batch editor window.

- Highlight "Reference Image" and then select the mean functional image `meansM03953_0005_0006.img`.
- Highlight "Source Image" and then select the structural image eg. `sM03953_0007.img`.
- Press the "Save" button and save the job as `coreg.job`
- Then press the "Run" button.

SPM will then implement a coregistration between the structural and functional data that maximises the mutual information. The image in figure 29.6 should then appear in the graphics window. SPM will have changed the header of the source file which in this case is the structural image `sM03953_0007.img`.

29.1.5 Segmentation

Press the SEGMENT button. This will call up the specification of a segmentation job in the batch editor window. Highlight the "Data" field and then select the subjects coregistered anatomical image eg. `sM03953_0007.img`. Save the job file as `segment.mat` and then press the `Run` button. SPM will segment the structural image using the default tissue probability maps as priors. SPM will create, by default, gray and white matter images and bias-field corrected structral image. These can be viewed using the CheckReg facility as described in the previous section. Figure 29.7 shows the gray matter image, `c1sM03953_0007.img`, along with the original structural[2].

SPM will also write a spatial normalisation eg. `sM03953_0007_seg_sn.mat` file in the original structural directory. This will be used in the next section to normalise the functional data.

[2]Segmentation can sometimes fail if the source (structural) image is not close in orientation to the MNI templates. It is generally advisable to manually orient the structural to match the template (ie MNI space) as close as possible by using the "Display" button, adjusting x/y/z/pitch/roll/yaw, and then pressing the "Reorient" button.

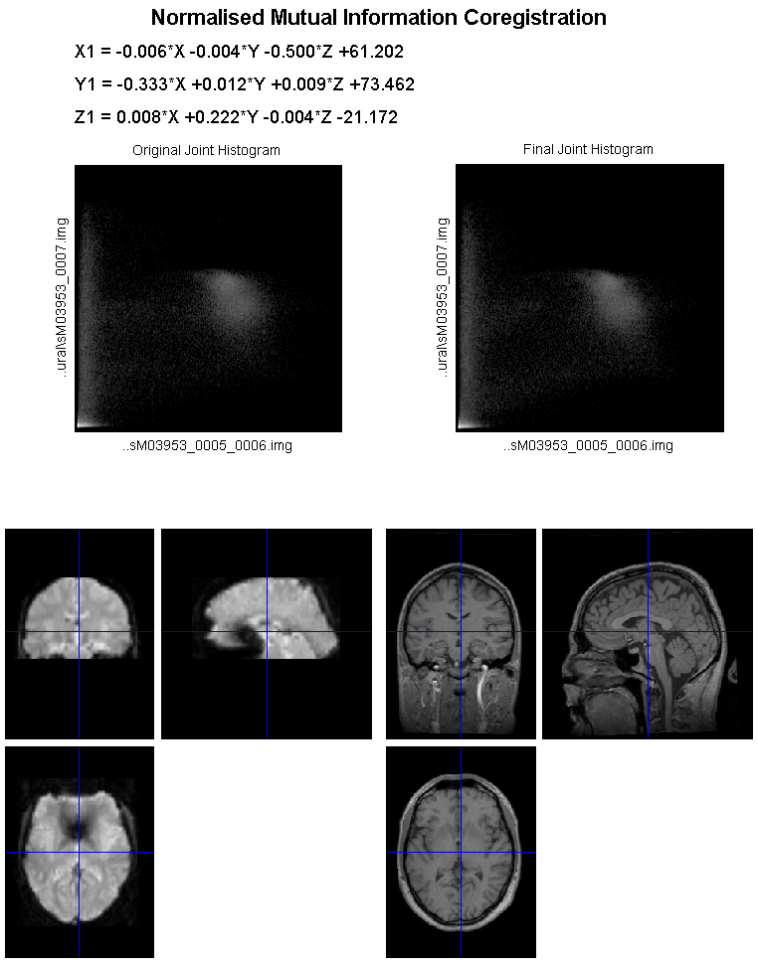


Figure 29.6: *Mutual Information Coregistration of Face data.*

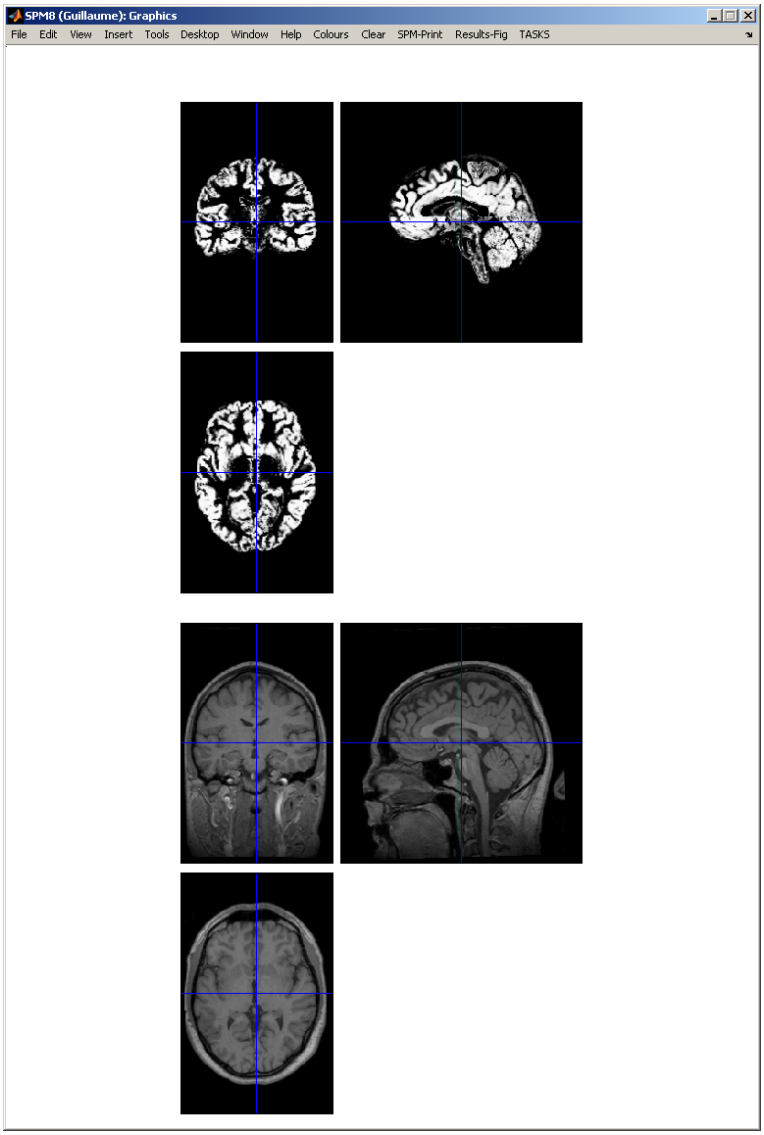


Figure 29.7: *Gray matter (top) produced by segmentation of structural image (below).*

29.1.6 Normalise

Select NORMALISE (WRITE) from the NORMALISE pulldown menu. This will call up the specification of a normalise job in the batch editor window.

- Highlight "Data", select "New Subject".
- Open "Subject", highlight "Parameter File" and select the `sM03953_0007_seg_sn.mat` file that you created in the previous section.
- Highlight images to write and select all of the slice-time corrected, realigned functional images `arsM*.img`. Note: This can be done efficiently by changing the filter in the SPM file selector to `^ar.*`. You can then right click over the listed files, choose "Select all". You might also want to select the mean functional image created during realignment (which would not be affected by slice-time correction), i.e, the `meansM03953_0005_006.img`. Then press "Done".
- Open "Writing Options", and change "Voxel sizes" from [2 2 2] to [3 3 3][3].
- Press "Save", save the job as normalise.mat and then press the `Run` button.

SPM will then write spatially normalised files to the functional data directory. These files have the prefix "`w`".

If you wish to superimpose a subject's functional activations on their own anatomy[4] you will also need to apply the spatial normalisation parameters to their (bias-corrected) anatomical image. To do this

- Select NORMALISE (WRITE), highlight 'Data', select "New Subject".
- Highlight "Parameter File", select the `sM03953_0007_seg_sn.mat` file that you created in the previous section, press "Done".
- Highlight "Images to Write", select the bias-corrected structural eg. `msM03953_0007.img`, press "Done".
- Open "Writing Options", select voxel sizes and change the default [2 2 2] to [1 1 1] which better matches the original resolution of the images [1 1 1.5].
- Save the job as `norm_struct.mat` and press `Run` button.

29.1.7 Smoothing

Press the SMOOTH button[5]. This will call up the specification of a smooth job in the batch editor window.

- Select "Images to Smooth" and then select the spatially normalised files created in the last section eg. `war*.img`.
- Save the job as `smooth.mat` and press `Run` button.

This will smooth the data by (the default) 8mm in each direction, the default smoothing kernel width.

[3] This step is not strictly necessary. It will write images out at a resolution closer to that at which they were acquired. This will speed up subsequent analysis and is necessary, for example, to make Bayesian fMRI analysis computationally efficient.

[4] Beginners may wish to skip this step, and instead just superimpose functional activations on an "canonical structural image".

[5] The smoothing step is unnecessary if you are only interested in Bayesian analysis of your functional data.

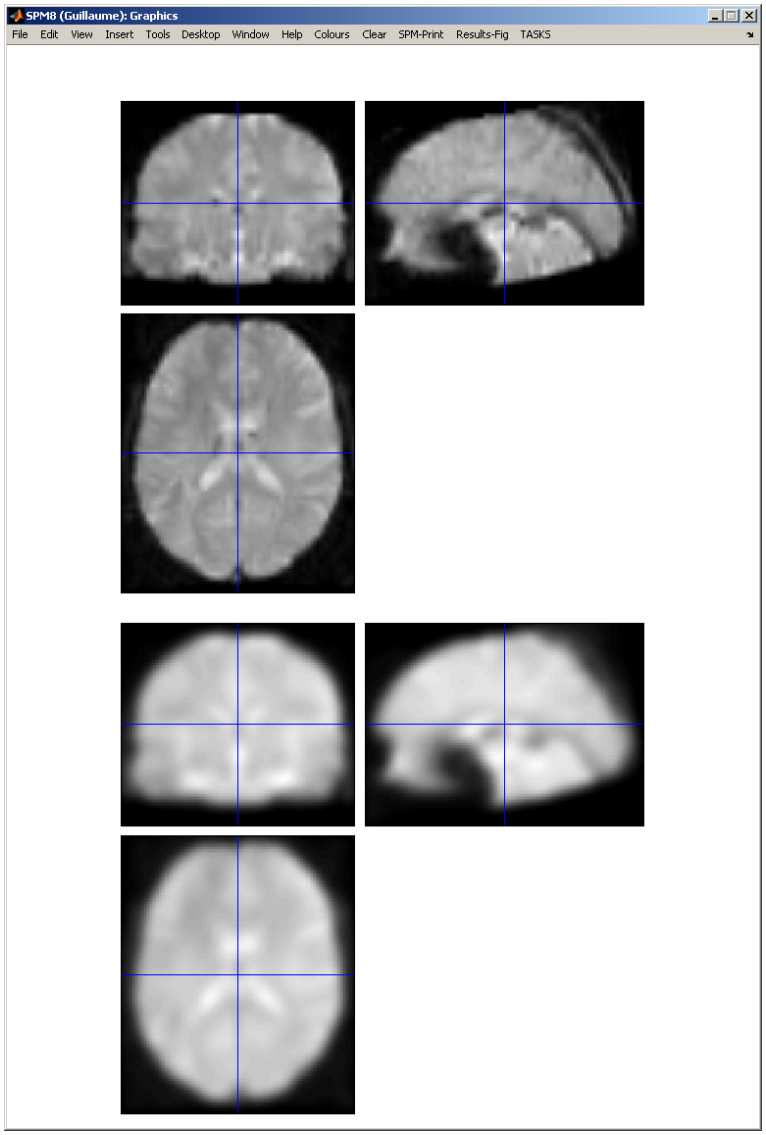


Figure 29.8: *Functional image (top) and 8mm-smoothed functional image (bottom). These images were plotted using SPM's "CheckReg" facility.*

29.2 Modelling categorical responses

Before setting up the design matrix we must first load the Stimulus Onsets Times (SOTs) and movement parameters into matlab. SOTs are stored in the `sots.mat` file in a cell array such that eg. `sot{1}` contains stimulus onset times in TRs for event type 1, which is N1. Event-types 2, 3 and 4 are N2, F1 and F2.[6]

- At the MATLAB command prompt type `load sots`
- Then type `load movepars.txt`

Now press the SPECIFY 1ST-LEVEL button. This will call up the specification of a fMRI specification job in the batch editor window. Then

- In the "Timing paramaters" option,
- Highlight "Units for design" and select "Scans",
- Highlight "Interscan interval" and enter 2,
- Highlight "Microtime resolution" and enter 24,
- Highlight "Microtime onset" and enter 12. These last two options make the creating of regressors commensurate with the slice-time correction we have applied to the data, given that there are 24 slices and that the reference slice to which the data were slice-time corrected was the 12th (middle slice in time).
- Highlight "Data and Design" and select "New Subject/Session".
- Highlight "Scans" and use SPM's file selector to choose the 351 smoothed, normalised, slice-time corrected, realigned functional images ie `swarsM.img`. These can be selected easily using the `^swar.*` filter, and select all. Then press "Done".
- Highlight "Conditions" and select "New condition"[7].
- Open the newly created "Condition" option. Highlight "Name" and enter "N1". Highlight "Onsets" and enter `sot{1}`. Highlight "Durations" and enter 0.
- Highlight "Conditions" and select "Replicate condition".
- Open the newly created "Condition" option (the lowest one). Highlight "Name" and change to "N2". Highlight "Onsets" and enter `sot{2}`.
- Highlight "Conditions" and select "Replicate condition".
- Open the newly created "Condition" option (the lowest one). Highlight "Name" and change to "F1". Highlight "Onsets" and enter `sot{3}`.
- Highlight "Conditions" and select "Replicate condition".
- Open the newly created "Condition" option (the lowest one). Highlight "Name" and change to "F2". Highlight "Onsets" and enter `sot{4}`.
- Highlight "Multiple Regressors" and select the `movepars.txt` file[8].
- Highlight "Factorial Design", select "New Factor", open the newly created "Factor" option, highlight "Name" and enter "Fam", highlight "Levels" and enter 2.

[6] Unlike previous analyses of these data in SPM99 and SPM2, we will not bother with extra event-types for the (rare) error trials.

[7] It is also possible to enter information about all of the conditions in one go. This requires much less button pressing and can be implemented by highlighting the "Multiple conditions" option and then selecting the `all-conditions.mat` file, which is also provided on the webpage.

[8] It is also possible to enter regressors one by one by highlighting "Regressors" and selecting "New Regressor" for each one. Here, we benefit from the fact that the realignment stage produced a text file with the correct number of rows (351) and columns (6) for SPM to add 6 regressors to model (linear) rigid-body movement effects.

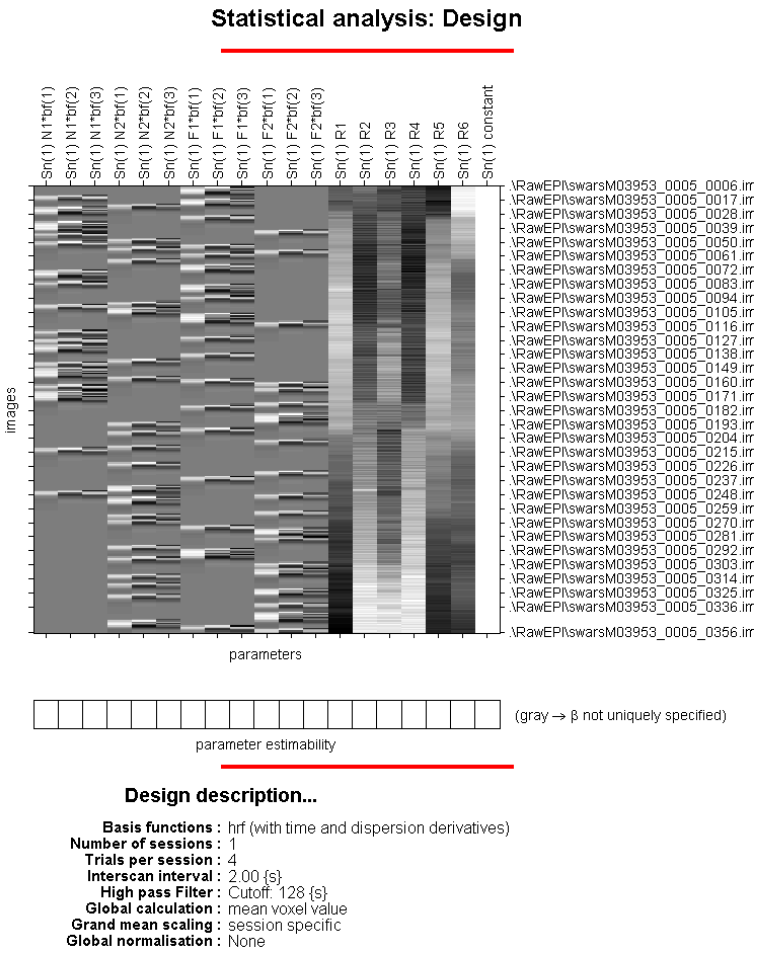


Figure 29.9: ***Design matrix.***

- Highlight "Factorial Design", select "New Factor", open the newly created "Factor" option, highlight "Name" and enter "Rep", highlight "Levels" and enter 2[9].

- Open "Canonical HRF" under "Basis Functions". Select "Model derivatives" and select "Time and Dispersion derivatives".

- Highlight "Directory" and select the `DIR/categorical` directory you created earlier.

- Save the job as `categorical_spec.mat` and press the `Run` button.

SPM will then write an `SPM.mat` file to the `DIR/categorical` directory. It will also plot the design matrix, as shown in Figure 29.9.

At this stage it is advisable to check your model specification using SPM's review facility which is accessed via the "Review" button. This brings up a "design" tab on the interactive window clicking on which produces a pulldown menu. If you select the first item "Design Matrix" SPM will produce the image shown in Figure 29.9. If you select "Explore" then "Session 1" then "N1", SPM will produce the plots shown in Figure 29.10.

[9]The order of naming these factors is important - the factor to be specified first is the one that "changes slowest" ie. as we go through the list of conditions N1, N2, F1, F2 the factor "repetition" changes every condition and the factor "fame" changes every other condition. So "Fam" changes slowest and is entered first.

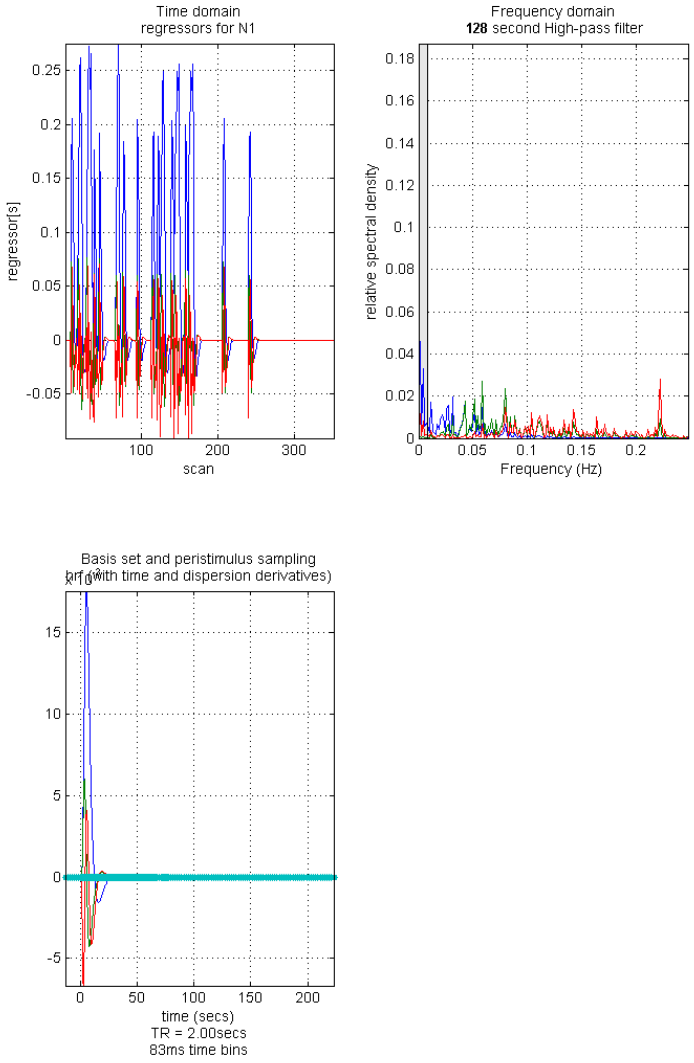


Figure 29.10: ***Exploring the design matrix in Figure 29.9.*** *This shows the time series of the "N1" regressor (top left), the three basis functions used to convert assumed neuronal activity into hemodynamic activity (bottom left), and a frequency domain plot of the three regressors for the basis functions in this condition (top right). The frequency domain plot shows that the frequency content of the "N1" condition is generally above the set frequencies that are removed by the High Pass Filter (HPF) (these are shown in gray - in this model we accepted the default HPF cut-off of 128s or 0.008Hz).*

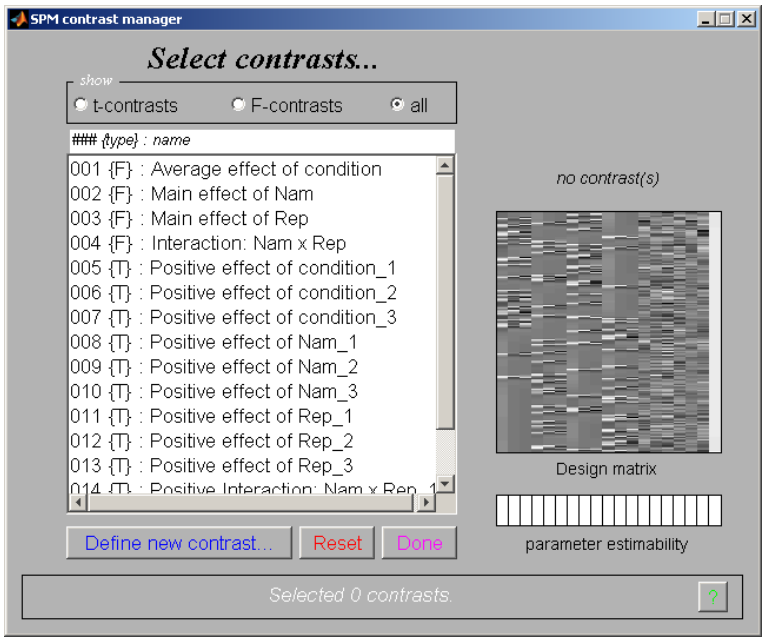


Figure 29.11: *Contrast Manager containing default contrasts for categorical design.*

29.2.1 Estimate

Press the ESTIMATE button. This will call up the specification of an fMRI estimation job in the batch editor window. Then

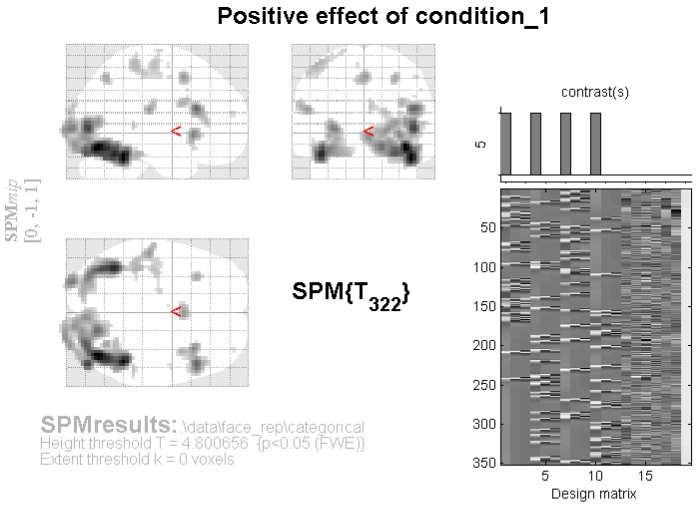
- Highlight the "Select SPM.mat" option and then choose the `SPM.mat` file saved in the `DIR/categorical` directory.
- Save the job as `categorical_est.job` and press `Run` button.

SPM will write a number of files into the selected directory including an `SPM.mat` file.

29.2.2 Inference for categorical design

Press "Results" and select the SPM.mat file from `DIR/categorical`. This will again invoke the contrast manager. Because we specified that our model was using a "Factorial design" a number of contrasts have been specified automatically, as shown in Figure 29.11.

- Select contrast number 5. This is a t-contrast `Positive effect of condition_1` This will show regions where the average effect of presenting faces is significantly positive, as modelled by the first regressor (hence the _1), the canonical HRF. Press 'Done".
- *Mask with other contrast ? [Yes/No]*
- Specify No.
- *Title for comparison ?*
- Enter "Canonical HRF: Faces > Baseline"
- *p value adjustment to control: [FWE/FDR/none]*
- Select FWE
- *Corrected p value(family-wise error)*
- Accept the default value, 0.05



Statistics: *p-values adjusted for search volume*

set-level		cluster-level				peak-level					mm mm mm
p	c	$p_{FWE-corr}$	$q_{FDR-corr}$	k_E	p_{uncorr}	$p_{FWE-corr}$	$q_{FDR-corr}$	T	$(Z_{\equiv})$	p_{uncorr}	
0.000	**16**	**0.000**	**0.000**	**1338**	**0.000**	**0.000**	**0.000**	**14.45**	**Inf**	**0.000**	**39 -70 -14**
						0.000	0.000	14.04	Inf	0.000	45 -46 -23
						0.000	0.000	11.25	Inf	0.000	48 -79 4
		0.000	**0.000**	**466**	**0.000**	**0.000**	**0.000**	**12.59**	**Inf**	**0.000**	**-42 -55 -20**
						0.000	0.000	11.13	Inf	0.000	-39 -67 -20
						0.000	0.000	8.95	Inf	0.000	-33 -79 -14
		0.000	**0.000**	**160**	**0.000**	**0.000**	**0.000**	**9.95**	**Inf**	**0.000**	**48 23 22**
						0.000	0.000	6.89	6.65	0.000	36 11 28
		0.000	**0.000**	**88**	**0.000**	**0.000**	**0.000**	**8.96**	**Inf**	**0.000**	**-27 -94 4**
		0.000	**0.000**	**103**	**0.000**	**0.000**	**0.000**	**8.55**	**Inf**	**0.000**	**33 20 -2**
						0.001	0.017	5.76	5.61	0.000	54 17 1
						0.004	0.082	5.41	5.29	0.000	51 20 -14
		0.000	**0.000**	**50**	**0.000**	**0.000**	**0.000**	**7.94**	**7.58**	**0.000**	**-54 -22 22**
		0.000	**0.000**	**54**	**0.000**	**0.000**	**0.000**	**7.71**	**7.38**	**0.000**	**0 11 52**
		0.000	**0.000**	**26**	**0.000**	**0.000**	**0.000**	**7.16**	**6.89**	**0.000**	**-33 23 -2**
		0.000	**0.001**	**24**	**0.000**	**0.000**	**0.000**	**7.02**	**6.76**	**0.000**	**30 -61 52**
		0.000	**0.000**	**146**	**0.000**	**0.000**	**0.000**	**6.61**	**6.39**	**0.000**	**-45 -34 61**
						0.000	0.002	6.24	6.06	0.000	-39 -16 64
						0.000	0.003	6.14	5.96	0.000	-51 -28 55

table shows 3 local maxima more than 8.0mm apart

Height threshold: T = 4.80, p = 0.000 (0.050)
Extent threshold: k = 0 voxels, p = 1.000 (0.050)
Expected voxels per cluster, <k> = 1.438
Expected number of clusters, <c> = 0.05
FWEp: 4.801, FDRp: 5.600, FWEc: 1, FDRc: 12

Degrees of freedom = [1.0, 322.0]
FWHM = 10.7 10.8 10.8 mm mm mm; 3.6 3.6 3.6 {voxels}
Volume: 1611441 = 59683 voxels = 1154.0 resels
Voxel size: 3.0 3.0 3.0 mm mm mm; (resel = 46.30 voxels)
Page 1

Figure 29.12: ***MIP and Volume table for Canonical HRF**: Faces > Baseline.*

- *Extent threshold {voxels} [0]*
- Accept the default value, 0.

SPM will then produce the MIP shown in Figure 29.12.

29.2.3 Statistical tables

To get a summary of local maxima, press the "whole brain" button in the p-values section of the interactive window. This will list all clusters above the chosen level of significance as well as separate (>8mm apart) maxima within a cluster, with details of significance thresholds and search volume underneath, as shown in Figure 29.12

The columns in volume table show, from right to left:

- **x, y, z (mm)**: coordinates in MNI space for each maximum.
- **peak-level**: the chance (p) of finding (under the null hypothesis) a peak with this or a greater height (T- or Z-statistic), corrected (FWE or FDR)/ uncorrected for search volume.
- **cluster-level**: the chance (p) of finding a cluster with this many(ke) or a greater number of voxels, corrected (FWE or FDR)/ uncorrected for search volume.

- **set-level**: the chance (p) of finding this (c) or a greater number of clusters in the search volume.

Right-click on the MIP and select "goto global maximum". The cursor will move to [39 -70 -14]. You can view this activation on the subject's normalised, attenuation-corrected structural (`wmsM03953_0007img`), which gives best anatomical precision, or on the normalised mean functional (`wmeansM03953_0005_0006.img`), which is closer to the true data and spatial resolution (including distortions in the functional EPI data).

If you select "plot" and choose "Contrast of estimates and 90% C.I" (confidence interval), and select the "Average effect of condition" contrast, you will see three bars corresponding to the parameter estimates for each basis function (summed across the 4 conditions). The BOLD impulse response in this voxel loads mainly on the canonical HRF, but also significantly (given that the error bars do not overlap zero) on the temporal and dispersion derivatives (see next Chapter).

29.2.4 F-contrasts

To assess the main effect of repeating faces, as characterised by both the hrf *and* its derivatives, an F-contrats is required. This is really asking whether repetition changes the *shape* of the impulse response (e.g, it might affect its latency but not peak amplitude), at least the range of shapes defined by the three basis functions. Because we have told SPM that we have a factorial design, this required contrast will have been created automatically - it is number 3.

- Press "Results" and select the `SPM.mat` file in the `DIR/categorical` directory.
- Select the "F-contrast" toggle and the contrast number 3, as shown in Figure 29.13. Press "Done".
- *Mask with other contrast ? [Yes/No].*
- Specify "Yes".
- Select contrast 5 - `Positive effect of condition_1` (the T-contrast of activation versus baseline, collapsed across conditions, that we evaluated above)
- *uncorrected mask p-value ?*
- Change to 0.001
- *nature of mask?*
- Select 'inclusive'
- *Title for comparison ?*
- Keep "Main effect of Rep (masked with ...)"
- *p value adjustment to control: [FWE/none]*
- Select none
- *threshold (F or p value)*
- Accept the default value, 0.001
- *Extent threshold {voxels} [0]*
- Accept the default value, 0

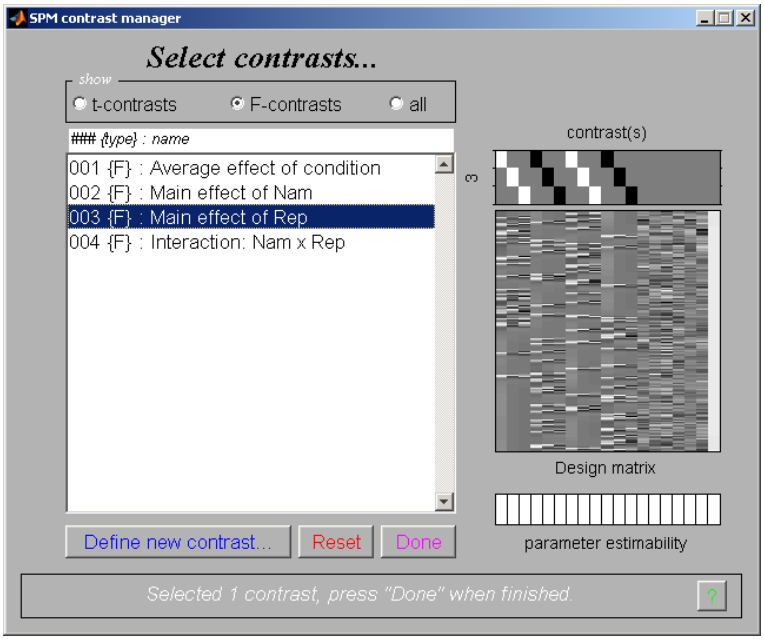


Figure 29.13: *Contrast manager showing selection of the first contrast "Main effect of Rep" (repetition: F1 and N1 vs F2 and N2)*

A MIP should then appear, the top half of which should look like Figure 29.14.

Note that this contrast will identify regions showing any effect of repetition (e.g, decreased or increased amplitudes) *within* those regions showing activations (on the canonical HRF) to faces versus baseline (at p<.05 uncorrected). Select "goto global max", which is in right ventral temporal cortex [42 -64 -8].

If you press plot and select "Event-related responses", then "F1", then "fitted response and PSTH", you will see the best fitting linear combination of the canonical HRF and its two derivatives (thin red line), plus the "selectively-averaged" data (peri-stimulus histogram, PSTH), based on an FIR refit (see next Chapter). If you then select the "hold" button on the Interactive window, and then "plot" and repeat the above process for the "F2" rather than "F1" condition, you will see two estimated event-related responses, in which repetition decreases the peak response (ie F2<F1), as shown in Figure 29.14.

You can explore further F-contrasts, which are a powerful tool once you understand them. For example, the MIP produced by the "Average effect of condition" F-contrast looks similar to the earlier T-contrast, but importantly shows the areas for which the sums across conditions of the parameter estimates for the canonical hrf *and/or* its temporal derivative *and/or* its dispersion derivative are different from zero (baseline). The first row of this F-contrast ([1 0 0 1 0 0 1 0 0 1 0 0]) is also a two-tailed version of the above T-contrast, ie testing for both activations and deactivations versus baseline. This also means that the F-contrasts [1 0 0 1 0 0 1 0 0 1 0 0] and [-1 0 0 -1 0 0 -1 0 0 -1 0 0] are equivalent. Finally, note that an F- (or t-) contrast such as [1 1 1 1 1 1 1 1 1 1 1 1], which tests whether the mean of the canonical hrf AND its derivatives for all conditions are different from (larger than) zero is not sensible. This is because the canonical hrf and its temporal derivative may cancel each other out while being significant in their own right. The basis functions are really quite different things, and need to represent separate rows in an F-contrast.

29.2.5 F-contrasts for testing effects of movement

To assess movement-related activation

- Press "Results", select the `SPM.mat` file, select "F-contrast" in the Contrast Manager. Specify e.g. "Movement-related effects" (name) and in the "contrasts weights matrix" window,

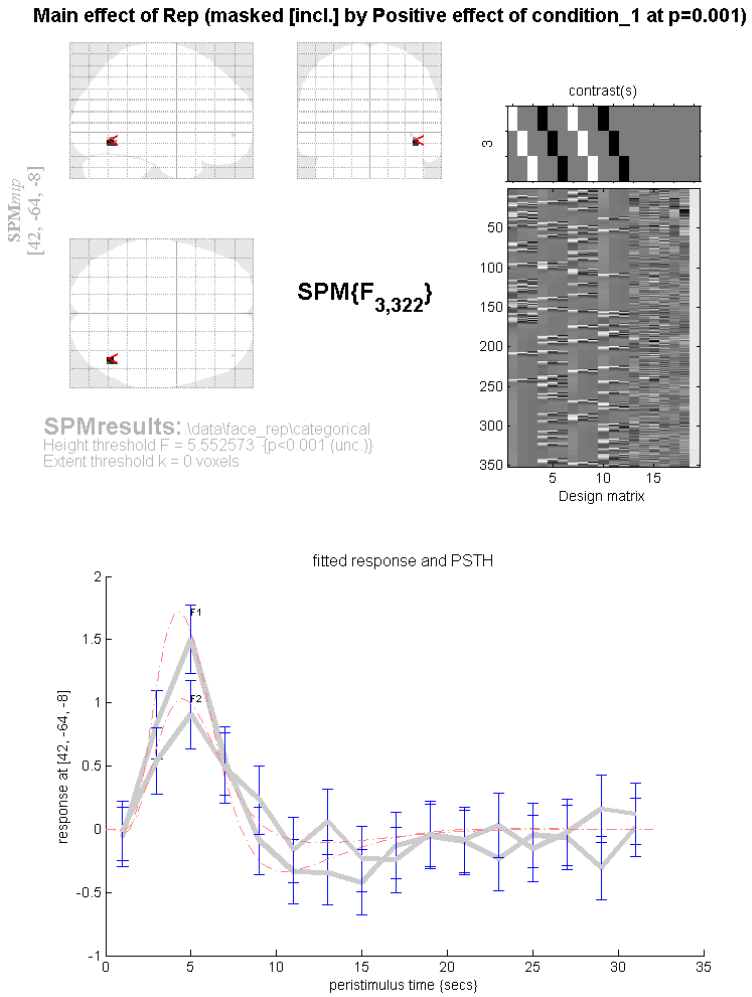


Figure 29.14: *MIP for Main effect of Rep, masked inclusively with Canonical HRF: Faces > Baseline at p<.001 uncorrected. Shown below are the best-fitting responses and peri-stimulus histograms (PSTH) for F1 and F2.*

or "1:12 19" in the "columns for reduced design" window.

- Submit and select the contrast, specify "mask with other contrasts?" (no), "title for comparison" (accept default), "corrected height threshold" (FWE), and "corrected p-value" (accept default).

- When the MIP appears, select "sections" from the "overlays" pulldown menu, and select the normalised structural image (`wmsM03953_0007.img`).

You will see there is a lot of residual movement-related artifact in the data (despite spatial realignment), which tends to be concentrated near the boundaries of tissue types (eg the edge of the brain; see Figure 29.15). (Note how the MIP can be misleading in this respect, since though it appears that the whole brain is affected, this reflects the nature of the (X-ray like) projections onto each orthogonal view; displaying the same datae as sections in 3D shows that not every voxel is suprathreshold.) Even though we are not interested in such artifact, by including the realignment parameters in our design matrix, we "covary out" (linear components) of subject movement, reducing the residual error, and hence improve our statistics for the effects of interest.

29.3 Modelling parametric responses

Before setting up the design matrix, we must first load into MATLAB the Stimulus Onsets Times (SOTs), as before, and also the "Lags", which are specific to this experiment, and which will be used as parametric modulators. The Lags code, for each second presentation of a face (N2 and F2), the number of other faces intervening between this (repeated) presentation and its previous (first) presentation. Both SOTs and Lags are represented by Matlab cell arrays, stored in the `sots.mat` file.

- At the MATLAB command prompt type `load sot`. This loads the stimulus onset times and the lags (the latter in a cell array called `itemlag`.

Now press the SPECIFY 1ST-LEVEL button. This will call up the specification of a fMRI specification job in the batch editor window. Then

- Press "Load" and select the `categorical_spec.mat` job file you created earlier.

- Open "Conditions" and then open the second "Condition".

- Highlight "Parametric Modulations", select "New Parameter".

- Highlight "Name" and enter "Lag", highlight values and enter `itemlag{2}`, highlight polynomial expansion and "2nd order".

- Now open the fourth "Condition" under "Conditions".

- Highlight "Parametric Modulations", select "New Parameter".

- Highlight "Name" and enter "Lag", highlight values and enter `itemlag{4}`, highlight polynomial expansion and "2nd order".

- Open "Canonical HRF" under "Basis Functions", highlight "Model derivatives" and select "No derivatives" (to make the design matrix a bit simpler for present purposes!).

- Highlight "Directory" and select `DIR/parametric` (having "unselected" the current definition of directory from the Categorical analysis).

- Save the job as `parametric_spec` and press the `Run` button.

This should produce the design matrix shown in Figure 29.16.

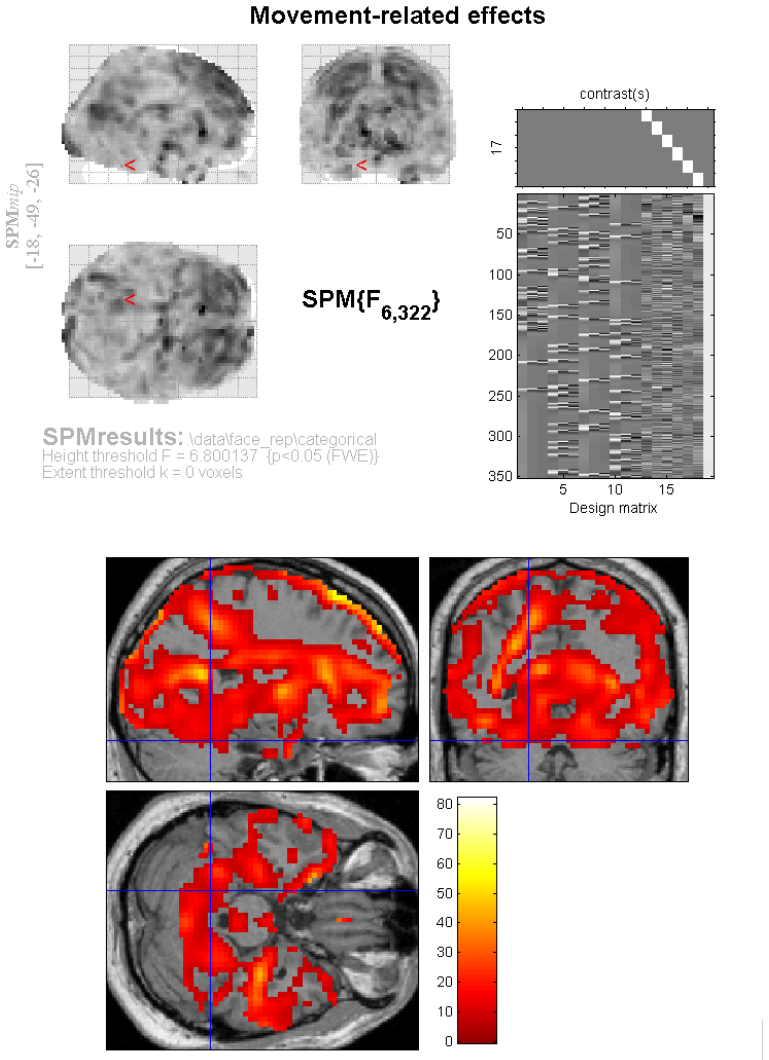


Figure 29.15: *Movement-related activations. These spurious 'activations' are due to residual movement of the head during scanning. These effects occur at tissue boundaries and boundaries between brain and non-brain, as this is where contrast differences are greatest. Including these regressors in the design matrix means these effects cannot be falsely attributed to neuronal activity.*

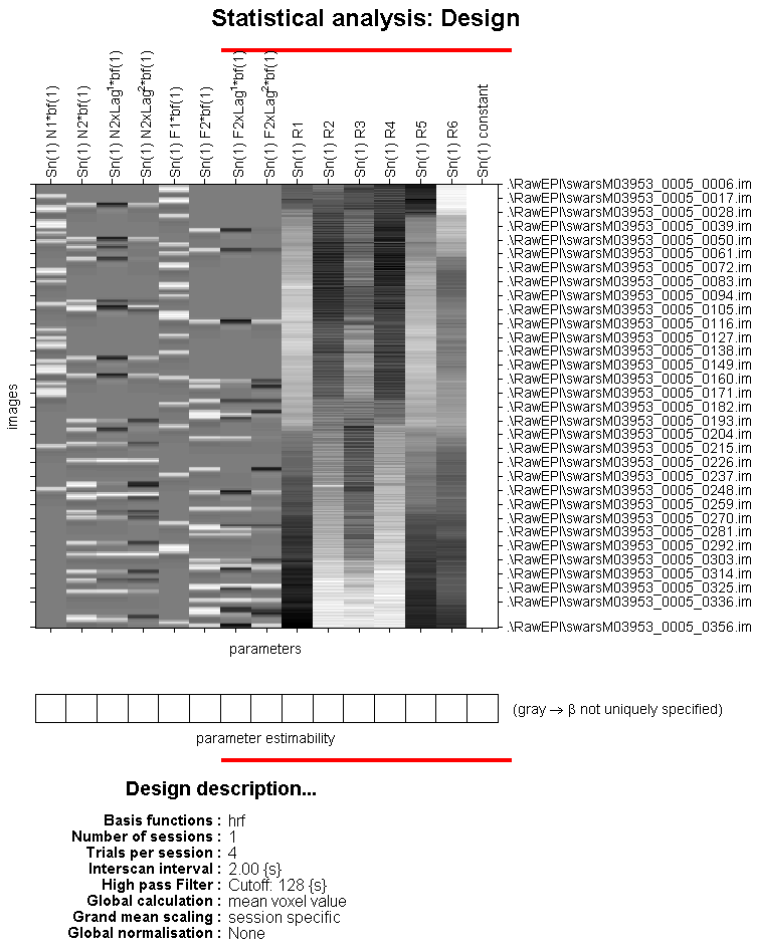


Figure 29.16: ***Design matrix for testing repetition effects parametrically.*** *Regressor 2 indicates the second occurence of a nonfamous face. Regressor 3 modulates this linearly as a function of lag (ie. how many faces have been shown since that face was first presented), and regressor 4 modulates this quadratically as a function of lag. Regressors 6,7 and 8 play the same roles, but for famous faces.*

29.3.1 Estimate

Press the ESTIMATE button. This will call up the specification of an fMRI estimation job in the batch editor window. Then

- Highlight the "Select SPM.mat" option and then choose the `SPM.mat` file saved in the `DIR/parametric` directory.
- Save the job as `parametric_est.job` and press the `Run` button.

SPM will write a number of files into the selected directory including an `SPM.mat` file.

29.3.2 Plotting parametric responses

We will look at the effect of lag (up to second order, ie using linear and quadratic terms) on the response to repeated Famous faces, within those regions generally activated by faces versus baseline. To do this

- Press "Results" and select the `SPM.mat` file in the `DIR/parametric` directory.
- Press "Define new contrast", enter the name "Famous Lag", press the "F-contrast" radio button, enter "1:6 9:15" in the "columns in reduced design" window, press "submit", "OK" and "Done".
- Select the "Famous Lag" contrast.
- *Mask with other contrast ? [Yes/No]*
- Specify "Yes".
- Select the "Positive Effect of Condition 1" T contrast.
- Change to an 0.05 uncorrected mask p-value.
- Nature of Mask ? inclusive.
- *Title for comparison ?*
- Accept what is offered
- *p value adjustment to control: [FWE/none]*
- Select None
- *Threshold {F or p value}*
- Accept the default value, 0.001
- *Extent threshold {voxels} [0]*
- Accept the default value, 0.

Figure 29.17 shows the MIP and an overlay of this parametric effect using overlays, sections and selecting the `wmsM03953_0007.img` image. The effect is plotted in the time domain in figure 29.18. This was obtained by

- Right clicking on the MIP and selecting "global maxima".
- Pressing Plot, and selecting "parametric responses" from the pull-down menu.
- Which effect ? select "F2".

This shows a quadratic effect of lag, in which the response appears negative for short-lags, but positive and maximal for lags of about 40 intervening faces (note that this is a very approximate fit, since there are not many trials, and is also confounded by time during the session, since longer lags necessarily occur later (for further discussion of this issue, see the SPM2 example analysis of these data on the webpage).

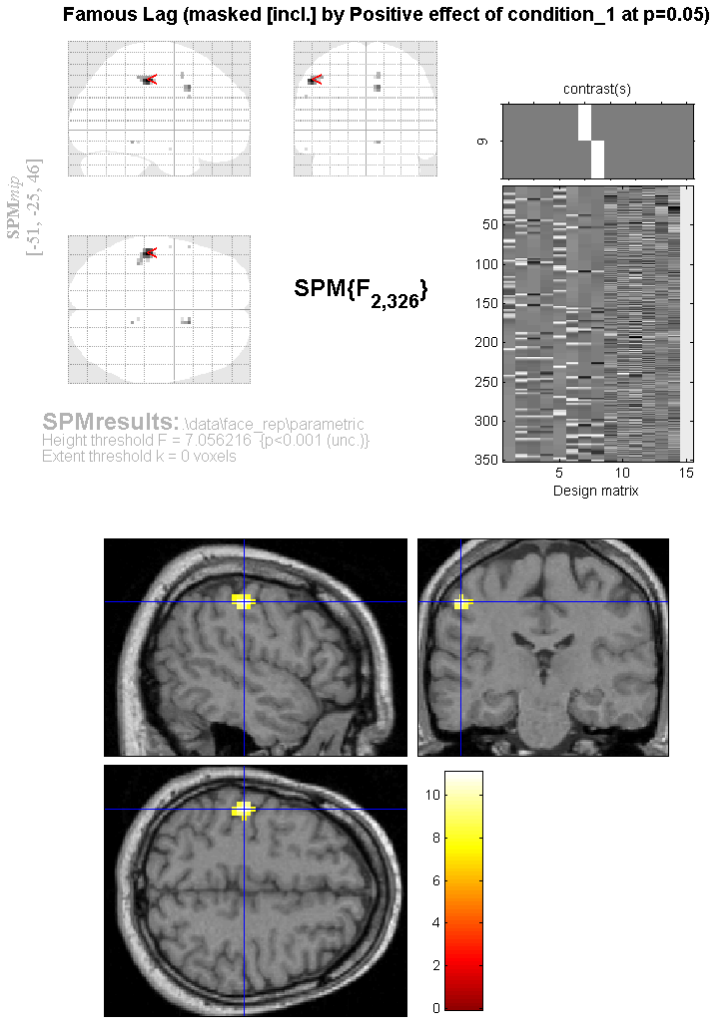


Figure 29.17: *MIP and overlay of parametric lag effect in parietal cortex.*

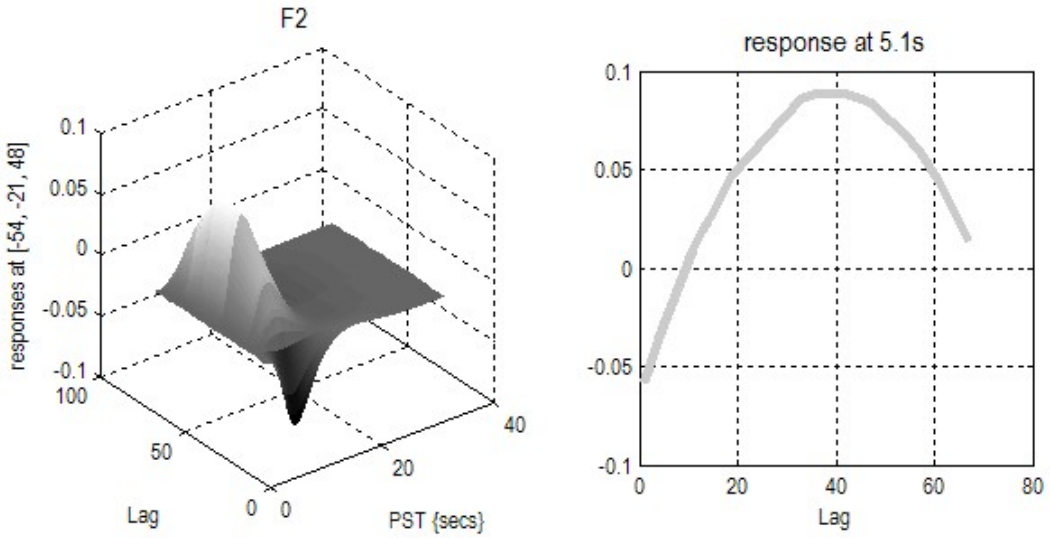


Figure 29.18: *Response as a function of lag.*

29.4 Bayesian analysis

29.4.1 Specification

Press the SPECIFY 1ST-LEVEL button. This will call up an fMRI specification job in the batch editor window. Then

- Load the `categorical_spec.mat` job file created for the classical analysis.
- Open "Subject/Session", highlight "Scans".
- Deselect the smoothed functional images using the 'unselect all' option available from a right mouse click in the SPM file selector (bottom window).
- Select the unsmoothed functional images using the `^wa.*` filter and "select all" option available from a right mouse click in the SPM file selector (top right window). The Bayesian analysis uses a spatial prior where the spatial regularity in the signal is estimated from the data. It is therefore not necessary to create smoothed images if you are only going to do a Bayesian analysis.
- Press "Done".
- Highlight "Directory" and select the `DIR/bayesian` directory you created earlier (you will first need to deselect the `DIR/categorical` directory).
- Save the job as `specify_bayesian.mat` and press the `Run` button.

29.4.2 Estimation

Press the ESTIMATE button. This will call up the specification of an fMRI estimation job in the batch editor window. Then

- Highlight the "Select SPM.mat" option and then choose the `SPM.mat` file saved in the `DIR/bayesian` subdirectory

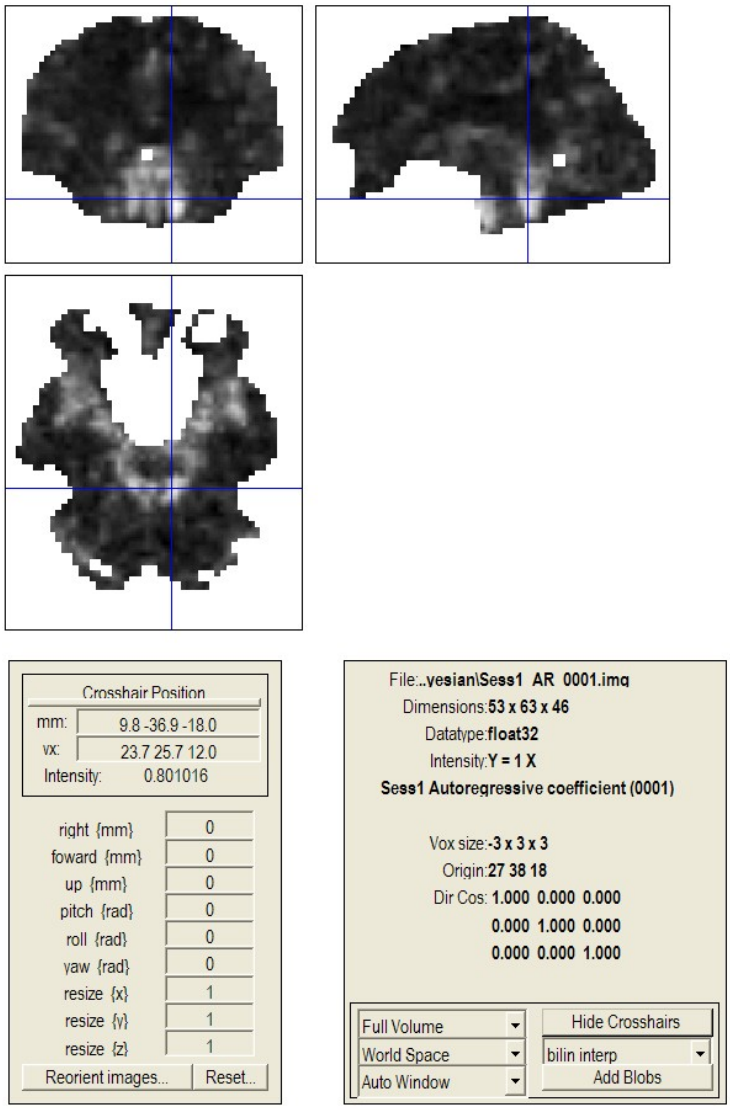


Figure 29.19: *Bayesian analysis: Estimated AR(1) coefficient image indicating heterogeneity near the circle of Willis*

- Highlight "Method" and select the "Choose Bayesian 1st-level" option.
- Save the job as `estimate_bayesian.job` and press the `Run` button.

SPM will write a number of files into the output directory including

- An `SPM.mat` file.
- Images `Cbeta_k.img` where k indexes the kth estimated regression coefficient. These filenames are prefixed with a "`C`"' indicating that these are the mean values of the "Conditional" or "Posterior" density.
- Images of error bars/standard deviations on the regression coefficients `SDbeta_k.img`.
- An image of the standard deviation of the error `Sess1_SDerror.img`.
- An image `mask.img` indicating which voxels were included in the analysis.
- Images `Sess1_AR_p.img` where p indexes the pth AR coefficient. See eg. Figure 29.19.

- Images `con_i.img` and `con_sd_i.img` which are the mean and standard deviation of the ith pre-defined contrast.

29.4.3 Inference

After estimation, we can make a posterior inference using a PPM. Basically, we identify regions in which we have a high probability (level of confidence) that the response exceeds a particular size (eg, % signal change). This is quite different from the classical inferences above, where we look for low probabilities of the null hypothesis that the size of the response is zero.

To determine a particular response size ("size threshold") in units of PEAK % signal change, we first need to do a bit of calculation concerning the scaling of the parameter estimates. The parameter estimates themselves have arbitrary scaling, since they depend on the scaling of the regressors. The scaling of the regressors in the present examples depends on the scaling of the basis functions. To determine this scaling, load the "SPM.mat" file and type in MATLAB `sf = max(SPM.xBF.bf(:,1))/SPM.xBF.dt` (alternatively, press "Design:Explore:Session 1" and select any of the conditions, then read off the peak height of the canonical HRF basis function (bottom left)).

Then, if you want a size threshold of 1% peak signal change, the value you need to enter for the PPM threshold (ie the number in the units of the parameter estimates) is `1/sf` (which should be 4.75 in the present case).[10]

Finally, if we want to ask where is there a signal greater than 1% (with a certain confidence) to faces versus baseline, we need to create a new contrast that takes the AVERAGE of the parameter estimates for the canonical HRF across the four conditions (N1 to F2), rather than the default `Positive effect of condition_1` contrast, which actually calculates the SUM of the parameter estimates for the canonical HRF across conditions (the average vs sum makes no difference for the classical statistics).

- Press "Results".
- Select the `SPM.mat` file created in the last section.
- Press "Define new contrast", enter the name "AVERAGE Canonical HRF: Faces > Baseline", press the "T-contrast" radio button, enter the contrast [1 0 0 1 0 0 1 0 0 1 0 0]/4, press "submit", "OK" and "Done".
- *Mask with other contrast ? [Yes/No]*
- Specify No
- *Title for comparison*
- Enter "AVERAGE Canonical HRF: Faces > Baseline"
- *Effect size threshold for PPM*
- Enter the value
- *Posterior probability threshold for PPM*
- Enter the value 0.95
- *Extent threshold [0]*
- Accept the default value
- *Plot effect size [Yes/No]*
- Select the default "Yes"

SPM will then plot a map of effect sizes at voxels where it is 95% sure that the effect size is greater than 1% of the global mean. Then use overlays, sections, select the normalised structural image created earlier and move the cursor to the activation in the left hemisphere. This should create the plot shown in Figure 29.20.

[10] Strictly speaking, this is the peak height of the canonical component of the best fitting BOLD impulse response: the peak of the complete fit would need to take into account all three basis functions and their parameter estimates.

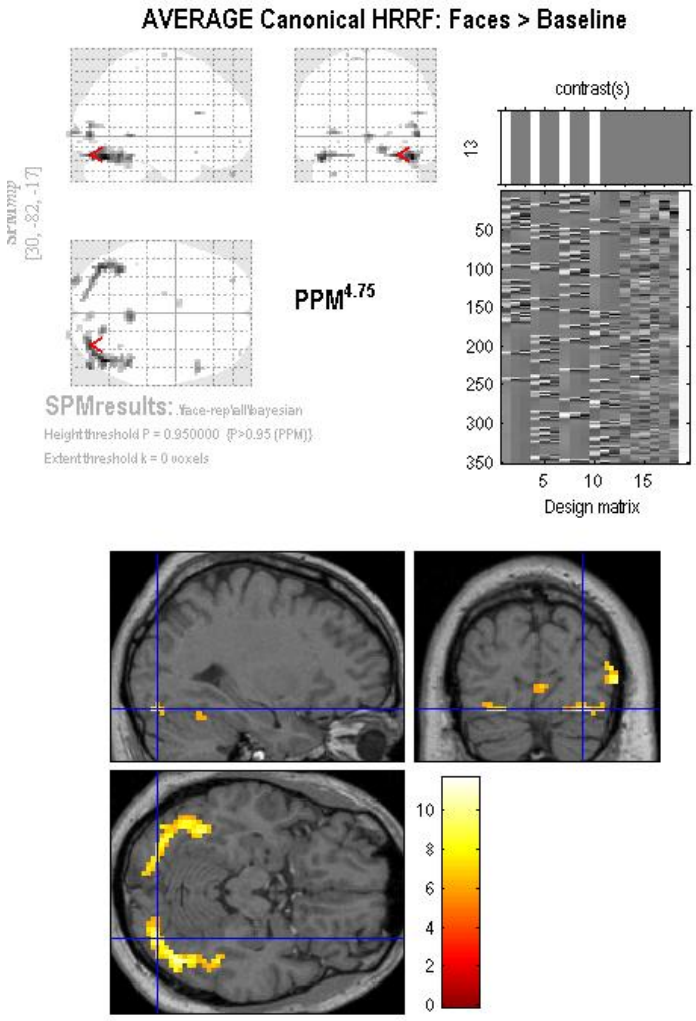


Figure 29.20: *Bayesian analysis: MIP and overlay of effect sizes at voxels where PPM is 95% sure that the effect size is greater than 1% of the global mean. The cursor is at the location* $x = 30, y = -82, z = -17mm$

Chapter 30

Face group data

30.1 Introduction

These examples illustrate multisubject "random effects" analyses or "second-level" models of fMRI data [62][1]. The examples consist of three basic types of 2nd-level model

1. **M2c:** Using contrast images for the canonical HRF only. This uses a single observation (contrast image) per subject only and data are analysed using a "One-sample t-test".

2. **M2i:** Using contrast images from an "informed" basis set, consisting of the canonical HRF and its two partial derivatives with respect to time (onset latency) and dispersion. This uses 3 observations (contrast images) per subject and data are analysed using a "One-way ANOVA" with 3 levels.

3. **M2f:** Using contrast images from a very general "Finite Impulse Response" (FIR) basis set, with 12 × 2 second timebins. This uses 12 observations (contrast images) per subject. Data are analysed using a "One-way ANOVA" with 12 levels.

30.2 Data

The data come from the "implicit" condition of the Henson et al. study [41]. Although the 1st-level design matrices (and therefore resulting contrast images) used do not correspond exactly to those used in that study.

It is also the same study from which one subject is used to illustrate a single-subject fixed effects analysis (see earlier Chapter in this manual).

Unlike the single-subject fixed effects example dataset, only two event-types were modelled: famous and nonfamous faces (initial and repeated presentations were collapsed together, as were correct and incorrect responses). Briefly, greyscale photographs of 52 famous and 52 nonfamous face were presented for 0.5s for fame judgment task (one of two right finger key presses). The minimal SOA (SOAmin) was 4.5s, with all faces randomly intermixed together with a further 52 null events (ie 2/3 probability of a face every SOAmin).

Original images were continuous EPI (TE=40ms,TR=2s) 24 descending slices (64×64 3×3 mm^2), 3mm thick, 1.5mm gap.

2nd-level models **M2c** and **M2i** derive from a 1st-level model (**M1i**), in which the events were modelled with Nf=3 basis functions: the canonical HRF, its partial derivative with respect to onset latency ("temporal derivative") and its partial derivative with respect to dispersion ("dispersion derivative").

2nd-level model **M2f** derives from an alternative 1st-level model (**M1f**), in which the same events were modelled with Nf=12 basis functions instead: corresponding to 2s timebins from 0-24s poststimulus (SPM's "Finite Impulse Response" or FIR basis set).

[1] This chapter has been largely cannibalised from an earlier document, available from http://www.fil.ion.ucl.ac.uk/spm/data/face_rfx/spm2_face_rfx.doc, which describes how to analyse this data using SPM2. That document additionally describes the analysis of differential effects, which we have omitted here.

In both first-level models (**M1i** and **M1f**), the contrast images (`con*.img`'s) come from session-specific contrasts within a large (multisession) 1st-level Fixed Effects design matrix, with one session per subject. (Note that the resulting `con*.img`'s could equally well have been produced from 12 separate 1st-level models, one per subject.)

For each type of model, the main effect of faces versus baseline (eg, a [0.5 ... 0.5] contrast for each basis function, or `kron(eye(Nf),[0.5 0.5])` more generally) was examined.

The 12 (subjects) `con*.img`s from the 1st-level model using the canonical HRF (**M1c**) are in the zipped file
http://www.fil.ion.ucl.ac.uk/spm/data/face_rfx/cons_can.zip.

The 12 (subjects) x 3 (basis functions) `con*.img`s from the 1st-level model using the informed basis (**M1i**) set are in the zipped file
http://www.fil.ion.ucl.ac.uk/spm/data/face_rfx/cons_informed.zip.

The 12 (subjects) x 12 (basis functions) x 2 (contrast-types) `con*.img`s from the 1st-level model using the FIR basis (**M1f**) set are in the zipped file
http://www.fil.ion.ucl.ac.uk/spm/data/face_rfx/cons_fir.zip.

Each contrast-type is examined in a separate SPM analysis. This chapter just describes analysis of the main effect of faces versus baseline. To analyse the data, first create a new directory `DIR` eg. `c:\data\face_group`, in which to place the results of your analysis. Then create 3 subdirectories (i) `Canonical`, (ii) `Informed`, and (iii) `FIR`. As the analysis proceeds these directories will be filled with job-specification files, design matrices and estimated models.

30.3 Canonical HRF

For the main effect versus baseline, these happen to correspond to the contrast images numbered 3-14 in 1st-level model `M1i`, ie:

- `con_0006.img` (canonical HRF, subject 1)
- `con_0007.img` (canonical HRF, subject 2)
- ...
- `con_0017.img` (canonical HRF, subject 12)

These images comprise the data for `M2c`, which is simply a "One-sample t-test". This can be implemented as follows.

- Start up MATLAB and type `spm fmri` at the prompt
- Press the 'Specify 2nd-level' button. This will open the batch editor.
- In the "Design", "One-sample t-test" option, select "Scans".
- Choose "Select Files" and use the SPM file selector to choose contrast images 6 to 17.
- Highlight "Directory", "Select files" and select the subdirectory "canonical", to place the design matrix in.
- Save the job file as eg. `DIR/canonical.mat`.
- Press the `Run` button (green arrow).

SPM will then show you the design matrix shown in Figure 30.1. This is simply a single column of 1's which will appear as a white box on a white background. This design is encoded in the `SPM.mat` file that is written to the output directory. Then press "Estimate", select the **SPM.mat** file just created, and press the `Run` button. SPM will now estimate the parameters, that is, the size of the population effect at each voxel. This is simply the average of the `con*.img`s you have specified.

- Now press the "Results" button.

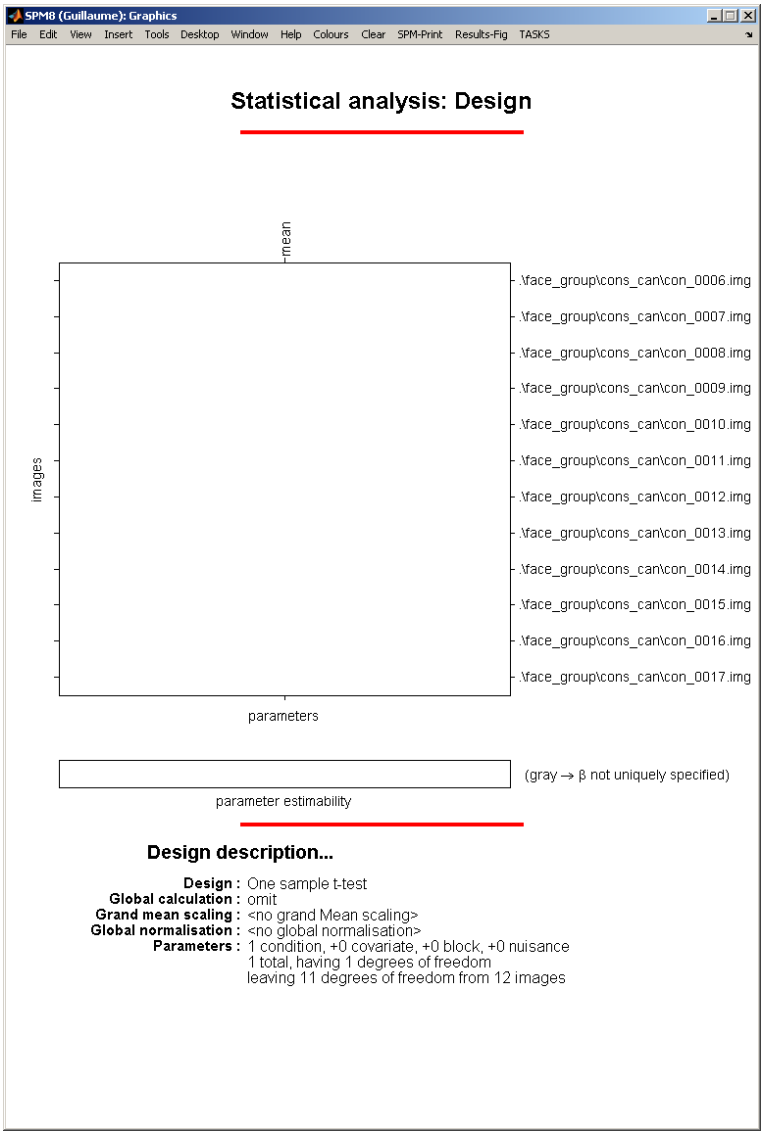


Figure 30.1: ***Design matrix for canonical responses.*** *This corresponds to a one-sample t-test.*

- Select the `SPM.mat` file.
- In the contrast manager press "Define new contrast" (select F). Enter [1] in the contrast section and enter "Faces vs Baseline: Canonical HRF' as a "name". Note: This [1] F-contrast tests for both "activations" and "deactivations" versus the interstimulus baseline, though in the present case, the regions are nearly all activations, as can be seen by entering the same contrast weight [1], but as a T rather than F contrast.
- Press the "..submit" button. Press OK.
- Now press the "Done" button.
- Mask with other contrast(s) [No]
- Title for comparison: accept [Faces vs Baseline: Canonical HRF]
- p value adjustment to control [FWE]
- Family-wise p-value [0.05]
- Extent threshold voxels [0]

SPM will now display the thresholded F-statistic image. This shows voxels that are significantly active (correcting for multiple comparisons across all voxels) in the population from which the subjects were drawn. They include bilateral posterior fusiform, SMA, and, at a more liberal threshold, left motor cortex). You can then press the volume to get a table of stastical information for clusters of activated voxels. SPM's graphics window should look like Figure 30.2.

30.4 Informed basis set

For this example, 3 contrast images per subject are taken to the 2nd-level. These are

- `con_0003.img` (canonical HRF, subject 1)
- `con_0004.img` (canonical HRF, subject 2)
- ...
- `con_0014.img` (canonical HRF, subject 12)
- `con_0015.img` (temporal derivative, subject 1)
- `con_0016.img` (temporal derivative, subject 2)
- ...
- `con_0026.img` (temporal derivative, subject 12)
- `con_0027.img` (dispersion derivative, subject 1)
- `con_0028.img` (dispersion derivative, subject 2)
- ...
- `con_0038.img` (dispersion derivative, subject 12)
- ...

These images comprise the data for `M2c`, which is simply a "One-way ANOVA" with 3-levels. This can be implemented as follows.

- Press the "Specify 2nd-level" button.
- In "Factorial design specification", highlight "Design" and then choose "Full Factorial".

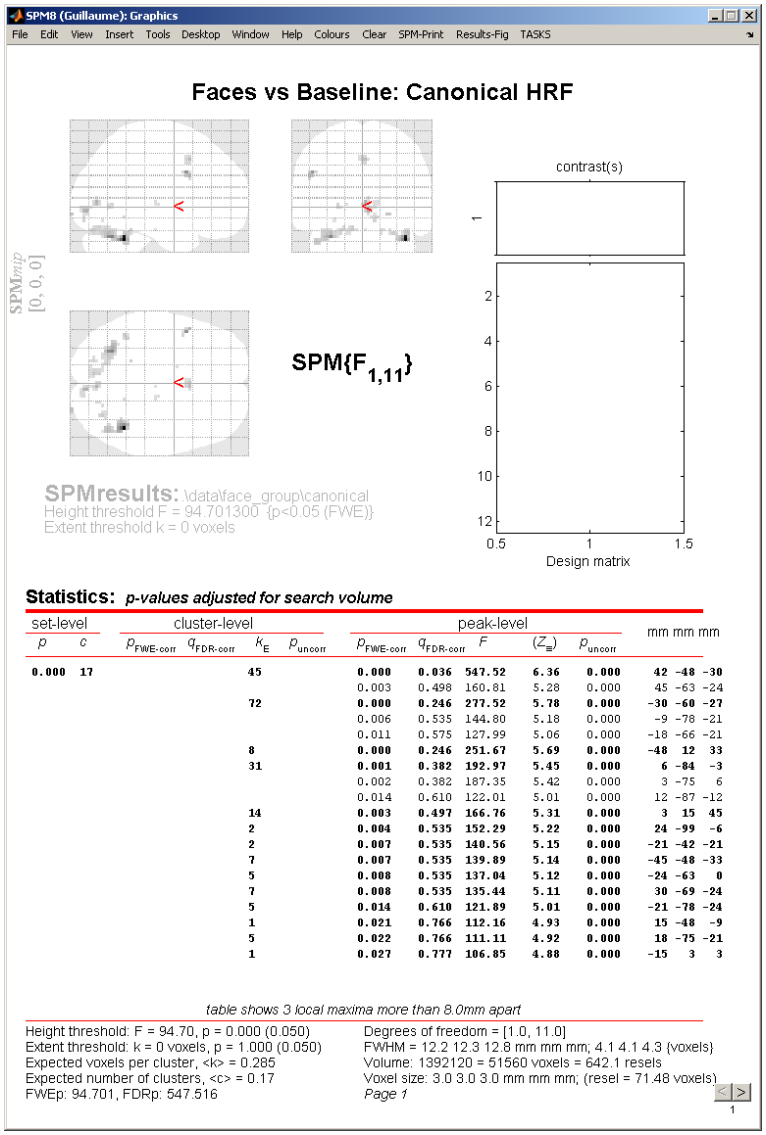


Figure 30.2: *Main population effect of faces vs baseline, as characterised using the Canonical HRF.*

- Under "Factors" create a single "New Factor".
- In this "Factor", type in "Basis" for Name and enter 3 under "Levels".
- Highlight "Independence" and select "No". SPM will then take into account possible correlations between these repeated measures (see section on Nonsphericity below for further discussion).
- Now highlight "Specify cells", and create 3 new cells.
- For the first cell, set "Levels" to `1`, and enter the canonical contrast images under scans (ie contrast images numbered `0003` to `0014`).
- For the second cell, set "Levels" to `2`, and enter the temporal derivative contrast images under scans (ie contrast images numbered `0015` to `0026`).
- For the third cell, set "Levels" to `3`, and enter the dispersion derivative contrast images under scans (ie contrast images numbered `0027` to `0038`.
- Highlight "Directory", "Specify files" and select the subdirectory "informed", to place the design matrix in.
- Save the job file as eg. `DIR/informed.mat`.
- Press the `Run` button in the batch editor.

SPM will then show you the design matrix shown in Figure 30.3. This design is encoded in the `SPM.mat` file that is written to the output directory. Then press "Estimate", select the `SPM.mat` file just created, and press the `Run` button. SPM will now estimate the parameters of the model (and hyperparameters governing the nonsphericity).

30.4.1 Nonsphericity

Setting the independence option described above to "No" allows SPM to take into account possible correlations between levels of the factor. Note that, by default, SPM assumes different variances for different levels of the factor (you can change this by setting "Variance" to "Equal" under the options for the factor).

In this way SPM can account for possible "non-sphericity" in the data. This is implemented in SPM using a set of matrices (bases) that characterise the covariance matrix. The first three correspond to the variance of each of the canonical, temporal and dispersion derivatives: `SPM.xVi.Vi{1}`, `SPM.xVi.Vi{2}`, and `SPM.xVi.Vi{3}`.

The next three correspond to covariances: `SPM.xVi.Vi{4}` (covariance between canonical and temporal derivative), `SPM.xVi.Vi{5}` (covariance between canonical and dispersion derivative), and `SPM.xVi.Vi{6}` (covariance between temporal and dispersion derivatives).

After estimation the actual covariance values (hyper-parameters) are given by `SPM.xVi.h` (the six entries correspond to the above bases). The corresponding estimated covariance matrix can be shown by pressing Review→Design→Explore→Covariance Structure. The estimated covariance for this data is shown in Figure 30.4. Note that these are "global" values which are scaled by a voxel specific-value to achieve a model covariance that best matches the empirical covariance at each voxel.

30.4.2 Informed Results

- Now press the "Results" button.
- Select the `SPM.mat` file.
- In the Contrast Manager press "Define new contrast" (select F). Enter `eye(3)` in the contrast section and enter "Faces vs Baseline: Informed" as a "name". Note: In MATLAB `eye(3)` evaluates to the identity matrix [1 0 0; 0 1 0; 0 0 1].[2]

[2]SPM will have produced some contrasts automatically, one of them being the "main effect of basis". This contrast is, however, not appropriate for our purposes.

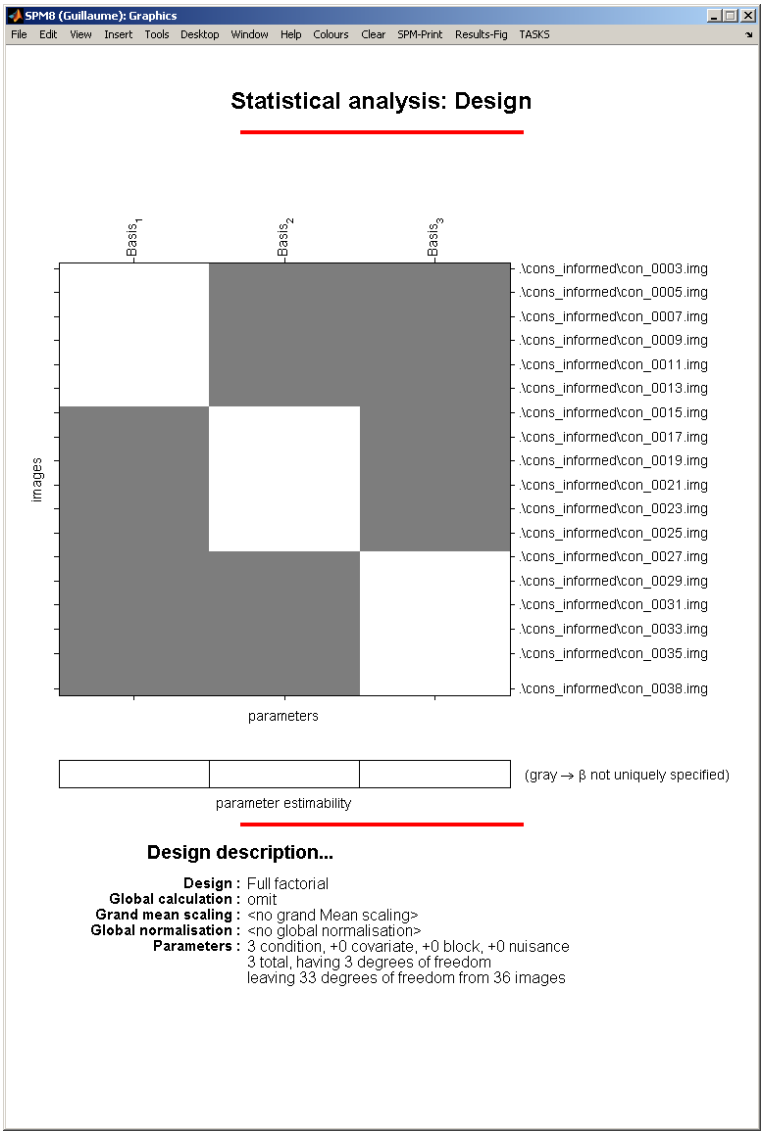


Figure 30.3: ***Design matrix for informed basis set.*** *This corresponds to a one-way ANOVA with three levels (but no constant term, since we want to test whether the basis functions are different from zero, not whether they are different from each other).*

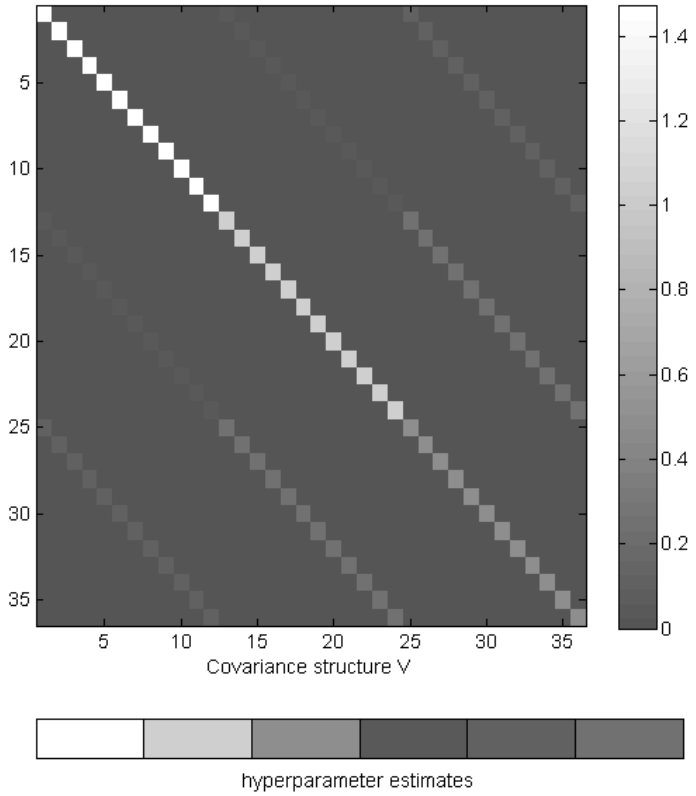


Figure 30.4: ***Estimated covariance matrix for informed basis set.*** *The 6 differently valued hyperparameters are shown in different shades of gray.*

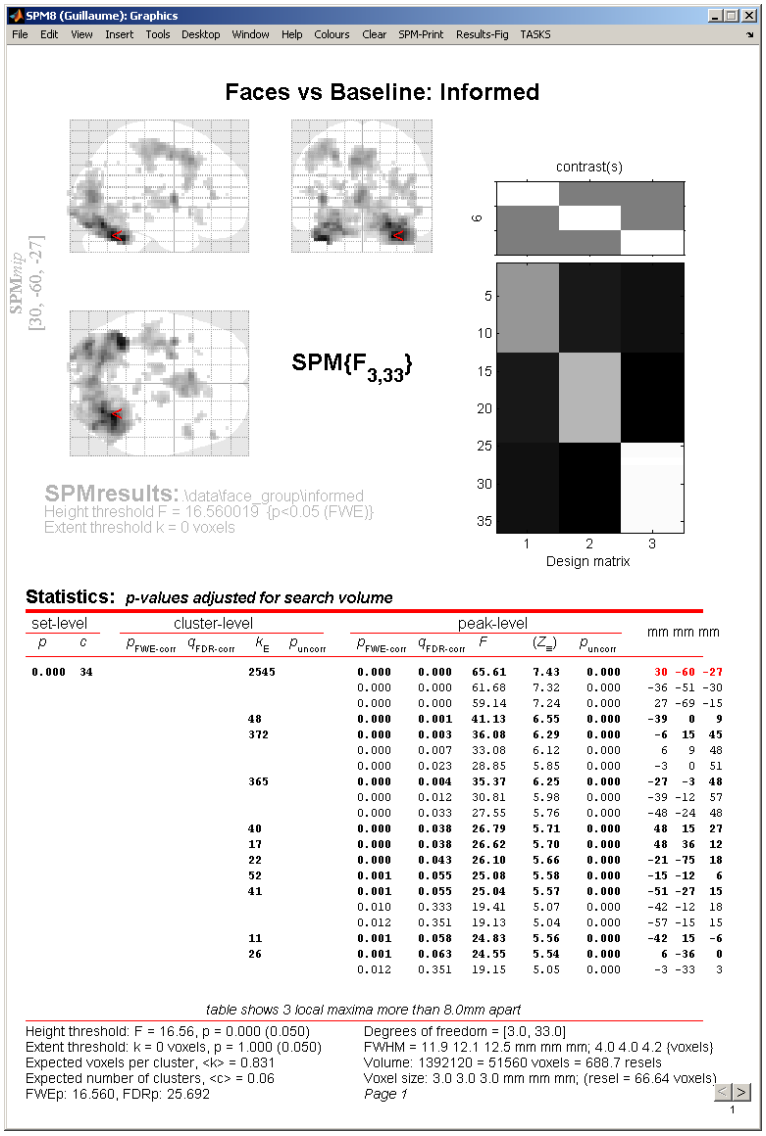


Figure 30.5: *Main population effect of faces, as characterised with the informed basis set.*

- Press the "..submit" button. Press OK.
- Now press the "Done" button.
- Mask with other contrast(s) [No]
- Title for comparison: accept [Faces vs Baseline: Informed]
- p value adjustment to control [FWE]
- Family-wise p-value [0.05]
- Extent threshold voxels [0]

This contrast will reveal voxels that show some form of event-related response that can be captured by (ie, lies in the space spanned by) the three basis functions (e.g, 30 -60 -27, Z=7.43), as shown in Figure 30.5.

Note how the design matrix appears to be different after estimation. This is because it has been pre-whitened (via the estimated nonsphericity). In particular, the (barely visible) off-diagonal entries in the design matrix give an indication of the degree of correlation between

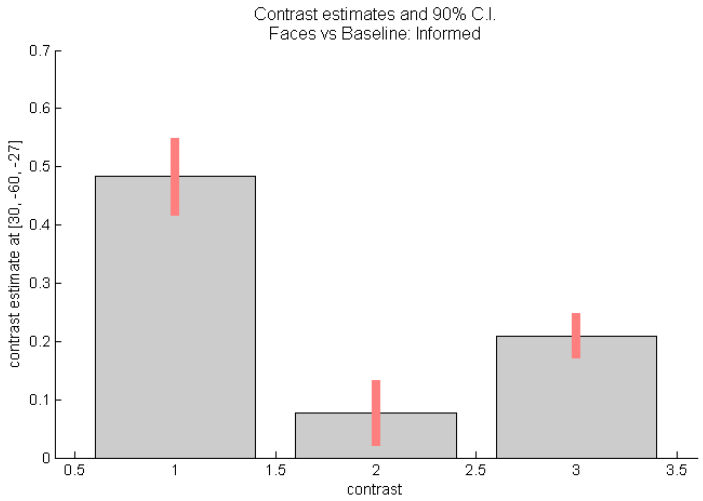


Figure 30.6: *Plotting the three basis functions for the global maximum showing reliable effects of the canonical HRF and its time and dispersion derivatives.*

the basis functions across subjects. However, because the data have also been pre-whitened our interpretation of the parameter estimates (the "betas") is unchanged. Effectively the parameters have been estimated using "Weighted Least Squares (WLS)", where the weights relate to the estimated error covariance structure. SPM implements WLS by pre-whitening the data and the design matrix and then using "Ordinary Least Squares" (OLS).

Note also how this F-contrast (Figure 30.5) produces more significant results than the corresponding F-contrast in the model with the canonical HRF shown in Figure 30.2. This suggests significant additional information in the two derivatives of the canonical HRF. If you right-click on the MIP and select "goto global maxima", then press "plot", select "Contrast estimates and 90% C.I.", and select the "Faces vs Baseline: Informed" contrast, you will get three bars and their confidence intervals, as in Figure 30.6. You can see that the canonical HRF (first bar) carries most of the response vs baseline, but nonetheless, both the temporal and dispersion derivatives (second and third bars) contribute significant additional effects (given that the error bars do not overlap zero). Note that the size of the bars cannot be compared directly since they depend on the (different) scaling of the three basis functions (their size RELATIVE TO the error bars is a fairer way to compare the contributions of the different basis functions).

30.4.3 T- and F-contrasts

It is also informative to evaluate the T-contrast [1 0 0] (ie positive loadings on the canonical HRF only). This is shown in Figure 30.7.

At a FWE correct p-value of 0.05, note more voxels (including now left motor cortex) and higher Z-values (e.g, 39 -57 -30, Z=7.53) for this main effect vs baseline compared to the equivalent T-contrast ([1]) in the model that uses only the canonical HRF (as in previous Section). The main reason for this increased power is the increase in the degrees of freedom, which entails better estimators of the underlying error (co)variance. The price of this increased power is a stronger assumption about the nonsphericity, namely that it has the same structure across (activated) voxels - the "pooling device", see Glaser et al. (2003) [36].

Finally, evaluate the F-contrasts [0 1 0] and [0 0 1]. These are shown in Figures 30.8 and 30.9. These contrasts reveal voxels that load (positively or negatively) on the temporal and dispersion derivatives respectively. These contrasts reveal that there is significant variability (at $p<.05$ corrected) that is not captured by the canonical HRF alone (see below for more discussion; see also to Henson et al (2000) [40].

In other words, some regions have earlier or later, or wider or narrower, BOLD impulse responses than the canonical HRF. This may reflect differences in vasculature (or even face-

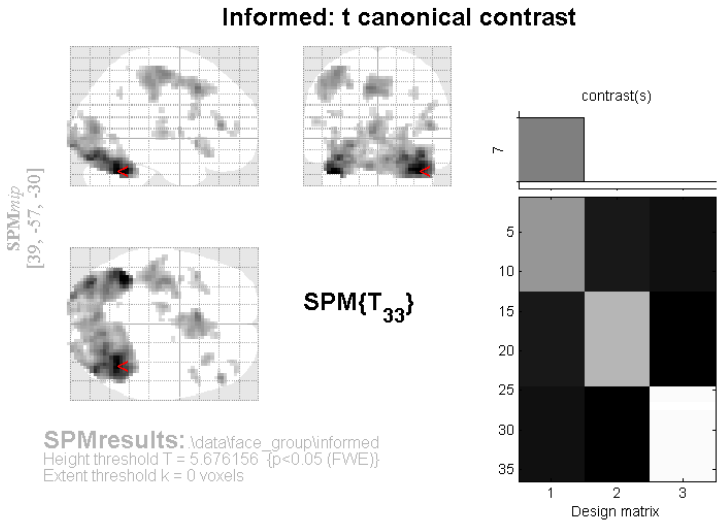


Figure 30.7: *Main population effect of faces, as characterised with the canonical HRF using a [1 0 0] t-contrast on the informed basis coefficients.*

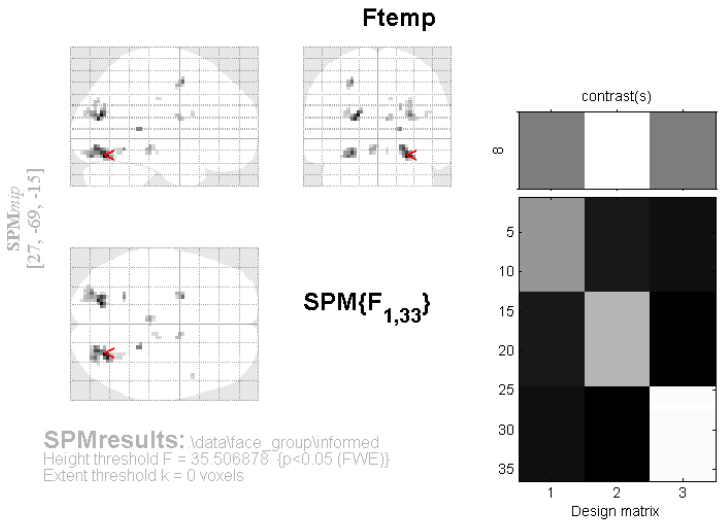


Figure 30.8: *Significantly non-zero temporal derivative coefficients. These voxels show responses earlier or later than canonical responses.*

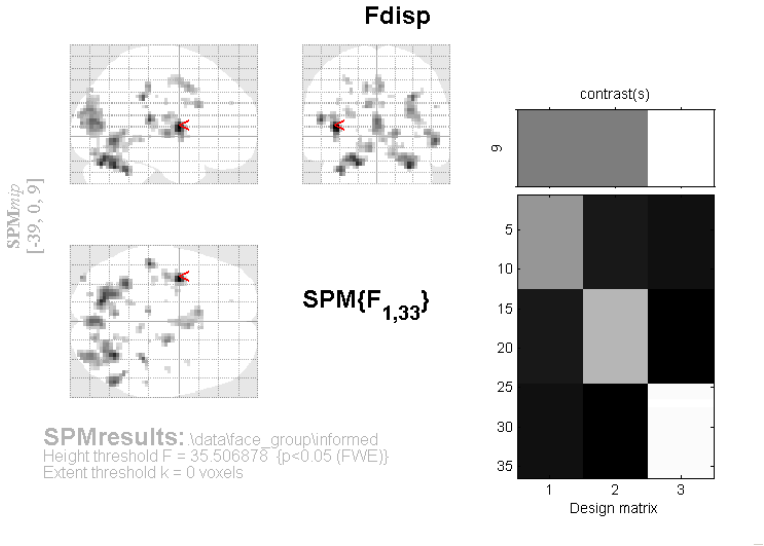


Figure 30.9: *Significantly non-zero dispersion derivative coefficients. These voxels show responses narrower or wider than canonical responses.*

related neural differences across regions).

On the other hand, note that most voxels in the above F-contrasts also show a positive loading on the canonical HRF (ie the previous [1 0 0] T-contrast), as can be revealed by Inclusive (or Exclusive) masking of the relevant contrasts. This is because the loadings on the derivatives reflect deviations ABOUT the canonical form (via a first-order Taylor expansion; see eg. Henson et al, 2002 [39]). Indeed, loadings on either derivative in the absence of a reliable loading (positive or negative) on the canonical HRF would be difficult to interpret (i.e, the derivative waveforms are probably too high frequency to reflect BOLD changes on their own).

One can also confirm this by going to various voxels in the above F-contrasts, pressing "plot", "contrast estimates" and selecting the "Can+Tem+Dis" F-contrast. The three bars indicate the loadings (and 90% confidence intervals) on the three different basis functions. Note that a positive estimate for the temporal derivative corresponds to an earlier response than the canonical (and negative for later), while a positive estimate for the dispersion derivative corresponds to a narrower (less dispersed) response (and negative for wider).

30.5 FIR basis set

For this example, 12 contrast images per subject are taken to the 2nd-level. These are the contrast images:

- `con_fir_bin01_sub01.img` (FIR bin 1, subject 1)
- `con_fir_bin01_sub02.img` (FIR bin 1, subject 2)
- ...
- `con_fir_bin02_sub01.img` (FIR bin 2, subject 1)
- ...

These images comprise the data for `M2f`, which is simply a "One-way ANOVA" with 12-levels (one for each time-bin). This can be implemented as follows.

- Start up MATLAB and type `spm fmri` at the prompt.
- Press the "Specify 2nd-level" button.

- The options for "Factorial design specification"[3] appear.
- Highlight "Design" and then choose "Full Factorial".
- Under "Full Factorial" and 'Factors' create a single "New Factor".
- In this "Factor", type in "TimeBin" for "Name" and enter 12 under "Levels".
- Highlight "Independence" and select "No". SPM will then take into account possible correlations between these repeated measures.
- Now highlight "Specify cells", and create 12 new cells.
- For the first cell, set "Levels" to `1`, and enter the contrast images for time bin 1 under scans. This is most easily done by changing the filter to `.*fir_bin01.*`.
- For the second cell, set "Levels" to `2`, and, under scans, enter the contrast images for time bin 2 This is most easily done by changing the filter to `.*fir_bin02.*`.
- Similarly for Levels 3 to 12.
- Highlight "Directory", "Specify files" and select the subdirectory `FIR`, to place the design matrix in.
- Save the job file as eg. `DIR/fir.mat`.
- Press the `Run` button in the batch editor.

SPM will then show you the design matrix shown in Figure 30.10. This design is encoded in the `SPM.mat` file that is written to the output directory. Then press "Estimate", select the `SPM.mat` file just created, and press the button `Run`. SPM will now estimate the parameters of the model.

30.5.1 Nonsphericity again

Setting the independence option to "No" allows SPM to take into account possible correlations between levels of the factor. Note that, by default, SPM assumes different variances for different levels of the factor (you can change this by setting "Variance" to "Equal" under the options for the factor).

In this way SPM can account for possible "non-sphericity" in the data. This is implemented in SPM using a set of matrices (bases) that characterise the covariance matrix. The first 12 correspond to the variance of each of the responses in each of the 12 time bins. The ones that follow correspond to covariances between different time bins.

After estimation the actual covariance values (hyper-parameters) are given by `SPM.xVi.h`. The corresponding estimated covariance matrix can be shown by pressing Review→Design→Explore→Covariance Structure. The estimated covariance for this data is shown in Figure 30.11. Note that these are "global" values which are scaled by a voxel specific-value to achieve a model covariance that best matches the empirical covariance at each voxel.

You can see the highest values on the leading diagonal occur for timebins 2-4 (scans 13-48). This is where the peak response occurs, and the large values imply that, as expected, the variance tends to increase with the mean. This "inhomogeniety of variance" is a problem for conventional ANOVAs, but not here, where it is explicitly modelled.

Notice also the high values close to the diagonal, which reflect the positive correlation between the error across adjacent timebins (as also expected).

[3] In SPM2, this data was analysed using the "One-way ANOVA without a constant" design. This option is no longer available in SPM5, as one-way ANOVA's are considered as factorial designs with a single factor.

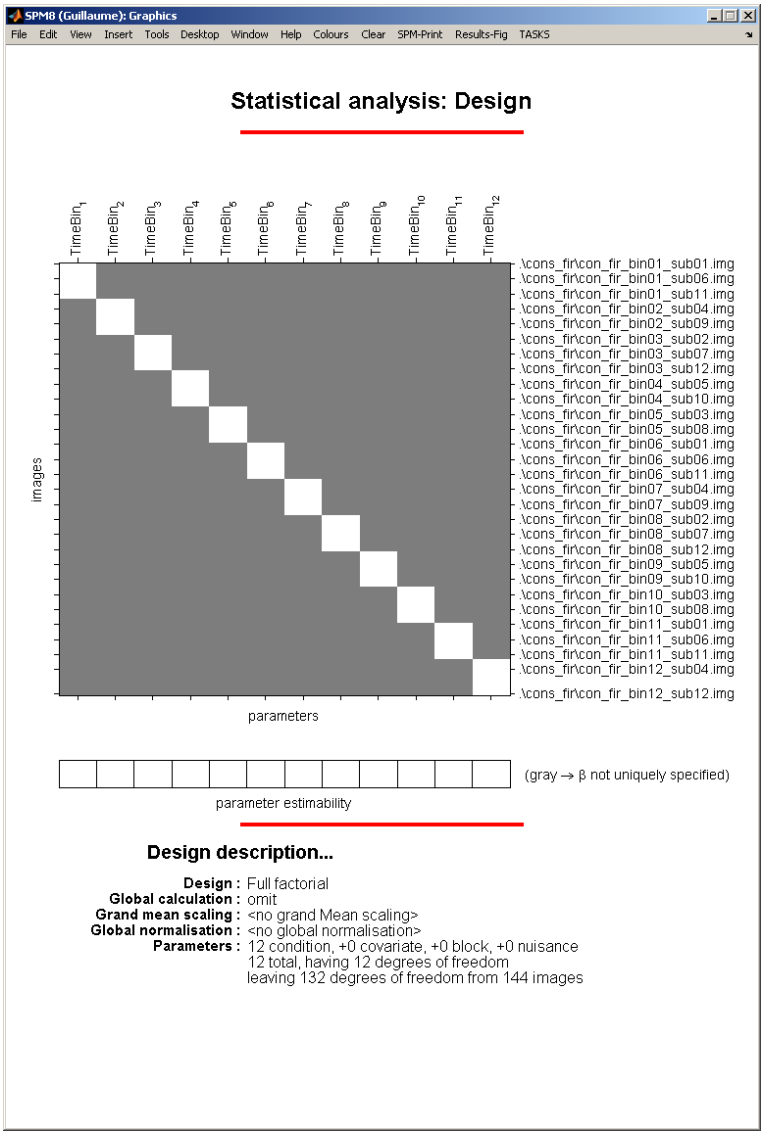


Figure 30.10: *Design matrix for FIR basis set. This corresponds to a one-way ANOVA with 12 levels.*

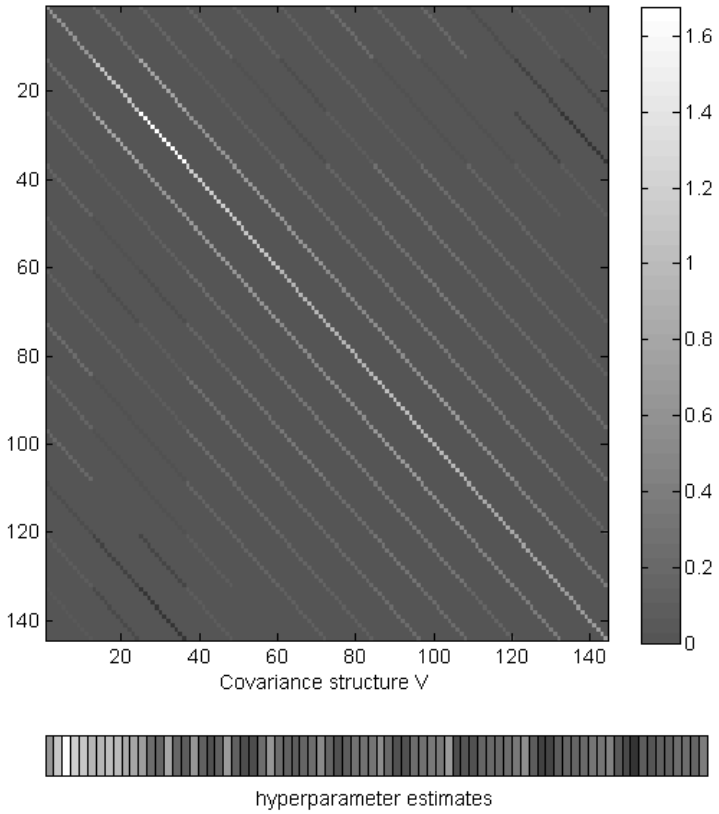


Figure 30.11: *Estimated covariance matrix for FIR basis set. The differently valued hyperparameters are shown in different shades of gray. Notice that the most variable responses occur in the third time bin (scans 25 to 36) corresponding to responses 4-6 seconds post stimulus, ie. at the peak of the hemodynamic response, as expected.*

30.5.2 FIR Results

- Now press the "Results" button.
- Select the `SPM.mat` file.
- In the contrast manager press "Define new contrast" (select F). Enter `eye(12)` in the contrast section and enter "Faces vs Baseline: FIR" as a "name'.[4]
- Press the "..submit" button. Press OK.
- Now press the "Done" button.
- Mask with other contrast(s) [No]
- Title for comparison: accept [Faces vs Baseline: FIR]
- p value adjustment to control [FWE]
- Family-wise p-value [0.05]
- Extent threshold voxels [0]

Note how the design matrix, shown in Figure 30.12 appears to be different after estimation. This is because it has been pre-whitened. In particular, the off-diagonal entries in the design matrix give an indication of the degree of correlation between the time bins across subjects (this is displayed explicitly in the covariance matrix in Figure 30.11).

The above contrast will reveal voxels that show *any* form of event-related response, within the range 0-24s post-stimulus and with 2s resolution, as shown in Figure 30.12. Selecting a voxel and plotting this contrast (using the *plot* button) will reveal that most voxels have a fairly "canonical" shape over the 12 timebins. One can also test for more constrained shapes of event-related responses within this model. For example, one can test for "canonical-shaped" responses by evaluating a contrast whose weights trace out SPM's canonical HRF (every 2s). To do this, switch to the MATLAB window for a moment and type:

- `xBF.dt = 1`
- `xBF.name = 'hrf (with time and dispersion derivatives)';`
- `xBF.length = 32;`
- `xBF.order = 1;`
- `xBF = spm_get_bf(xBF);`

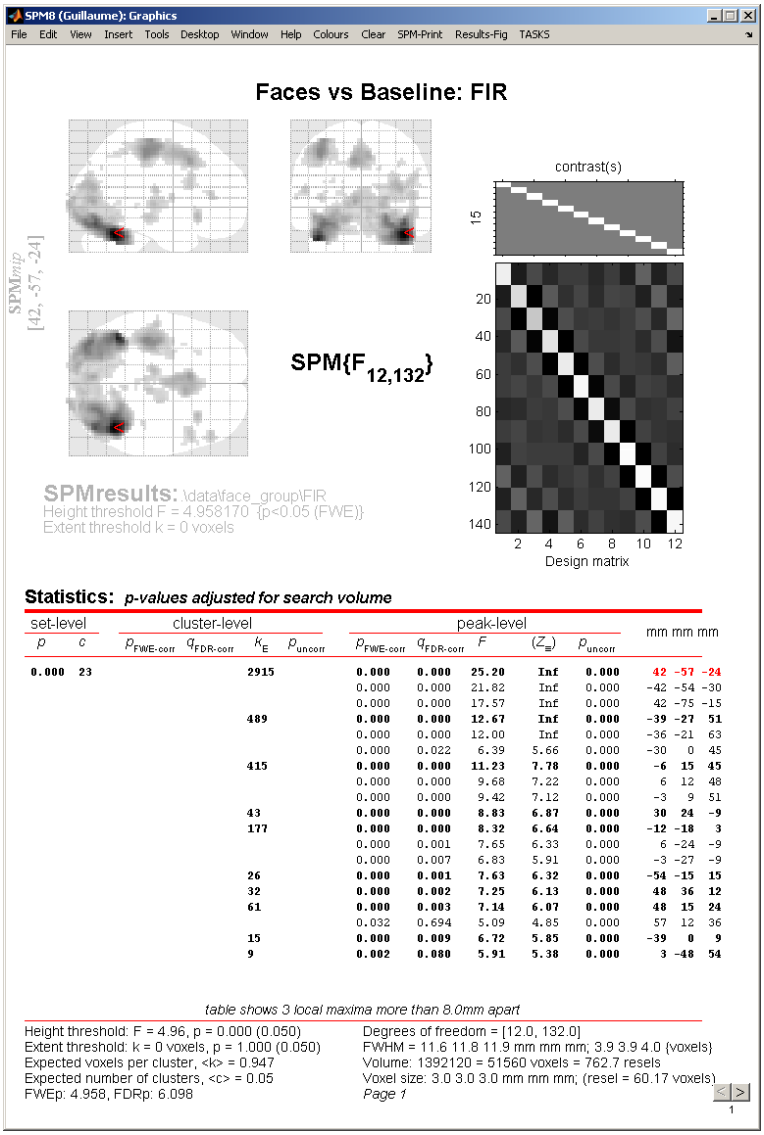
This returns the canonical and two derivatives in the matrix `xBF.bf` (type `help spm_get_bf` for more info), with one value every 1 second. For convenience, then define:

- `all  = xBF.bf(2:2:24,:)';`
- `can  = all(1,:);`
- `tem  = all(2,:);`
- `dis  = all(3,:);`

These commands downsample the basis functions every 2s, which is the bin-width of the FIR. If you type `corrcoef(all')`, you will see that the basis functions are slightly correlated (in the off-diagonal terms), due to this undersampling every 2s.

- In the contrast manager press "Define new contrast" (select T).

[4] SPM will have produced some contrasts automatically, one of them being the "main effect of TimeBin". This contrast is, however, not appropriate for our purposes.



Statistics: *p-values adjusted for search volume*

set-level		cluster-level				peak-level					mm mm mm
p	c	$p_{FWE\text{-}corr}$	$q_{FDR\text{-}corr}$	k_E	p_{uncorr}	$p_{FWE\text{-}corr}$	$q_{FDR\text{-}corr}$	F	$(Z_{\equiv})$	p_{uncorr}	
0.000	**23**			**2915**		**0.000**	**0.000**	**25.20**	**Inf**	**0.000**	**42 -57 -24**
						0.000	0.000	21.82	Inf	0.000	-42 -54 -30
						0.000	0.000	17.57	Inf	0.000	42 -75 -15
				489		**0.000**	**0.000**	**12.67**	**Inf**	**0.000**	**-39 -27 51**
						0.000	0.000	12.00	Inf	0.000	-36 -21 63
						0.000	0.022	6.39	5.66	0.000	-30 0 45
				415		**0.000**	**0.000**	**11.23**	**7.78**	**0.000**	**-6 15 45**
						0.000	0.000	9.68	7.22	0.000	6 12 48
						0.000	0.000	9.42	7.12	0.000	-3 9 51
				43		**0.000**	**0.000**	**8.83**	**6.87**	**0.000**	**30 24 -9**
				177		**0.000**	**0.000**	**8.32**	**6.64**	**0.000**	**-12 -18 3**
						0.000	0.001	7.65	6.33	0.000	6 -24 -9
						0.000	0.007	6.83	5.91	0.000	-3 -27 -9
				26		**0.000**	**0.001**	**7.63**	**6.32**	**0.000**	**-54 -15 15**
				32		**0.000**	**0.002**	**7.25**	**6.13**	**0.000**	**48 36 12**
				61		**0.000**	**0.003**	**7.14**	**6.07**	**0.000**	**48 15 24**
						0.032	0.694	5.09	4.85	0.000	57 12 36
				15		**0.000**	**0.009**	**6.72**	**5.85**	**0.000**	**-39 0 9**
				9		**0.002**	**0.080**	**5.91**	**5.38**	**0.000**	**3 -48 54**

table shows 3 local maxima more than 8.0mm apart

Height threshold: F = 4.96, p = 0.000 (0.050)
Extent threshold: k = 0 voxels, p = 1.000 (0.050)
Expected voxels per cluster, <k> = 0.947
Expected number of clusters, <c> = 0.05
FWEp: 4.958, FDRp: 6.098

Degrees of freedom = [12.0, 132.0]
FWHM = 11.6 11.8 11.9 mm mm mm; 3.9 3.9 4.0 {voxels}
Volume: 1392120 = 51560 voxels = 762.7 resels
Voxel size: 3.0 3.0 3.0 mm mm mm; (resel = 60.17 voxels)
Page 1

Figure 30.12: *Main population effect of faces, as characterised with the FIR basis set.*

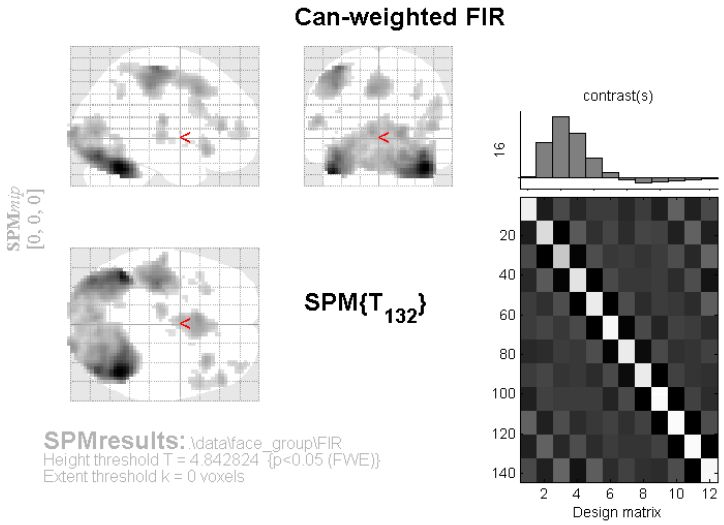


Figure 30.13: *Main population effect of faces, as characterised with a canonically weighted contrast of FIR bases.*

- Enter `can` as the contrast weights (defined in MATLAB workspace as above), and "Can-weighted FIR" as the name.

This produces the MIP in Figure 30.13. At a FWE correct p value of 0.05, there are many more voxels compared to the equivalent T-contrast [1] in the model using only canonical HRF. The main reason for this increased power is again the increase in the degrees of freedom, which entails better estimators of the underlying error (co)variance (though if the FIR parameters were estimated very inefficiently, the extra contrast images might add more noise, outweighing any advantage of higher degrees of freedom). Again, this increased power comes with a stronger assumption about the nonsphericity, namely that it has the same structure across (activated) voxels [36]. One can also test the variance captured by the temporal and dispersion derivatives by creating new contrasts (though as F rather than T contrasts) and simply typing "tem" and "dis" respectively as the contrast weights.

More interesting is the ability to ask, within this model, how much event-related variance is *not* captured by the canonical HRF. To do this, first create the variable in MATLAB

- `nullcan = eye(12) - pinv(can)*can;`

This creates a matrix for an F-contrast that spans the "null space" of the canonical HRF.

- In the contrast manager press "Define new contrast" (select F).
- Enter `nullcan` as the contrast weights (defined in MATLAB workspace as above), and "Null space of canonical HRF" as the name.

You can see, in Figure 30.14 that several regions express variability not captured by the canonical HRF. This is not surprising, because you will notice that many of these regions appeared in the individual F-tests on the temporal and dispersion derivatives above, suggesting that what is not captured by the canonical HRF is captured by its two derivatives.

Yet even more interesting is the ability to ask how much event-related variance is *not* captured by the canonical HRF or its two derivatives (ie. not captured by SPM's 'informed' basis set). To do this, first create the variable in MATLAB

- `nullall = eye(12) - pinv(all)*all;`

This creates a matrix for an F-contrast that spans the "null space" of all three informed basis functions.

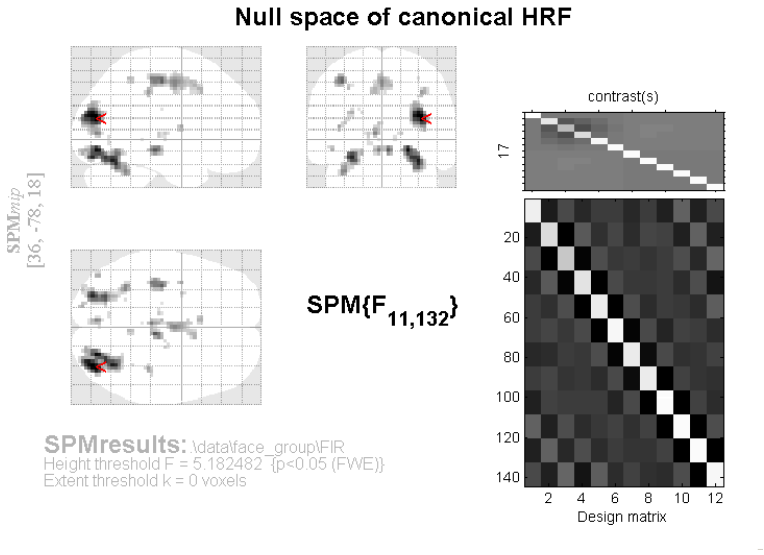


Figure 30.14: *Regions expressing variability across subjects not captured by canonical HRF.*

- In the contrast manager press "Define new contrast" (select F).
- Enter `nullall` as the contrast weights (defined in MATLAB workspace as above), and "Null space of informed basis set" as the name.

You will see, in Figure 30.15 that only 2 voxels (in one cluster with maximum -21 -18 27) express variability not captured by the informed basis set. This reinforces the point that, while there is certainly variability in the HRF across different brain regions, the canonical HRF and its two derivatives are sufficient to capture the majority of this regional variability (at least on average across the 12 subjects in this dataset). See [40] for further details.

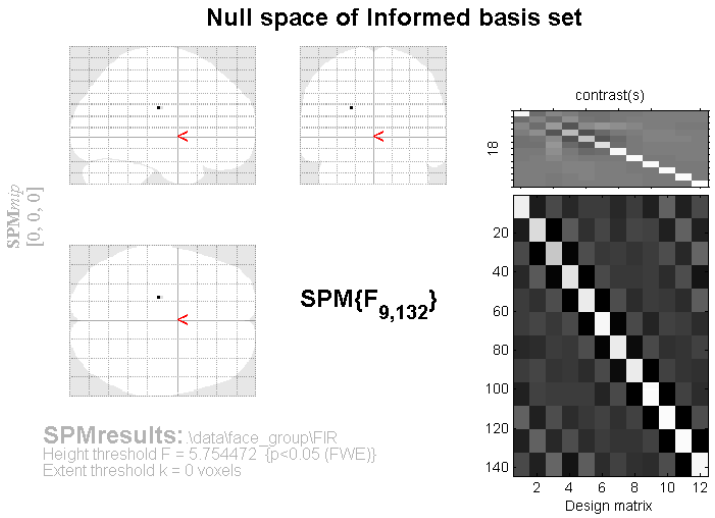


Figure 30.15: *Regions expressing variability across subjects not captured by informed basis set.*

Chapter 31

Verbal Fluency PET data

31.1 Introduction

These data come from a 5 subject PET study of a verbal fluency with two alternating word generation conditions: A (baseline) - word shadowing; B - (activation) - paced orthographic word generation. This involved responding with a word beginning with an aurally presented letter. Both conditions were identically paced at 1 word every 2 seconds. The presentation order alternated between AB and BA across subjects as shown in Table 31.1.

Scan:	1	2	3	4	5	6	7	8	9	10	11	12
Subject 1	A	B	A	B	A	B	A	B	A	B	A	B
Subject 2	B	A	B	A	B	A	B	A	B	A	B	A
Subject 3	A	B	A	B	A	B	A	B	A	B	A	B
Subject 4	B	A	B	A	B	A	B	A	B	A	B	A
Subject 5	A	B	A	B	A	B	A	B	A	B	A	B

Table 31.1: ***Conditions for PET data:*** *(A) word shadowing and (B) word generation.*

The files are named `./p#/snrp#_##.{img,hdr}` and are SPM compatible (Analyze) images following realignment, normalization and smoothing with a 16mm isotropic Gaussian kernel with `#` indicating the subject and `##` the scan. The data set is available from the SPM website[1].

To analyse the data, first create a new directory `DIR`, eg. `c:\data\pet`, in which to place the results of your analysis. Then create 4 subdirectories (i) `single`, (ii) `subject-condition`, (iii) `subject-time` and (iv) `multiple`. As the analysis proceeds these directories will be filled with job-specification files, design matrices and estimated models.

31.2 Single subject

Firstly, we will analyse the data from a single subject. This can be implemented as follows.

- Start up MATLAB and type `spm pet` at the prompt
- Press the "Basic models" button.
- In 'Factorial design specification', choose the 'Flexible Factorial' design.
- Highlight 'Factors' and create a new Factor and enter 'Word' for the name.
- Then, under 'Specify Subject or all Scans and Factors', highlight 'Subjects' and create a new subject.

[1] Verbal Fluency PET dataset: http://www.fil.ion.ucl.ac.uk/spm/data/fluency/

- Highlight 'Scans', select 'Specify Files' and use the SPM file selector to choose the 12 images for that subject. This can be most easily achieved by specifying '.*snrp1.*' as a filter in the file selector.
- Under 'Conditions' enter the vector [1 2 1 2 1 2 1 2 1 2 1 2].
- Under 'Main effects and interactions', create a single main effect with factor number equal to 1
- Under 'Covariates', create a new covariate and enter 'Time' for 'Name' and the vector '1:12'.
- Under 'Global calculation' choose 'Mean'
- Under Global normalisation and Normalisation, choose 'Proportional' scaling.[2]
- Under Global normalisation and Overall grand mean scaling, select YES.
- Highlight Directory, Specify files and select the subdirectory 'single', to place the design matrix in.
- Save the job file as eg. `DIR/single_design.mat`.
- Press the `Run` button (green arrow).

SPM will then show you the design matrix shown in Figure 31.1. This design is encoded in the `SPM.mat` file that is written to the output directory. Then press 'Estimate' and select the `SPM.mat` file just created. SPM will now estimate the parameters, that is, the size of the population effect at each voxel.

- Now press the 'Results' button.
- Select the `SPM.mat` file.
- In the contrast manager press 'Define new contrast' (select T). Enter [-1 1] in the contrast section and enter 'activation' as a 'name'.
- Press the '..submit' button. Press OK.
- Now press the 'Done' button.
- Mask with other contrast(s) [No]
- Title for comparison [activation]
- p value adjustment to control [FWE]
- Family-wise p-value [0.05]
- Extent threshold voxels [0]

You should see a blank MIP as, sadly, we rarely have enough sensitivity to find activations in single subject PET data. This is why we scan multiple subjects.

31.3 Multiple subjects

The data set can be analysed in several ways which are discussed in [46].

[2] Normalisation using ANCOVA is advised for multi-subject studies unless differences in global flow are large eg. due to variability in injected tracer dose. Because ANCOVA uses one degree of freedom for each subject/group, proportional scaling may be preferable for single-subject studies.

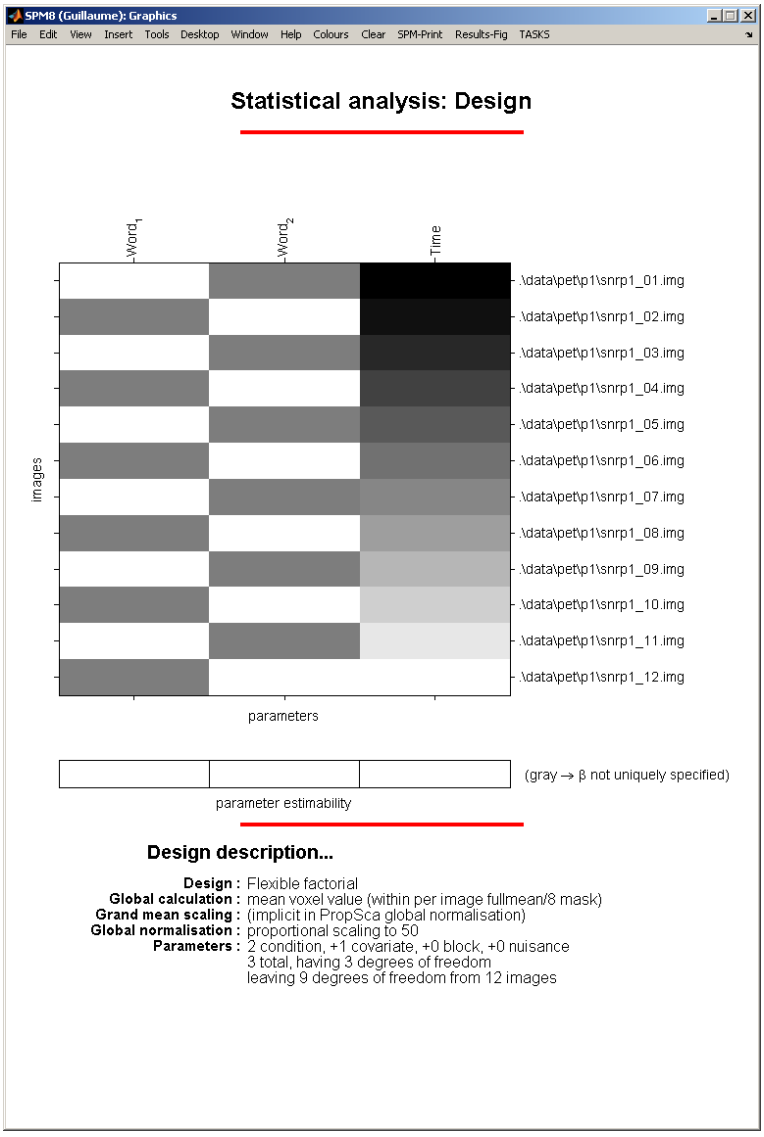


Figure 31.1: *Design matrix for single-subject data. The first two columns model responses to word shadowing and word generation. The third column models time-varying responses.*

31.3.1 Subject and Condition design

First we set up a design that allows us to test for the main effects of 'Subject' and 'Condition'. The design can be set-up as follows.

- Start up MATLAB and type `spm pet` at the prompt
- Press the 'Basic Models' button.
- In 'Factorial design specification', under 'Design', choose the 'Flexible Factorial' design.
- Highlight 'Factors' and create a new Factor.
- Enter 'Subject' for the name and select 'Equal' under 'Variance'.
- Then create another factor and call it 'Word'
- Then, under 'Specify Subject or all Scans and Factors', highlight 'Subjects' and create a 5 new subjects.
- For the first subject, highlight 'Scans', select 'Specify Files' and use the SPM file selector to choose the 12 images for the first subject. This can be most easily achieved by specifying `.*snrp1.*` as a filter in the file selector and then using a right click to 'select all'.
- Under 'Conditions' enter the vector [1 2 1 2 1 2 1 2 1 2 1 2].
- Repeat the specification of scans and conditions for each of the four other subjects, remembering that the order of conditions has been balanced across the group (see Table 31.1).
- Under 'Main effects and interactions', create two main effects, the first with factor number 1 (ie. Subject) and the second with factor number 2 (ie. Word).
- Under Masking, select 'Relative' for 'Threshold Masking' and accept the default value of 0.8. Voxels with mean value less than 0.8 of the mean are deemed extra-cranial and will be excluded from the analysis.
- Under 'Global calculation' choose 'Mean'
- Under, Global normalisation, and Normalisation, select 'ANCOVA'.
- Highlight Directory, Specify files and select the subdirectory 'subject-condition', to place the design matrix in.
- Save the job file as eg. `DIR/sc_design.mat`.
- Press the `Run` button.

SPM will then show you the design matrix shown in Figure 31.2. This design is encoded in the `SPM.mat` file that is written to the output directory.

31.3.2 Subject and Time design

We now set up a design that allows us to test for the effects of Time (ie. scan number) and Subject. If you have already specified the Subject and Conditions design, then you can set up the Subject and Time design by editing the `sc_design.mat` file (and just changing the name of the second factor, Conditions vector and output directory - see below). Otherwise, the design can be set-up as follows.

- Start up MATLAB and type `spm pet` at the prompt
- Press the 'Basic Models' button.
- In 'Factorial design specification', under 'Design', choose the 'Flexible Factorial' design.
- Highlight 'Factors' and create a new Factor.

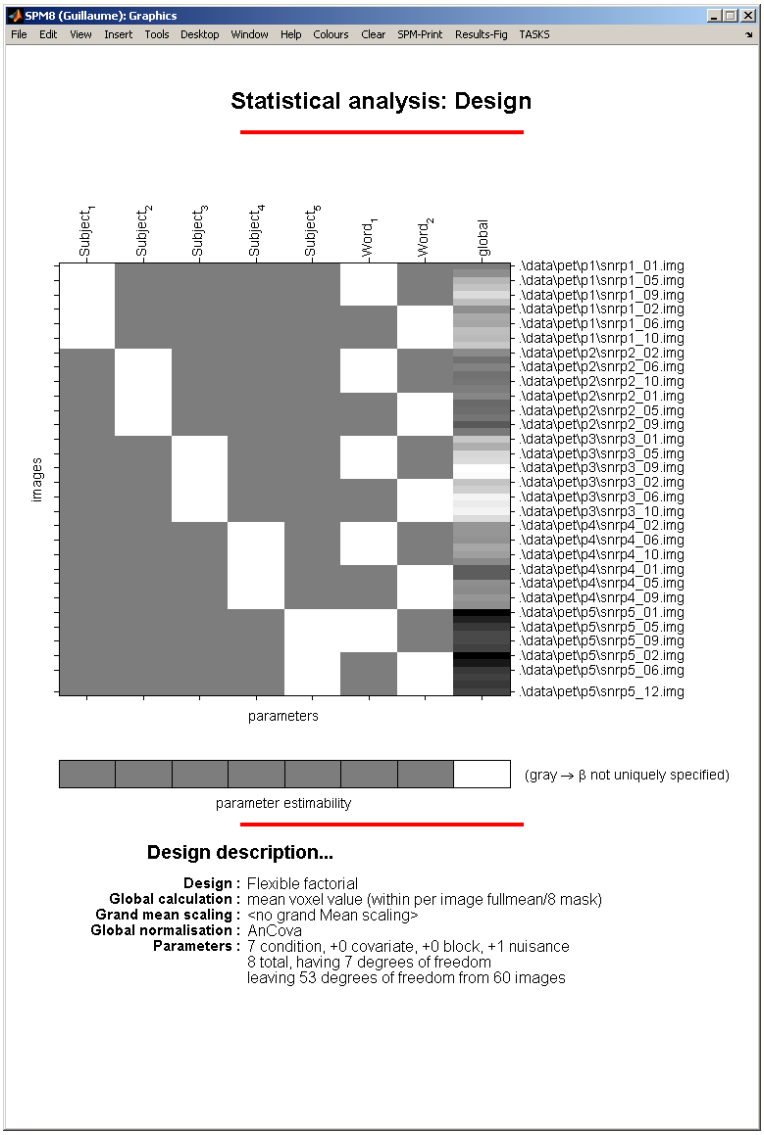


Figure 31.2: *Subjects and Conditions design for multiple-subject data. The first five columns model effect and the next two columns model condition effects. The last colum models global effects (ANCOVA).*

- Enter 'Subject' for the name and select 'Equal' under 'Variance'.
- Then create another factor and call it 'Time'. This factor extends over time for each subject.
- Then, under 'Specify Subject or all Scans and Factors', highlight 'Subjects' and create a 5 new subjects.
- For the first subject, highlight 'Scans', select 'Specify Files' and use the SPM file selector to choose the 12 images for the first subject. This can be most easily achieved by specifying `.*snrp1.*` as a filter in the file selector and then using a right click to 'select all'.
- Under 'Conditions' enter the vector [1:12].
- Repeat the specification of scans and conditions for each of the four other subjects.
- Under 'Main effects and interactions', create two main effects, the first with factor number 1 (ie. Subject) and the second with factor number 2 (ie. Time).
- Under Masking, select 'Relative' for 'Threshold Masking' and accept the default value of 0.8. Voxels with mean value less than 0.8 of the mean are deemed extra-cranial and will be excluded from the analysis.
- Under 'Global calculation' choose 'Mean'
- Under, Global normalisation, and Normalisation, select 'ANCOVA'.
- Highlight Directory, Specify files and select the subdirectory 'subject-condition', to place the design matrix in.
- Save the job file as eg. `DIR/st_design.mat`.
- Press the `Run` button.

SPM will then show you the design matrix shown in Figure 31.3. This design is encoded in the `SPM.mat` file that is written to the output directory.

31.3.3 Subject by Condition design

This design models the interacts between 'Subject' and 'Condition'. It allows effects to be assessed separately for each subject. It will also allow us to implement a conjunction analysis over subjects.

If you have already specified the Subject and Conditions or Subject and Time designs then this design can be more easily specified by editing the `sc_design.mat` or `st_design.mat` files (and changing the name of the second factor, removing main effects, adding the interaction term and specifying a new output directory - see below). Otherwise, the design can be set-up as follows.

- Start up MATLABand type `spm pet` at the prompt
- Press the "Basic Models" button.
- In 'Factorial design specification', under 'Design', choose the 'Flexible Factorial' design.
- Highlight 'Factors' and create a new Factor.
- Enter 'Subject' for the name and select 'Yes' under ANCOVA, as we will be implementing ANCOVA-by-subject. Select 'Equal' under 'Variance'.
- Then create another factor and call it 'Word'
- Then, under 'Specify Subject or all Scans and Factors', highlight 'Subjects' and create a 5 new subjects.
- For the first subject, highlight 'Scans', select 'Specify Files' and use the SPM file selector to choose the 12 images for the first subject. This can be most easily achieved by specifying '.*snrp1.*' as a filter in the file selector and then using a right click to 'select all'.

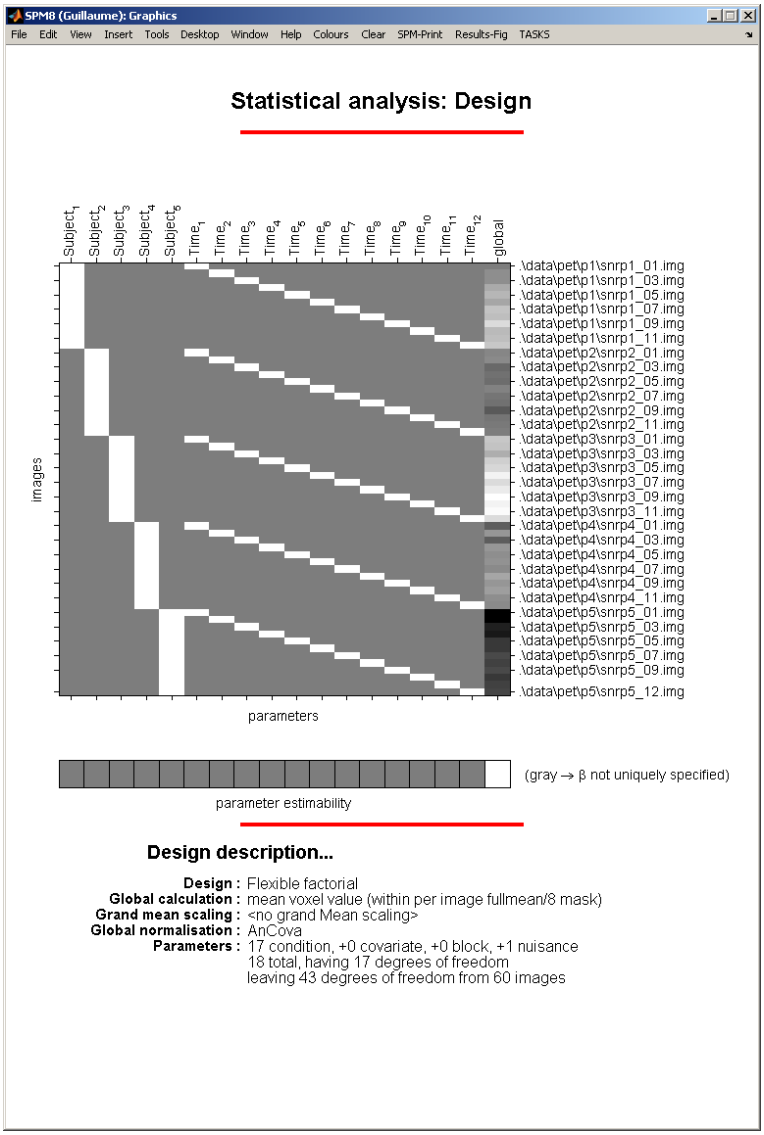


Figure 31.3: *Subjects and Time design for multiple-subject data. The first five columns model subjects effects and the next 12 model time effects. The last column models global effects (ANCOVA).*

- Under 'Conditions' enter the vector [1 2 1 2 1 2 1 2 1 2 1 2].
- Repeat the specification of scans and conditions for each of the four other subjects, remembering that the order of conditions has been balanced across the group (see Table 31.1).
- Under 'Main effects and interactions', create an interaction with factor numbers equal to [1 2]. This will create a block in the design matrix that models interactions between the factors 'Subject' and 'Word'.
- Under Masking, select 'Relative' for 'Threshold Masking' and accept the default value of 0.8. Voxels with mean value less than 0.8 of the mean are deemed extra-cranial and will be excluded from the analysis.
- Under 'Global calculation' choose 'Mean'
- Highlight Directory, Specify files and select the subdirectory `multiple`, to place the design matrix in.
- Save the job file as eg. `DIR/multi_design.mat`.
- Press the `Run` button.

SPM will then show you the design matrix shown in Figure 31.4. This design is encoded in the 'SPM.mat' file that is written to the output directory. Then press 'Estimate' and select the `SPM.mat` file just created. SPM will now estimate the parameters, that is, the size of the effect at each voxel. The rest of this chapter pursues the 'Subject-by-Condition' design.

31.3.4 Contrast manager

We can then examine relative activations, that is, regions which respond more strongly during word generation than word shadowing, for each subject. For subject 2:

- Press the 'Results' button.
- Select the `SPM.mat` file.
- In the contrast manager press 'Define new contrast' (select T)
- Specify e.g. `Subject 2: Gen > Shad` (name) and '0 0 -1 1' (contrast).
- Press the '..submit' button. Press OK.
- Now press the 'Done' button.
- Mask with other contrast(s) [No]
- Title for comparison [activation]
- p value adjustment to control [FWE]
- Family-wise p-value [0.05]
- Extent threshold voxels [0]

This should produce the contrast in Figure 31.5. As shown, SPM will automatically pad '0 0 -1 1' with zeros at the end. To examine group effects:

- Press the 'Results' button.
- Select the SPM.mat file.
- In the contrast manager press 'Define new contrast' (select T)
- Specify e.g. `All: Gen > Shad` (name) and '-1 1 -1 1 -1 1 -1 1 -1 1' and select it (press 'Done') (contrast).

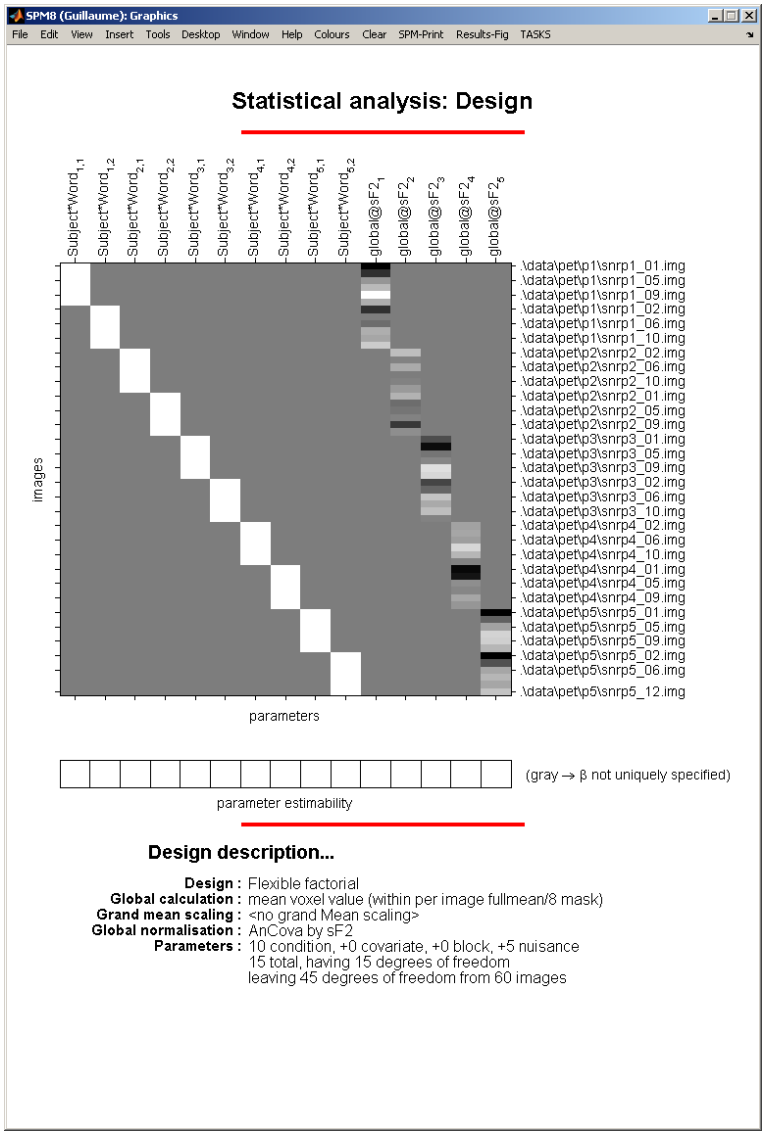


Figure 31.4: *Subject by Condition design for multiple-subject data. The first ten columns model interactions between 'Subject' and 'Word'. The last five columns model out global effects for each subject. Inclusion of these last five regressors implements a so-called 'ANCOVA-by-subject' normalisation.*

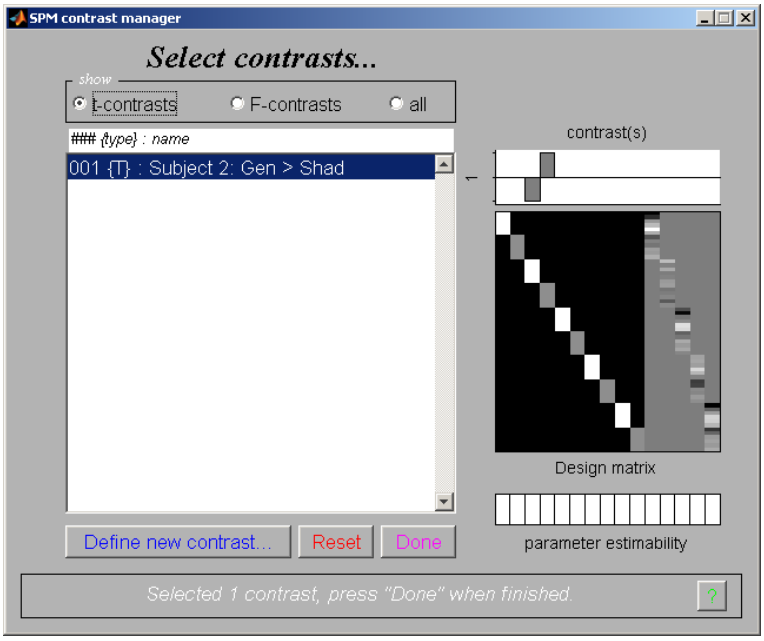


Figure 31.5: *Activation contrast for subject 2. Note that the block of the design matrix encoding the experimental conditions is now coloured differently. This is because we have allowed the variance of responses over subjects to be different between word shadowing and generation conditions. This 'nonsphericity' affects parameter estimation in a way that is implemented in SPM by first 'colouring' the design matrix and then implementing ordinary least squares. This, in no way however, affects interpretation of effects.*

- Mask with other contrast(s) [No]
- Title for comparison [activation]
- p value adjustment to control [FWE]
- Family-wise p-value [0.05]
- Extent threshold voxels [0]

Before looking at the results we describe the masking and thresholding options in more detail.

31.3.5 Masking and thresholds

Masking implies selecting voxels specified by other contrasts. If 'yes', SPM will prompt for (one or more) masking contrasts, the significance level of the mask (default p = 0.05 uncorrected), and will ask whether an inclusive or exclusive mask should be used. Exclusive will remove all voxels which reach the default level of significance in the masking contrast, inclusive will remove all voxels which do not reach the default level of significance in the masking contrast. Masking does not affect p-values of the 'target' contrast.

Selecting a height threshold for examine results uses either a threshold corrected for multiple comparisons ('yes'), or uncorrected ('no'). The latter will produce many false positives (FPs) in the SPM. On average, the number of false positives will be equal to the number of voxels in the volume times the p-value (eg. $50,000 \times 0.001 = 50$). If you correct for multiple comparisons, however, then there will typically be only one FP *anywhere* in 20 SPMs. Correcting for multiple comparisons is the recommended option.

Specifying an extent threshold x tells SPM not to plot clusters containing fewer than x voxels. The default, $x = 0$ allows single voxel activations to be displayed.

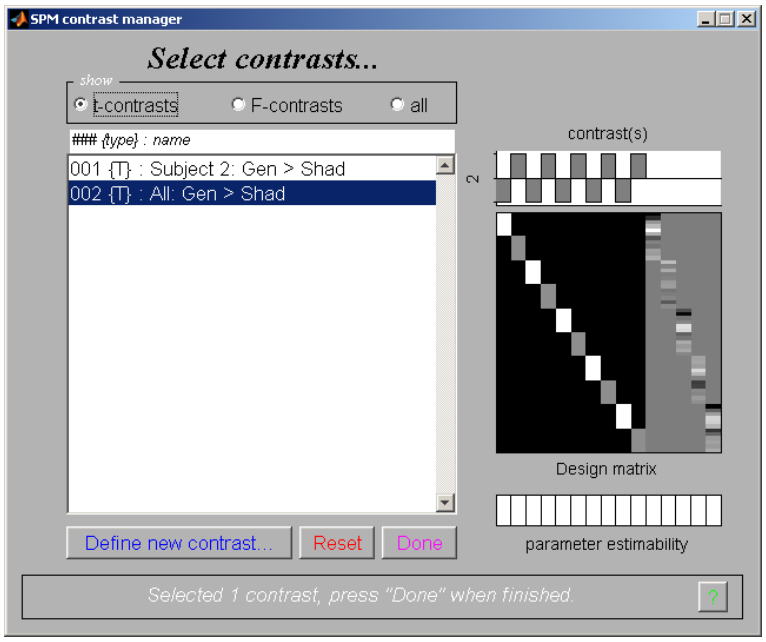


Figure 31.6: *Activation contrast for all subjects.*

31.3.6 MIPs and results tables

The above contrast specifications should configure the contrast manager to appear as in Figure 31.6 and will configure SPM's graphics window to look like Figure 31.7.

SPM will also produce a number of files: images containing weighted parameter estimates are saved as `con_0002.hdr/img`, `con_0003.hdr/img`, etc. in the output directory. Images of T-statistics are saved as `spmT_0002.hdr/img`, `spmT_0003.hdr/img`, etc., also in the output directory. A number of further options are available from SPM Interactive window shown in Figure 31.8.

In the SPM Interactive window (lower left panel) a button box appears with various options for displaying statistical results (p-values panel) and creating plots/overlays (visualisation panel). Clicking 'Design' (upper left) will activate a pulldown menu as in the 'Explore design' option. To get a summary of local maxima, press 'volume'. This will produce the table shown in Figure 31.9. As in the previous example, this will list all clusters above the chosen level of significance as well as separate (>8mm apart) maxima within a cluster, with details of significance thresholds and search volume underneath. The columns show, from right to left:

- **x, y, z (mm):** coordinates in Talairach space for each maximum.
- **peak-level:** the chance (p) of finding (under the null hypothesis) a peak with this or a greater height (T- or Z-statistic), corrected / uncorrected for search volume.
- **cluster-level:** the chance (p) of finding a cluster with this or a greater size (ke), corrected / uncorrected for search volume.
- **set-level:** the chance (p) of finding this or a greater number of clusters (c) in the search volume.

Its also worth noting that

- The table is surfable: clicking a row of cluster coordinates will move the pointer in the MIP to that cluster, clicking other numbers will display the exact value in the Matlab window (e.g. 0.000 = 6.1971e-07).
- To inspect a specific cluster, either move the cursor in the MIP (by L-clicking & dragging the cursor, or R-clicking the MIP background which will activate a pulldown menu).

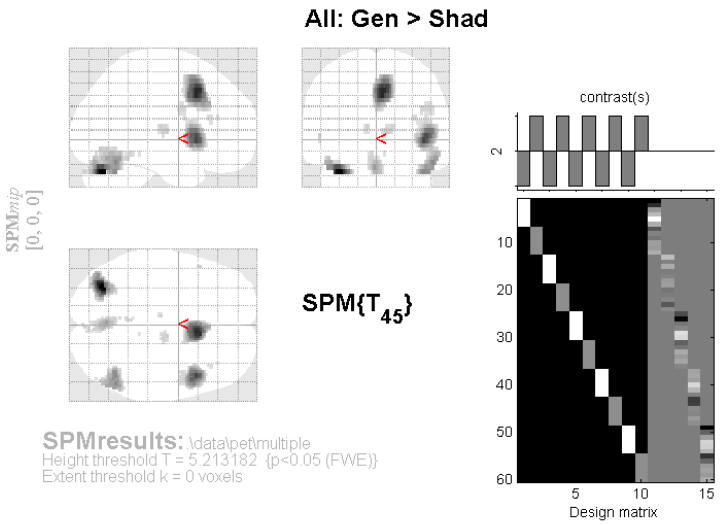


Figure 31.7: *SPMs graphics window displays (Left) a maximum intensity projection (MIP) on a glass brain in three orthogonal planes. The MIP is surfable: right-clicking in the MIP will activate a pulldown menu, left-clicking on the red cursor will allow it to be dragged to a new position, (Right) the design matrix (showing the selected contrast). The design matrix is also surfable: right-clicking will show parameter names, left-clicking will show design matrix values for each scan.*

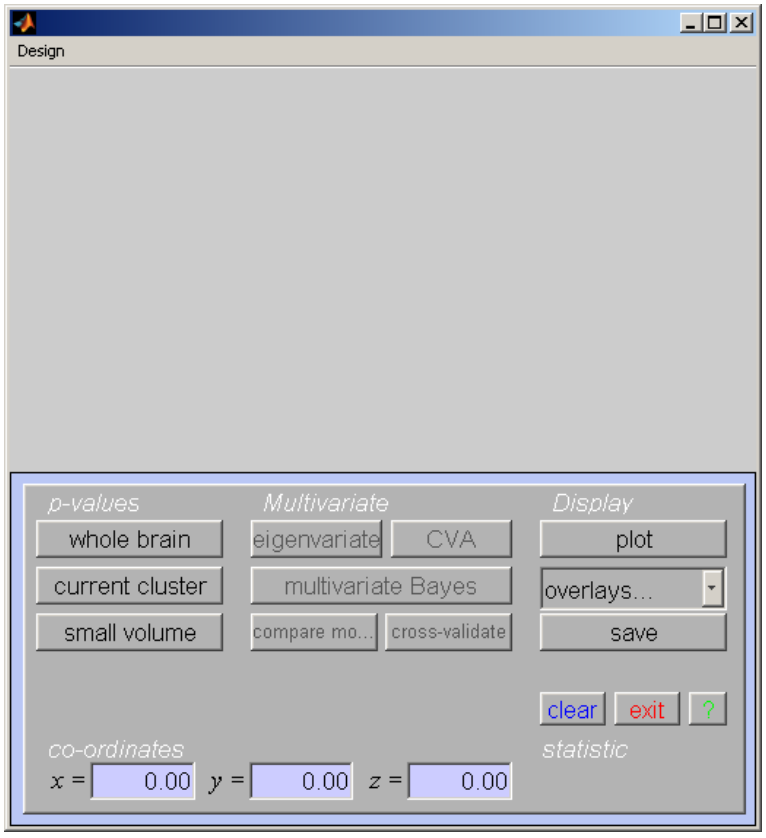
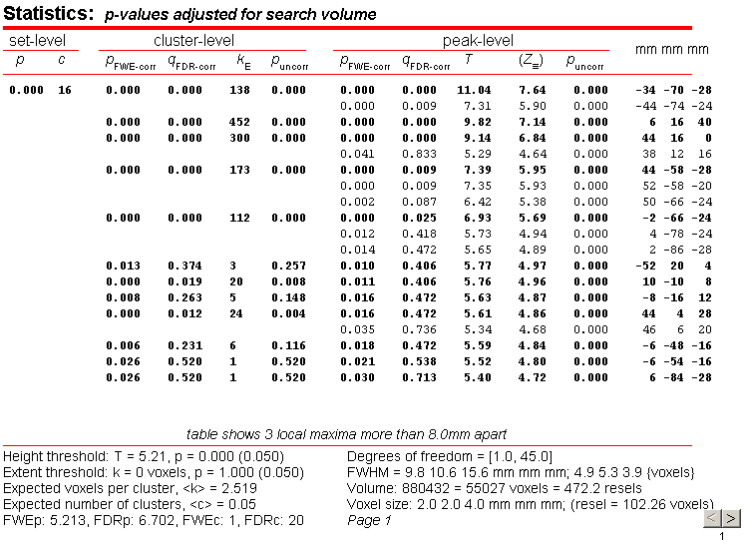


Figure 31.8: *SPM's interactive window.*



Statistics: *p-values adjusted for search volume*

set-level		cluster-level				peak-level					mm mm mm	
p	c	$p_{FWE\text{-}corr}$	$q_{FDR\text{-}corr}$	k_E	p_{uncorr}	$p_{FWE\text{-}corr}$	$q_{FDR\text{-}corr}$	T	$(Z_{\equiv})$	p_{uncorr}		
0.000	**16**	**0.000**	**0.000**	**138**	**0.000**	**0.000**	**0.000**	**11.04**	**7.64**	**0.000**	**-34**	**-70** **-28**
						0.000	0.009	7.31	5.90	0.000	-44	-74 -24
		0.000	**0.000**	**452**	**0.000**	**0.000**	**0.000**	**9.82**	**7.14**	**0.000**	**6**	**16** **40**
		0.000	**0.000**	**300**	**0.000**	**0.000**	**0.000**	**9.14**	**6.84**	**0.000**	**44**	**16** **0**
						0.041	0.833	5.29	4.64	0.000	38	12 16
		0.000	**0.000**	**173**	**0.000**	**0.000**	**0.009**	**7.39**	**5.95**	**0.000**	**44**	**-58** **-28**
						0.000	0.009	7.35	5.93	0.000	52	-58 -20
						0.002	0.087	6.42	5.38	0.000	50	-66 -24
		0.000	**0.000**	**112**	**0.000**	**0.000**	**0.025**	**6.93**	**5.69**	**0.000**	**-2**	**-66** **-24**
						0.012	0.418	5.73	4.94	0.000	4	-78 -24
						0.014	0.472	5.65	4.89	0.000	2	-86 -28
		0.013	**0.374**	**3**	**0.257**	**0.010**	**0.406**	**5.77**	**4.97**	**0.000**	**-52**	**20** **4**
		0.000	**0.019**	**20**	**0.008**	**0.011**	**0.406**	**5.76**	**4.96**	**0.000**	**10**	**-10** **8**
		0.008	**0.263**	**5**	**0.148**	**0.016**	**0.472**	**5.63**	**4.87**	**0.000**	**-8**	**-16** **12**
		0.000	**0.012**	**24**	**0.004**	**0.016**	**0.472**	**5.61**	**4.86**	**0.000**	**44**	**4** **28**
						0.035	0.736	5.34	4.68	0.000	46	6 20
		0.006	**0.231**	**6**	**0.116**	**0.018**	**0.472**	**5.59**	**4.84**	**0.000**	**-6**	**-48** **-16**
		0.026	**0.520**	**1**	**0.520**	**0.021**	**0.538**	**5.52**	**4.80**	**0.000**	**-6**	**-54** **-16**
		0.026	**0.520**	**1**	**0.520**	**0.030**	**0.713**	**5.40**	**4.72**	**0.000**	**6**	**-84** **-28**

table shows 3 local maxima more than 8.0mm apart

Height threshold: T = 5.21, p = 0.000 (0.050)
Extent threshold: k = 0 voxels, p = 1.000 (0.050)
Expected voxels per cluster, <k> = 2.519
Expected number of clusters, <c> = 0.05
FWEp: 5.213, FDRp: 6.702, FWEc: 1, FDRc: 20

Degrees of freedom = [1.0, 45.0]
FWHM = 9.8 10.6 15.6 mm mm mm; 4.9 5.3 3.9 {voxels}
Volume: 880432 = 55027 voxels = 472.2 resels
Voxel size: 2.0 2.0 4.0 mm mm mm; (resel = 102.26 voxels)
Page 1 < >
1

Figure 31.9: *SPM results table. This appears below the MIP, shown in Figure 31.7, in the graphics window.*

- Alternatively, click the cluster coordinates in the volume table, or type the coordinates in the lower left windows of the SPM Interactive window.

Selecting 'cluster' will show coordinates and voxel-level statistics for local maxima (>4mm apart) in the selected cluster. See Figure 31.10. The table is also surfable. Both in the 'volume' and 'cluster' options, p-values are corrected for the entire search volume.

31.3.7 Small volume correction

If one has an a priori anatomical hypothesis, eg. in the present example Broca's area will likely be activated during word generation, one may use the small volume correction option. Press the "small volume" button in SPM Interactive (bottom left) window and select a suitable region, e.g., a 30mm sphere with its centre at 44 16 0. The region can also be defined using mask images derived from previous imaging data. The corrected p-values will change, as shown in Figure 31.11.

31.3.8 Extracting data from regions

To extract a time course for data in this region of interest (this uses the SPM function `spm_regions.m`):

- Select "eigenvariate" from the "Multivariate" section in the Interactive window
- Select ('don't adjust')
- Specify 'Broca' for name of region and 0 for the VOI radius.

SPM displays a graph of the first eigenvariate of the data in or centered around the chosen voxel, as shown in Figure 31.12. It also lists the eigenvariate values Y in the Matlab window. Adjustment is with respect to the null space of a selected contrast. This means that any effects not spanned by the chosen contrast are removed from the data, before extraction. Adjustment can be omitted by selecting 'don't adjust', as above.

SPM extracts the eigenvariate values in a region, rather than the mean values, as the former is more robust to heterogeneity of response within a cluster. The mean value can be thought of as a special case of the eigenvariate if the corresponding eigenvector weights all voxels in a cluster equally. Effectively, the eigenvariate provides a weighted mean where atypical voxels are downweighted.

Statistics: *p-values adjusted for search volume*

cluster-level				peak-level					mm mm mm		
$p_{FWE\text{-}corr}$	$q_{FDR\text{-}corr}$	k_E	p_{uncorr}	$p_{FWE\text{-}corr}$	$q_{FDR\text{-}corr}$	T	$(Z_{\equiv})$	p_{uncorr}			
0.000	**0.000**	**300**	**0.000**	**0.000**	**0.000**	**9.14**	**6.84**	**0.000**	**44**	**16**	**0**
				0.000	0.000	8.45	6.51	0.000	48	12	4
				0.041	0.833	5.29	4.64	0.000	38	12	16

table shows 32 local maxima more than 4.0mm apart

Height threshold: T = 5.21, p = 0.000 (0.050)
Extent threshold: k = 0 voxels, p = 1.000 (0.050)
Expected voxels per cluster, <k> = 2.519
Expected number of clusters, <c> = 0.05
FWEp: 5.213, FDRp: 6.702, FWEc: 1, FDRc: 20

Degrees of freedom = [1.0, 45.0]
FWHM = 9.8 10.6 15.6 mm mm mm; 4.9 5.3 3.9 {voxels}
Volume: 880432 = 55027 voxels = 472.2 resels
Voxel size: 2.0 2.0 4.0 mm mm mm; (resel = 102.26 voxels)

Figure 31.10: *SPM results table for a single cluster with p-values corrected for the whole brain.*

Statistics: *search volume: 30.0mm sphere at [44,16,0]*

set-level		cluster-level				peak-level					mm mm mm		
p	c	$p_{FWE\text{-}corr}$	$q_{FDR\text{-}corr}$	k_E	p_{uncorr}	$p_{FWE\text{-}corr}$	$q_{FDR\text{-}corr}$	T	$(Z_{\equiv})$	p_{uncorr}			
0.000	**5**	**0.000**	**1.000**	**300**	**0.000**	**0.000**	**0.000**	**9.14**	**6.84**	**0.000**	**44**	**16**	**0**
						0.000	0.000	8.45	6.51	0.000	48	12	4
						0.006	1.000	5.29	4.64	0.000	38	12	16
		0.002	**1.000**	**3**	**0.257**	**0.004**	**0.548**	**5.44**	**4.74**	**0.000**	**44**	**6**	**28**
		0.001	**1.000**	**5**	**0.148**	**0.004**	**0.589**	**5.40**	**4.72**	**0.000**	**50**	**2**	**24**
						0.005	1.000	5.34	4.68	0.000	46	6	20
		0.004	**1.000**	**1**	**0.520**	**0.005**	**1.000**	**5.37**	**4.70**	**0.000**	**34**	**36**	**20**
		0.002	**1.000**	**3**	**0.257**	**0.005**	**1.000**	**5.34**	**4.68**	**0.000**	**36**	**30**	**20**

table shows 16 local maxima more than 4.0mm apart

Height threshold: T = 5.21, p = 0.000 (0.008)
Extent threshold: k = 0 voxels, p = 1.000 (0.008)
Expected voxels per cluster, <k> = 2.519
Expected number of clusters, <c> = 0.01
FWEp: 4.510, FDRp: 6.344, FWEc: 312, FDRc: 312

Degrees of freedom = [1.0, 45.0]
FWHM = 9.8 10.6 15.6 mm mm mm; 4.9 5.3 3.9 {voxels}
Volume: 66128 = 4133 voxels = 69.1 resels
Voxel size: 2.0 2.0 4.0 mm mm mm; (resel = 102.26 voxels)

Figure 31.11: *SPM results table for a single cluster with p-values corrected using the Small Volume Correction (SVC) option. This used a 30mm sphere centred at 44 16 0. Note the reduced number of voxels in the search volume (bottom right text in Figure) and more significant p-values as compared to Figure 31.10.*

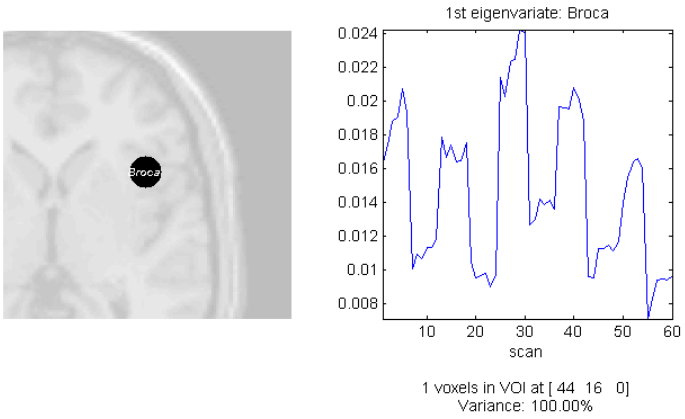


Figure 31.12: *Data extracted from a Volume of Interest (VOI).*

A file called `VOI_regionname.mat` is created in the working directory containing Y and VOI details (in the data structure xY).

31.3.9 Inclusive Masking

We have so far looked at the *average* effect over the five subjects in our group using the 'All: Gen ¿ Shad' contrast. To assess condition effects that are *common* to all subjects, one can either mask (inclusively) the 'All: Gen ¿ Shad' contrast with the individual contrasts, or perform a conjunction analysis. Firstly we'll use the inclusive masking approach.

- Press the 'Results' button.
- Select the SPM.mat file.
- Select the `All: Gen > Shad` contrast and press 'Done'.
- Mask with other contrast(s) [Yes]
- Then hold down the [control] button whilst selecting all the individual contrasts. The contrast manager should then appear as in Figure 31.13.
- Uncorrected mask p-value [0.05]
- Nature of mask [inclusive]
- Title for comparison [accept the default]
- p value adjustment to control [FWE]
- Family-wise p-value [0.05]
- Extent threshold voxels [0]

This should produce the MIP and results table shown in Figure 31.14.

31.3.10 Conjunctions

To perform a conjunction approach across subjects:

- Press the 'Results' button.
- Select the SPM.mat file.

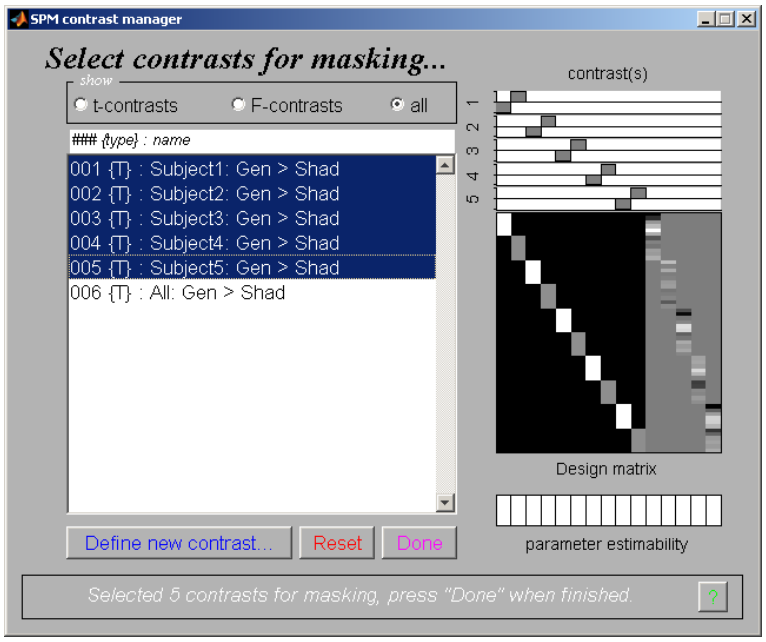


Figure 31.13: *SPM can produce maps based on multiple contrasts by holding down [control] whilst selecting contrasts. This can be used during masking and when making a conjunction inference.*

- Then hold down the [control] button whilst selecting all the individual contrasts. The contrast manager should then appear as in Figure 31.13 (except that, in the white text at the bottom, it should indicate that a conjunction will be performed).
- Null hyp. to assess [Global]
- Mask with other contrasts [No]
- Title for comparison [accept the default]
- p value adjustment to control [FWE]
- Family-wise p-value [0.05]
- Extent threshold voxels [0]

SPM checks whether the contrasts are orthogonal and, if not, makes them so. Contrasts are orthogonolized with respect to the first contrast specified.

SPM should produce the MIP and table of results shown in Figure 31.15. The p-value (corrected or uncorrected) refers to the threshold of the conjunction. SPM will compute corresponding thresholds for individual contrasts. For uncorrected thresholds, the individual threshold will be p^1/n, where p is the individual threshold and n is the number of contrasts in the conjunction.

Height, and not extent, is used to specify thresholding because the distributional approximations for the spatial extent of a conjunction SPM are not known (at present), so that inference based on spatial extent is precluded.

Although the MIP's of the masked group contrast and the conjunction are similar, for the conjunction an intersection SPM or 'minimum T-field' is computed. This intersection is the same as thresholding a map of the minimum T-values. If the smallest T-value is above the specified threshold then all the T-values associated with the component SPMs are above threshold.

Conjunction SPMs are very useful for testing multiple hypotheses (each component hypothesis being specified by a contrast). In this example, we have chosen to use the Global Null Hypothesis. The set of hypotheses tested jointly is that the first subject did not activate, the second subject did not activate and so on.

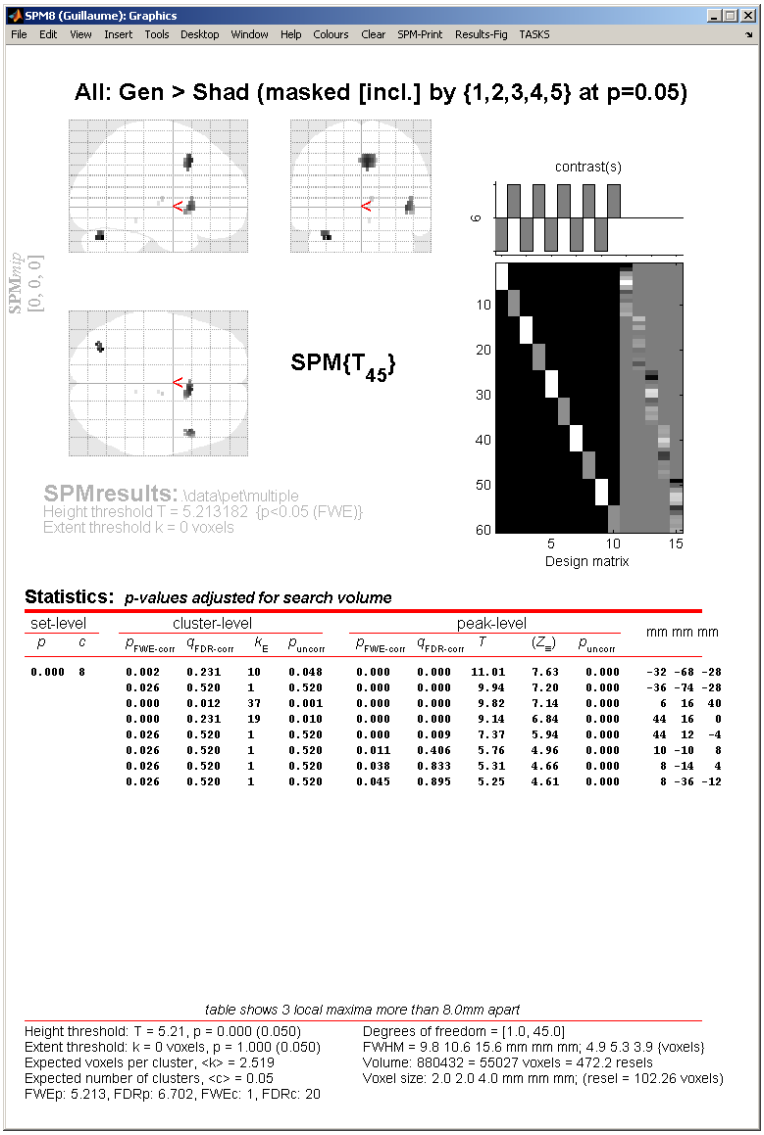


Figure 31.14: *The SPM shows results from the inclusive masking approach. It shows all voxels which are (a) significant at $p < 0.05$ corrected across all subjects and (b) significant at $p < 0.05$ uncorrected for each subject individually.*

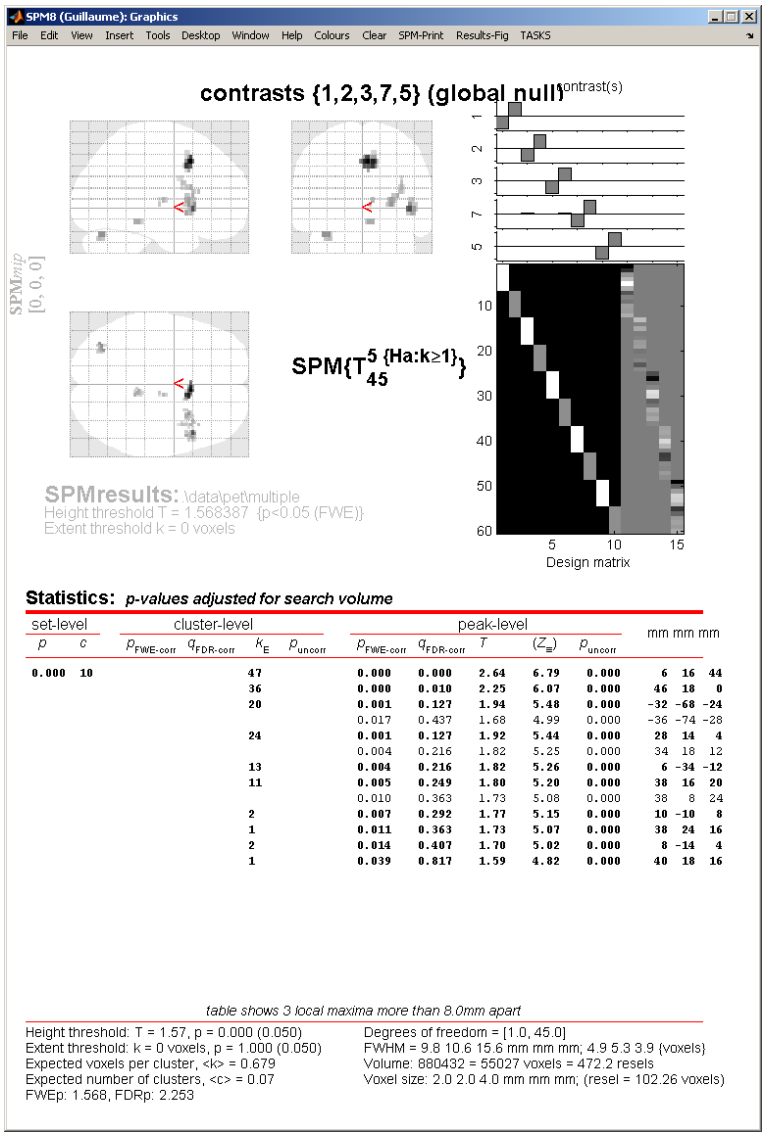


Figure 31.15: *Conjunction SPM.*

SPM also provides an option to use the Conjunction Null hypothesis. This can be thought of as enabling an inference that subject 1 activated AND subject 2 activated AND subject 3... etc. For more discussion on this issue, see [31] and [59].

Gaussian field theory results are available for SPMs of minimum T- (or F-) statistics and therefore corrected p-values can be computed. Note that the minimum T-values do not have the usual Student's T-distribution and small minimum T-values can be very significant.

Chapter 32

Dynamic Causal Modeling for fMRI

32.1 Theoretical background

Dynamic Causal Modelling (DCM) is a method for making inferences about neural processes that underlie measured time series, e.g. fMRI data. The general idea is to estimate the parameters of a reasonably realistic neuronal system model such that the predicted blood oxygen level dependent (BOLD) signal, which results from converting the modeled neural dynamics into hemodynamic responses, corresponds as closely as possible to the observed BOLD time series. This section gives a short introduction to the theoretical background of DCM for fMRI; details can be found in [27]. Note that DCMs can be formulated, in principle, for any measurement technique. Depending on the spatio-temporal properties of a given measurement technique, one needs to define an adequate state equation and an observation model. See Fig 32.1 for a summary of the differences between DCM implementations for fMRI and Event-Related Potentials (ERPs).

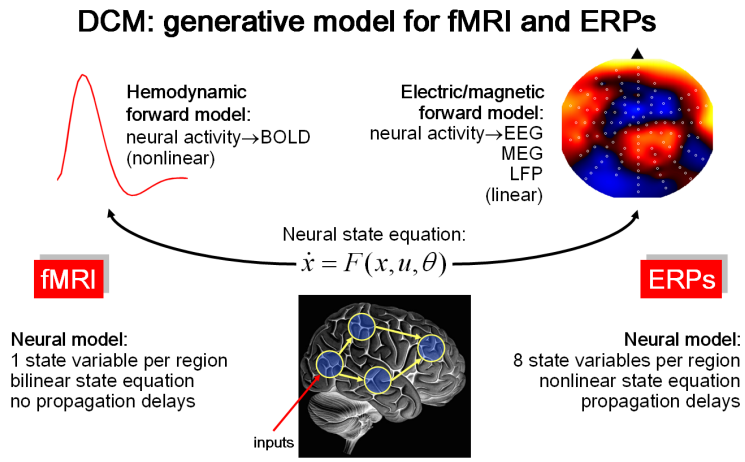


Figure 32.1: *A schematic overview of the differences between between the DCM implementations for fMRI and ERPs (as measured by EEG or MEG). Whereas the state equation of DCM for fMRI is bilinear and uses only a single state variable per region, that for ERPs is more complex and requires 8 state variables per region. Moreover, DCM for ERPs models the delays of activity propagation between areas. At the level of the observation model, DCM for fMRI is more complex than DCM for ERPs. While the former uses a non-linear model of the hemodynamic response that contains a cascade of differential equations with five state variables per region, the latter uses a simple linear model for predicting observed scalp data.*

As in state-space models, two distinct levels constitute a DCM (see Figure 32.2). The hidden level, which cannot be directly observed using fMRI, represents a simple model of neural dynamics in a system of k coupled brain regions. Each system element i is represented by a single state variable z_i, and the dynamics of the system is described by the change of the neural state vector over time.

The neural state variables do not correspond directly to any common neurophysiological measurement (such as spiking rates or local field potentials) but represent a summary index of neural population dynamics in the respective regions. Importantly, DCM models how the neural dynamics are driven by external perturbations that result from experimentally controlled manipulations. These perturbations are described by means of external inputs u that enter the model in two different ways: they can elicit responses through direct influences on specific regions ("driving" inputs, e.g. evoked responses in early sensory areas) or they can change the strength of coupling among regions ("modulatory" inputs, e.g. during learning or attention).

Overall, DCM models the temporal evolution of the neural state vector, i.e. , as a function of the current state, the inputs u and some parameters that define the functional architecture and interactions among brain regions at a neuronal level (n denotes "neural"):

$$\begin{bmatrix} \dot{z}_1 \\ \dot{z}_2 \\ .. \\ \dot{z}_k \end{bmatrix} = \dot{z} = \frac{dz}{dt} = F(z, u, \theta^n) \tag{32.1}$$

In this neural state equation, the state z and the inputs u are time-dependent whereas the parameters are time-invariant. In DCM, F has the bilinear form

$$\dot{z} = Az + \sum_{j=1}^{m} u_j B_j z + Cu \tag{32.2}$$

The parameters of this bilinear neural state equation, $\theta^n = \{A, B_1, ..., B_m, C\}$, can be expressed as partial derivatives of F:

$$\begin{aligned} A &= \frac{\partial F}{\partial z} = \frac{\partial \dot{z}}{\partial z} \\ B_j &= \frac{\partial^2 F}{\partial z \partial u_j} = \frac{\partial}{\partial u_j}\frac{\partial \dot{z}}{\partial z} \\ C &= \frac{\partial F}{\partial u} \end{aligned} \tag{32.3}$$

These parameter matrices describe the nature of the three causal components which underlie the modeled neural dynamics: (i) context-independent effective connectivity among brain regions, mediated by anatomical connections ($k \times k$ matrix A), (ii) context-dependent changes in effective connectivity induced by the jth input u_j ($k \times k$ matrices B_1, ..., B_m), and (iii) direct inputs into the system that drive regional activity ($k \times m$ matrix C). As will be demonstrated below, the posterior distributions of these parameters can inform us about the impact that different mechanisms have on determining the dynamics of the model. Notably, the distinction between "driving" and "modulatory" is neurobiologically relevant: driving inputs exert their effects through direct synaptic responses in the target area, whereas modulatory inputs change synaptic responses in the target area in response to inputs from another area. This distinction represents an analogy, at the level of large neural populations, to the concept of driving and modulatory afferents in studies of single neurons.

DCM combines this model of neural dynamics with a biophysically plausible and experimentally validated hemodynamic model that describes the transformation of neuronal activity into a BOLD response. This so-called "Balloon model" was initially formulated by Buxton and colleagues and later extended by [29]. Briefly summarized, it consists of a set of differential equations that describe the relations between four hemodynamic state variables, using five parameters (θ^h). More specifically, changes in neural activity elicit a vasodilatory signal that leads to increases in blood flow and subsequently to changes in blood volume v and deoxyhemoglobin content q. The predicted BOLD signal y is a non-linear function of blood volume and deoxyhemoglobine content.

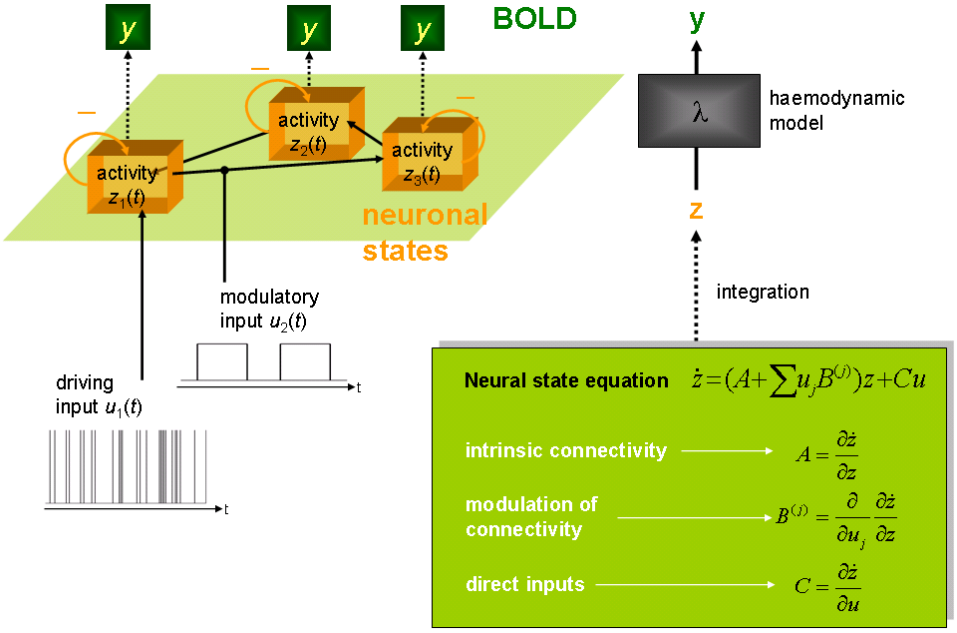


Figure 32.2: *Schematic summary of the conceptual basis of DCM. The dynamics in a system of interacting neuronal populations (orange boxes), which are not directly observable by fMRI, is modeled using a bilinear state equation (grey box). Integrating the state equation gives predicted neural dynamics (z) that enter a model of the hemodynamic response (λ) to give predicted BOLD responses (y) (green boxes). The parameters at both neural and hemodynamic levels are adjusted such that the differences between predicted and measured BOLD series are minimized. Critically, the neural dynamics are determined by experimental manipulations. These enter the model in the form of "external" or "driving" inputs. Driving inputs (u_1; e.g. sensory stimuli) elicit local responses directly that are propagated through the system according to the intrinsic connections. The strengths of these connections can be changed by modulatory inputs (u_2; e.g. changes in cognitive set, attention, or learning).*

Details of the hemodynamic model can be found in other publications [29]. By combining the neural and hemodynamic states into a joint state vector x and the neural and hemodynamic parameters into a joint parameter vector $\theta = [\theta^n, \theta^h]^T$, we obtain the full forward model that is defined by the neural and hemodynamic state equations

$$\begin{aligned} \dot{x} &= F(x, u, \theta) \\ y &= \lambda(x) \end{aligned} \tag{32.4}$$

For any given set of parameters θ and inputs u, the joint state equation can be integrated and passed through the output nonlinearity λ to give a predicted BOLD response $h(u, \theta)$. This can be extended to an observation model that includes observation error e and confounding effects X (e.g. scanner-related low-frequency drifts):

$$y = h(u, \theta) + X\beta + e \tag{32.5}$$

This formulation is the basis for estimating the neural and hemodynamic parameters from the measured BOLD data, using a fully Bayesian approach with empirical priors for the hemodynamic parameters and conservative shrinkage priors for the neural coupling parameters.

Details of the parameter estimation scheme, which rests on a Fisher scoring gradient ascent scheme with Levenburg-Marquardt regularisation, embedded in an expectation maximization (EM) algorithm, can be found in the original DCM publication (Friston et al. 2003). In brief, under Gaussian assumptions about the posterior distributions, this scheme returns the posterior

expectations $\eta_{\theta|y}$ and posterior covariance $C_{\theta|y}$ for the parameters as well as hyperparameters for the covariance of the observation noise, C_e.

After fitting the model to measured BOLD data, the posterior distributions of the parameters can be used to test hypotheses about the size and nature of effects at the neural level. Although inferences could be made about any of the parameters in the model, hypothesis testing usually concerns context-dependent changes in coupling (i.e. specific parameters from the B matrices; see Fig. 32.7). As will be demonstrated below, at the single-subject level, these inferences concern the question of how certain one can be that a particular parameter or, more generally, a contrast of parameters, $c^T\eta_{\theta|y}$, exceeds a particular threshold γ (e.g. zero).

Under the assumptions of the Laplace approximation, this is easy to test (Φ_N denotes the cumulative normal distribution):

$$p(c^T\eta_{\theta|y} > \gamma) = \Phi_N\left(\frac{c^T\eta_{\theta|y} - \gamma}{c^T C_{\theta|y} c}\right) \tag{32.6}$$

For example, for the special case $c^T\eta_{\theta|y} = \gamma$ the probability is $p(c^T\eta_{\theta|y} > \gamma) = 0.5$, i.e. it is equally likely that the parameter is smaller or larger than the chosen threshold γ. We conclude this section on the theoretical foundations of DCM by noting that the parameters can be understood as rate constants (units: $1/s = Hz$) of neural population responses that have an exponential nature. This is easily understood if one considers that the solution to a linear ordinary differential equation of the form $\dot{z} = Az$ is an exponential function (see Fig. 32.3).

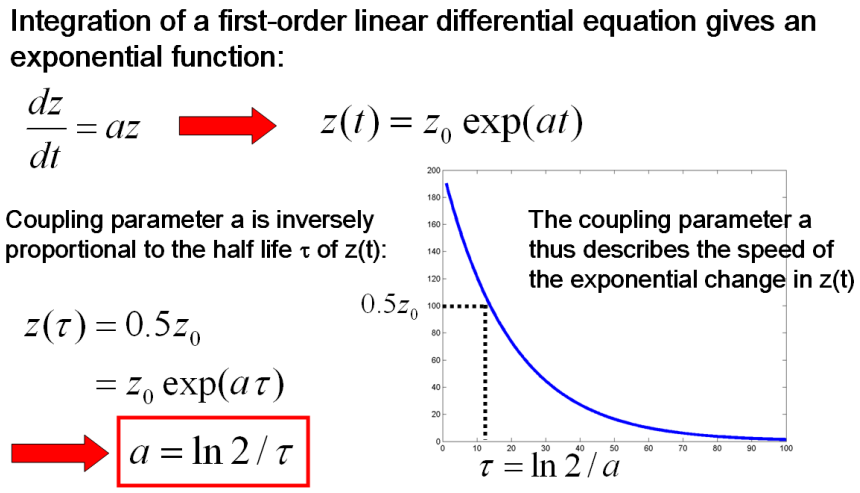


Figure 32.3: *A short mathematical demonstration, using a simple linear first-order differential equation as an example, explaining why the coupling parameters in a DCM are inversely proportional to the half-life of the modelled neural responses and are therefore in units of 1/s = Hertz.*

32.2 Bayesian model selection

A generic problem encountered by any kind of modeling approach is the question of model selection: given some observed data, which of several alternative models is the optimal one? This problem is not trivial because the decision cannot be made solely by comparing the relative fit of the competing models. One also needs to take into account the relative complexity of the models as expressed, for example, by the number of free parameters in each model.

Model complexity is important to consider because there is a trade-off between model fit and generalizability (i.e. how well the model explains different data sets that were all generated from the same underlying process). As the number of free parameters is increased, model fit increases monotonically whereas beyond a certain point model generalizability decreases. The reason for

this is "overfitting": an increasingly complex model will, at some point, start to fit noise that is specific to one data set and thus become less generalizable across multiple realizations of the same underlying generative process.

Therefore, the question "What is the optimal model?" can be reformulated more precisely as "What is the model that represents the best balance between fit and complexity?". In a Bayesian context, the latter question can be addressed by comparing the evidence, $p(y|m)$, of different models. According to Bayes theorem

$$p(\theta|y,m) = \frac{p(y|\theta,m)p(\theta|m)}{p(y|m)} \tag{32.7}$$

the model evidence can be considered as a normalization constant for the product of the likelihood of the data and the prior probability of the parameters, therefore

$$p(y|m) = \int p(\theta|y,m)p(\theta|m)d\theta \tag{32.8}$$

Here, the number of free parameters (as well as the functional form) are considered by the integration. Unfortunately, this integral cannot usually be solved analytically, therefore an approximation to the model evidence is needed. One such approximation used by DCM, and many other models in SPM, is to make use of the Laplace approximation [1].

As shown in [64], this yields the following expression for the natural logarithm (ln) of the model evidence ($\eta_{\theta|y}$ denotes the posterior mean, $C_{\theta|y}$ is the posterior covariance of the parameters, C_e is the error covariance, θ_p is the prior mean of the parameters, and C_p is the prior covariance):

$$\begin{aligned} lnp(y|m) &= accuracy(m) - complexity(m) \\ &= \left[-\frac{1}{2}ln|C_e| - \frac{1}{2}(y - h(u,\eta_{\theta|y}))^T C_e^{-1}(y - h(u,\eta_{\theta|y}))\right] \\ &- \left[\frac{1}{2}ln|C_p| - \frac{1}{2}ln|C_{\theta|y}| + \frac{1}{2}(\eta_{\theta|y} - \theta_p)^T C_p^{-1}(\eta_{\theta|y} - \theta_p)\right] \end{aligned} \tag{32.9}$$

This expression properly reflects the requirement, as discussed above, that the optimal model should represent the best compromise between model fit (accuracy) and model complexity. The complexity term depends on the prior density, for example, the prior covariance of the intrinsic connections.

Two models can then be compared using the Bayes factor:

$$BF_{ij} = \frac{p(y|m_i)}{p(y|m_j)} \tag{32.10}$$

Given uniform priors over models, the posterior probability for model i is greater 0.95 if BF_{ij} is greater than twenty.

This results in a robust procedure for deciding between competing hypotheses represented by different DCMs. These hypotheses can concern any part of the structure of the modeled system, e.g. the pattern of intrinsic connections or which inputs affect the system and where they enter. Note, however, that this comparison is only valid if the data y are identical in all models. This means that in DCM for fMRI, where the data vector results from a concatenation of the time series of all areas in the model, only models can be compared that contain the same areas. Therefore, model selection cannot be used to address whether or not to include a particular area in the model. In contrast, in DCM for ERPs, the data measured at the sensor level are independent of how many neuronal sources are assumed in a given model. Here, model selection could also be used to decide which sources should be included.

32.3 Practical example

The following example refers to the "attention to visual motion" data set available from the SPM web site[2]. This data set was obtained by Christian Buchel and is described in [13]. Note that the

[1] This should perhaps more correctly be referred to as a fixed-form variational approximation, where the fixed form is chosen to be a Gaussian. The model evidence is approximated by the negative free energy, F.

[2] Attention to visual motion dataset: http://www.fil.ion.ucl.ac.uk/spm/data/attention/

data available from the web have been pre-processed using SPM99, therefore ensure `flip` is set to `1` in the `spm_flip_analyze_images.m` file before proceeding further.

Making a DCM requires two ingredients: (i) a design matrix and (ii) the time series, stored in VOI files. The regressors of the design matrix define the inputs for the DCM. Note that this means that the design matrix that is optimal for a given DCM is often somewhat different than the one for the corresponding GLM. DCM does not require the design matrix to be part of an estimated model, however. It just needs to be defined.

32.3.1 Defining the GLM

The present experiment consisted of 4 conditions: (i) fixation (F), (ii) static (S, non-moving dots), (iii) no attention (N, moving dots but no attention required), (iv) attention (A). The GLM analyses by Christian showed that activity in area V5 was not only enhanced by moving stimuli, but also by attention to motion. In the following, we will try to model this effect in V5, and explain it as a context-dependent modulation or "enabling" of V5 afferents, using a DCM. First, we need to set up the GLM analysis and extract our time series from the results. In this example, we want to use the same design matrix for GLM and DCM, therefore we recombine the above regressors to get the following three conditions:

1. **photic**: this comprises all conditions with visual input, i.e. S, N, and A.
2. **motion**: this includes all conditions with moving dots, i.e. N and A.
3. **attention**: this includes the attention-to-motion (A) condition only.

Now we need to define and estimate the GLM. See chapters 8 and 9 on how to do this. Here are the relevant details for this data set that you need to set up the GLM (this information can also be found online[3] note this web site describes the analysis for SPM2).

- The onsets for the conditions can be found in the file factors.mat. They are named phot (photic), mot (motion) and att (attention) and are defined in scans (not seconds!). They are blocks of 10 TRs each.
- The TR is 3.22 seconds.
- There are 360 scans.

32.3.2 Extracting time series

Once you have specified and estimated the GLM, you should define t-contrasts that test for photic, motion, and attention, respectively. These serve to locate areas that show effects due to visual stimulation (e.g. in V1), motion (e.g. V5) and attention (e.g. V5 and superior parietal cortex, SPC). Because V5 shows both motion and attention effects, it is useful to mask the motion-contrast inclusively with the attention-contrast when extracting time series for V5. You should also compute the usual "effects of interest" F-contrast, this is needed for mean-correcting the extracted time series (see below). Here is a step-by-step example for extracting the V5 time series:

1. Press "Results".
2. Select the `SPM.mat` file.
3. Choose the t-contrast for the motion condition.
4. Mask with other contrasts: Yes
5. Choose the t-contrast for the attention condition.
6. Mask inclusively and choose a threshold of $p \leq 0.05$ uncorrected.

[3] ftp://ftp.fil.ion.ucl.ac.uk/spm/data/attention/README_GLM_DCM.txt

7. Select the local maximum of a blob that looks V5-ish, e.g. -45/-81/-9 (by overlaying the activations onto the normalised structural image you should be able to identify V5 more easily).
8. Press the "VOI" button.
9. Name of region: V5
10. Adjust data for: effects of interest (this mean-corrects the time series)
11. VOI definition: sphere
12. VOI radius(mm): e.g. 8 mm

SPM now computes the first principal component of the time series from all voxels included in the sphere. The result is stored (together with the original time series) in a file named `VOI_V5_1.mat` in the working directory. You can now proceed to select time series for V1 (using the "photic" contrast) and SPC (using the "attention" contrast). For this example, we selected -6/-84/-6 for V1 and -18/-57/66 for SPC.

32.3.3 Specifying and estimating the DCM

Now we have defined the inputs (via the design matrix) and the time series, we are ready to build the DCM. We will look at a simplified version of the model described in [27]. In our example here, we will model a hierarchically connected system comprising V1, V5 and SPC, i.e. reciprocal connections between V1-V5 and V5-SPC, but not between V1-SPC. We will assume that (i) V1 is driven by any kind of visual stimulation (direct input "photic"), (ii) motion-related responses in V5 can be explained through an increase in the influence of V1 onto V5 whenever the stimuli are moving (i.e. "motion" acts as modulatory input onto the $V1 \rightarrow V5$ connection) and (iii) attention enhances the influence of SPC onto V5 (i.e. "attention" acts as modulatory input onto the $SPC \rightarrow V5$ connection). This DCM is shown schematically in Figure 32.4, and can be made as follows:

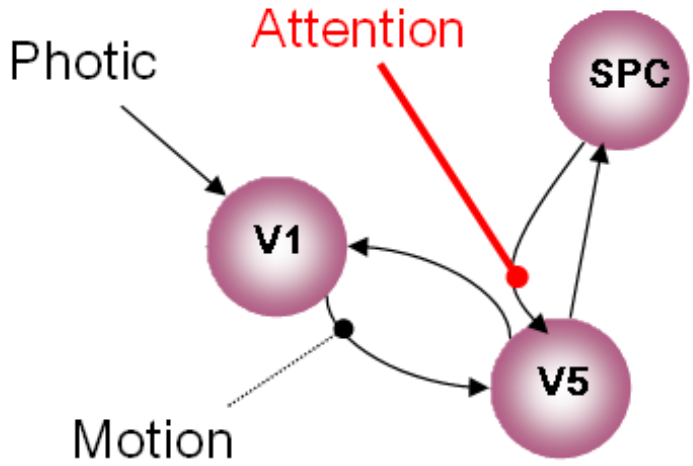


Figure 32.4: *DCM with attentional modulation of backwards connection. Dotted lines denote modulatory connections.*

1. Press the `DCM` button.
2. Choose "specify".

3. Select the `SPM.mat` file you just created when specifying the GLM.
4. Name for `DCM_???.mat`: e.g. `mod_bwd` (for "attentional modulation of backward connection")
5. Select all VOIs in order `VOI_V1_1, VOI_V5_1, VOI_SPC_1`
6. Include Photic: Yes
7. Include Motion: Yes
8. Include Attention: Yes
9. Define the following intrinsic connections: V1 to V5, V5 to V1, V5 to SPC, SPC to V5, i.e. a hierarchy with reciprocal connections between neighbouring areas. Note that the columns specify the source of the connection and the rows specify its target. Your connectivity matrix should look like the one in Fig. 32.5.
10. Specify Photic as a driving input into V1. See Fig. 32.6
11. Specify Motion to modulate the connection from V1 to V5. See Fig. 32.7
12. Specify Attention to modulate the connection from SPC to V5. See Fig. 32.8
13. Say No for "Nonlinear DCM?"
14. Specify slice timings for each area. This is a new option described in [51]. The default values are set to the last slice of the data, which was the default in the original DCM version. For sequential (as opposed to interleaved) data, this modelling option allows to use DCM in combination with any TR (slice timing differences). Here, we proceed with the default values.
15. Enter for "Echo Time, TE[s]".

A polite "Thank you" completes the model specification process.

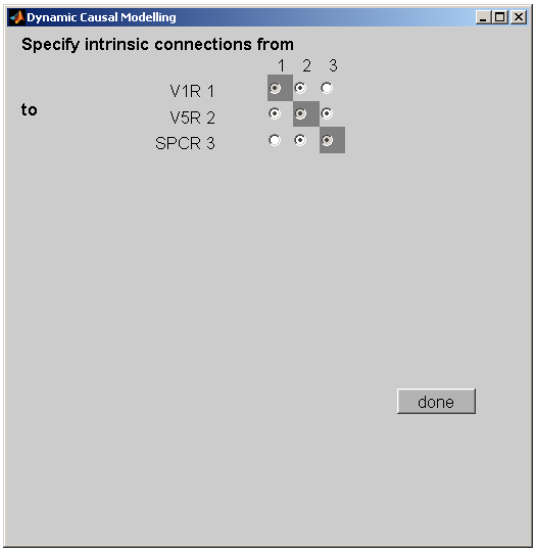


Figure 32.5: *Filled circles define the structure of the intrinsic connections C such that eg. there are no connections from V1R to SPCR or from SPCR to V1R. See also Fig 32.4.*

You can now estimate the model parameters, either by pressing the DCM button again and choosing "estimate", or by typing `spm_dcm_estimate('DCM_mod_bwd')` from the MATLAB command line. Once this is completed, you can review the results as follows:

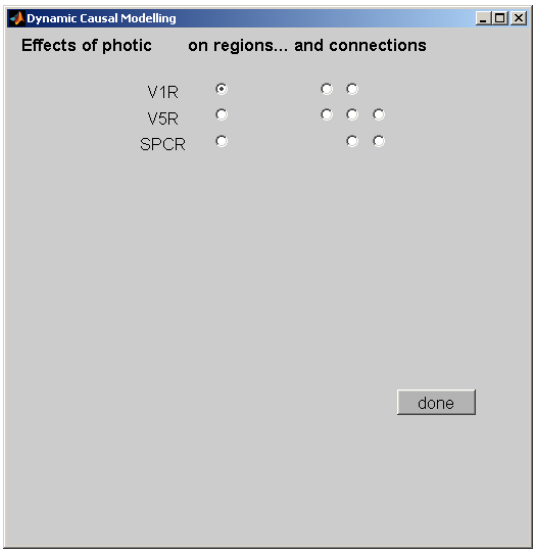


Figure 32.6: *The filled circle specifies that the input 'phot' connects to region V1R. See also Fig 32.4.*

1. Press the DCM button.
2. Choose "review".
3. Select `DCM_mod_bwd`
4. Threshold: 0

Now you have multiple options, e.g. you can revisit the fit of the model ("Outputs") or look at the parameter estimates for the intrinsic connections ("Intrinsic connections") or for the parameters associated with the driving or modulatory inputs ("Effects of Photic", "Effects of Motion", "Effects of Attention").

Also, you can use the "Contrasts" option to determine how confident you can be that a contrast of certain parameter estimates exceeds the threshold you chose in step 4. Of course, you can also explore the model results at the level of the MATLAB command line by loading the model and inspecting the parameter estimates directly. These can be found in `DCM.A` (intrinsic connections), `DCM.B` (modulatory inputs) and `DCM.C` (driving inputs).

32.3.4 Comparing models

Let us now specify an alternative model and compare it against the one that we defined and estimated above. The change that we are going to make is to assume that attention modulates the $V1 \rightarrow V5$ connection (as opposed to the $SPC \rightarrow V5$ connection in the previous model). For defining this model, you repeat all the steps from the above example, the only differences being that the model gets a new name (e.g. `mod_fwd`) and that attention now acts on the forward connection. This DCM is shown schematically in Figure 32.9. Once you have estimated this new model, you can perform a Bayesian model comparison as follows:

1. Press the "DCM" button.
2. Choose "compare".
3. Number of models to compare: 2
4. Select the two models, e.g. in the order `DCM_mod_bwd` and `DCM_mod_fwd`.

The Graphics window, Fig. 32.10, now shows a bar plot of the model evidence. You can see that our second model is better than the first one.

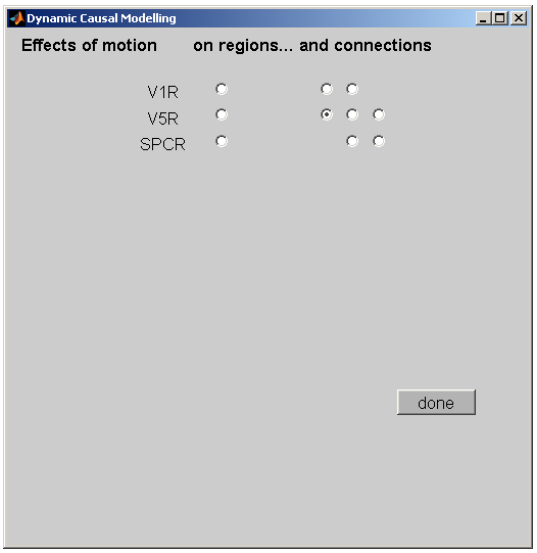


Figure 32.7: *The filled circle indicates that the input variable 'mot' can modulate the connection from V1R to V5R. This specifies a 'modulatory' connection. See also Fig 32.4.*

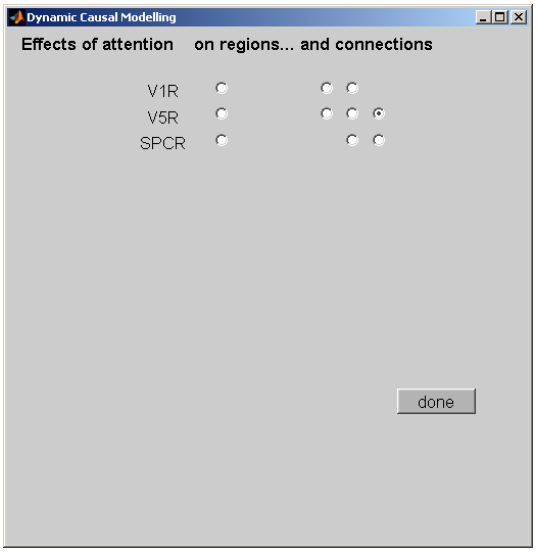


Figure 32.8: *The filled circle indicates that attention can modulate the connection from SPCR to V5R. See also Fig 32.4.*

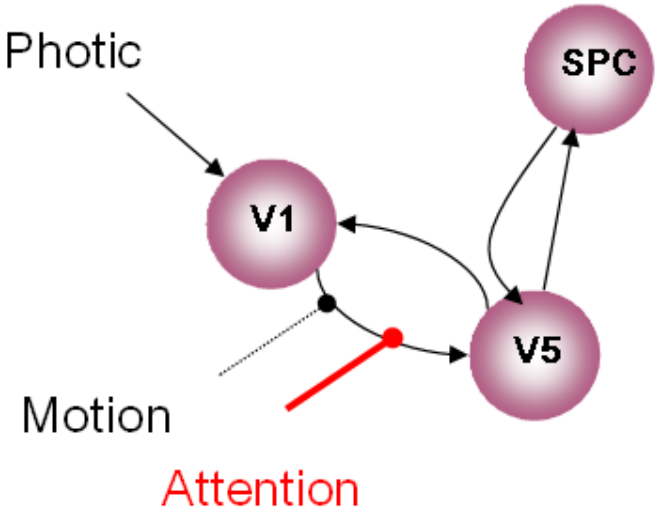


Figure 32.9: *DCM with attentional modulation of forwards connection. Dotted lines denote modulatory connections.*

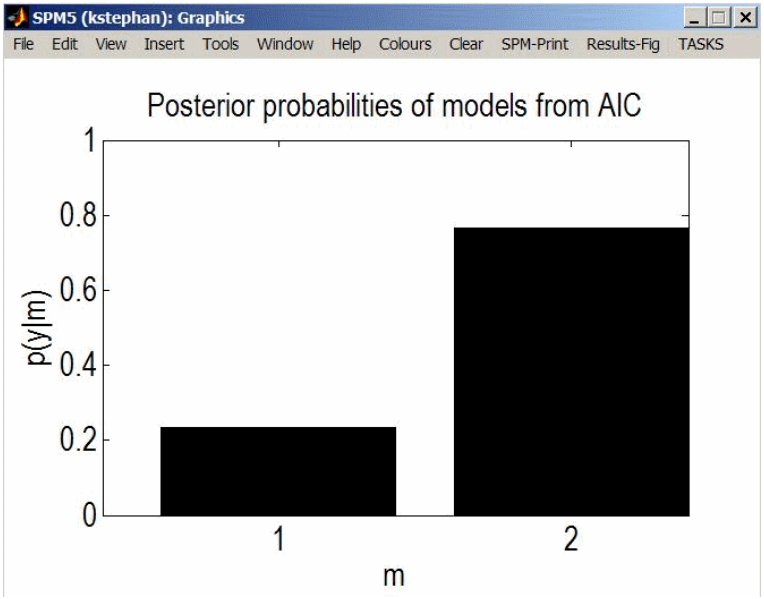


Figure 32.10: *Model 2 (shown in Fig 32.9) is preferred to model 1 (shown in Fig 32.4).*

Chapter 33

Mismatch negativity data

This chapter describes the analysis of a 128-channel single subject EEG data set acquired from a study of mismatch negativity in the auditory system [33]. We thank Marta Garrido for providing us with these data. The experiment comprised an auditory oddball paradigm in which subjects heard standard (500Hz) and deviant (550Hz) tones, occuring 80% (480 trials) and 20% (120 trials) of the time, respectively, in a pseudo-random sequence subject to the constraint that two deviant tones did not occur together.

EEG data were recorded with a Biosemi[1] system at 128 scalp electrodes and a sampling rate of 512Hz. Vertical and horizontal eye movements were monitored using EOG electrodes. See [33] for full details of experimental stimuli and recording. To proceed with the data analysis, first download the data set from the SPM website[2]. The data comprises a file called `subject1.bdf` whose size is roughly 200MB. We will refer to the directory in which you have placed it as `DATA_DIR`. This chapter takes you through different stages of analysis:

- Preprocessing
- Sensor space analysis
- Source reconstruction
- Dynamic Causal Modelling

33.1 Preprocessing

There is also an example MATLAB script under `man\example_scripts\history_subject1.m` in the SPM distribution which repeats the preprocessing route we take here.

33.1.1 Convert

At the MATLAB prompt type `spm eeg`, press the CONVERT button and select the `subject1.bdf` file. At the prompt "Define settings ?" select "just read".

SPM will now read the original Biosemi format file and create an SPM combatible data file, called `spm8_subject1.mat` and `spm8_subject1.dat` in the directory containing the original data file (`DATA_DIR`).

33.1.2 Montage

In this step, we will identify the VEOG and HEOG channels, and also remove several channels that don't carry EEG data and are of no importance to the following. We generally recommend to remove all data channels that are no longer needed because it will reduce the total file size. To do so, we use the MONTAGE tool in SPM, which is a general approach for pre-multiplying the

[1] BioSemi: http://www.biosemi.com/
[2] EEG MMN dataset: http://www.fil.ion.ucl.ac.uk/spm/data/eeg_mmn/

data matrix (channels × time) by another matrix that linearly weights all channel data. This provides a very general method for data transformation in M/EEG analysis.

The appropriate montage-matrix can be derived as follows. In our case, we would like to only keep channels 1 to 128. In addition, there were three EOG channels (129, 130, 131), where the HEOG is computed as the difference between channels 131 and 130, and the VEOG by the difference between channels 130 and 129. This matrix can be specified in SPM by either using a graphical interface, or by supplying the matrix saved in a file. We will do the latter. The script to generate this file can be found in the `example_scripts` folder: `montage_subject1.m`. Copy this script into `DATA_DIR` and run it. This will generate a file named `MONT_EXP.mat`.

You now call the montage function by choosing MONTAGE in the "Other" drop-down menu and:

- Select the M/EEG-file `spm8_subject1.mat`
- 'How to specify the montage ?' Answer "file".
- Then select the generated `MONT_EXP.mat` file
- "Keep the other channels?" : "No"

This will remove the uninteresting channels from the data. The progress bar appears and SPM will generate two new files `Mspm8_subject1.mat` and `Mspm8_subject1.dat`.

This step will also assign default locations to the sensors, as this information is unfortunately not contained in the original Biosemi `*.bdf` file. It is usually the responsibility of the user to link the data to sensors which are located in a coordinate system. In our experience this is a critical step. SPM provide tools (PREPARE) for linking data and location information, leaving it the responsibility of the user to verify the success of this process. Chapter 12 describes in detail how you can use the PREPARE tool from the "Other" drop-down menu to use digitized sensor location data.

33.1.3 Epoch

To epoch the data click on EPOCHING. Select the `Mspm8_subject1.mat` file. Choose the peri-stimulus time window, first the start `-100`, then the end `400` ms. Choose 2 conditions. You can call the first condition "standard". A GUI pops up which gives you a complete list of all events in the EEG file. The standard condition had 480 trials, so select the type with value 1 and press OK. The second condition can be called "rare". The rare stimulus was given 120 times and has value 3 in the list. Select this trial type and press OK. Answer two times "no" to the questions "review individual trials", and "save trial definitions". The progress bar will appear and the epoched data will be saved to files `eMspm8_subject1.mat` and `eMspm8_subject1.dat`.

33.1.4 Downsample

Here, we will downsample the data in time. This is useful when the data were acquired like ours with a high sampling rate of 512 Hz. This is an unnecessarily high sampling rate for a simple evoked response analysis, and we will now decrease the sampling rate to 200 Hz, thereby reducing the file size by more than half. Select DOWNSAMPLE from the "Other" drop-down menu and select the `eMspm8_subject1.mat` file. Choose a new sampling rate of 200 (Hz). The progress bar will appear and the resulting data will be saved to files `deMspm8_subject1.mat` and `deMspm8_subject1.dat`. This step is computationally demanding.

33.1.5 Filter

Filtering the data in time removes unwanted frequency bands from the data. Usually, for evoked response analysis, the low frequencies are kept, while the high frequencies are assumed to carry noise. Here, we will use a bandpass filter to remove ultra-low frequencies close to DC, and also remove high frequencies at the same time

- Click on FILTER and select the `deMspm8_subject1.mat` file.

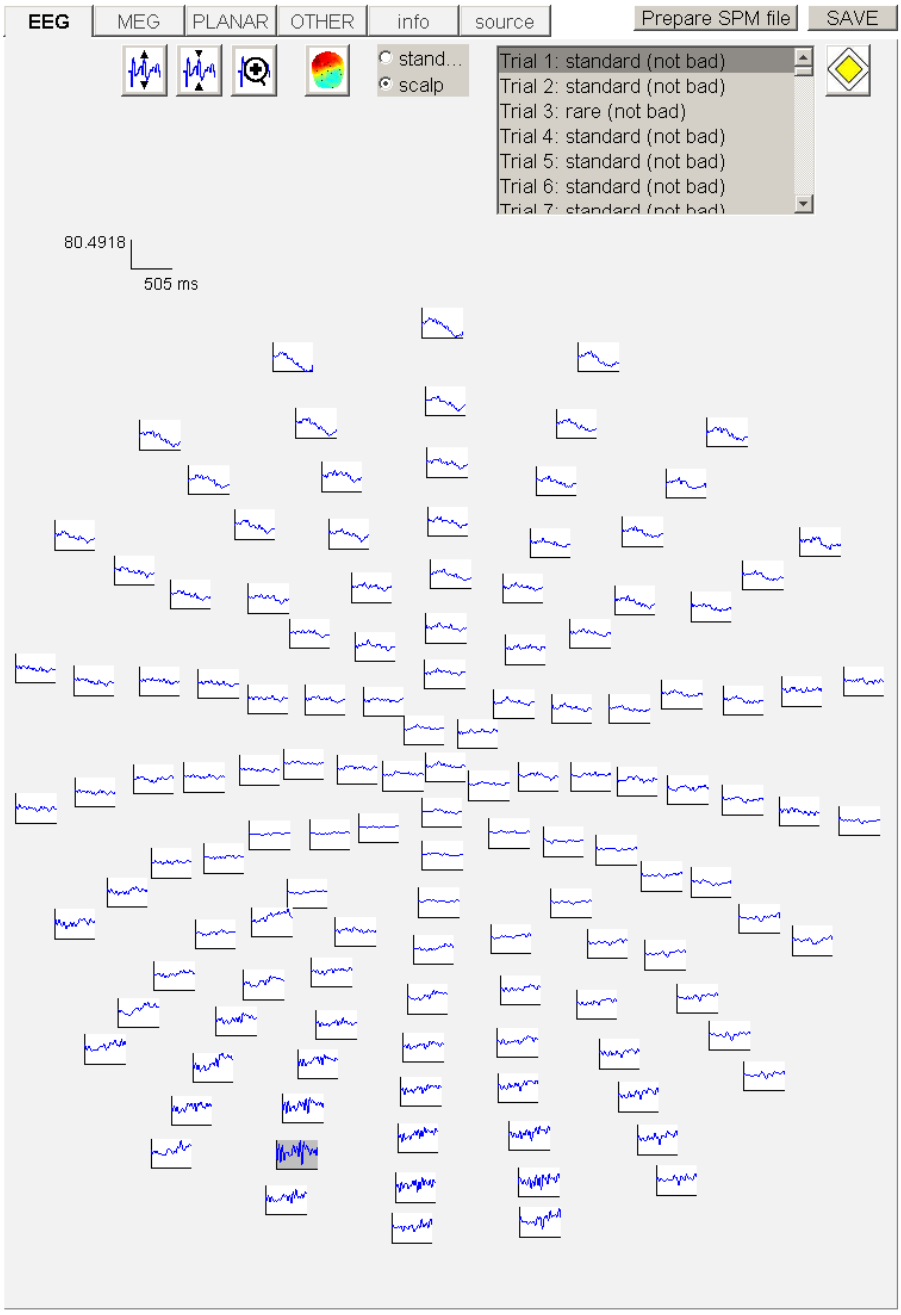


Figure 33.1: *Scalp topography of single trial MMN EEG data. Channel 14, second-row from bottom, left hemisphere contains (slightly) higher variability data than the others. This channel is to be marked as artefactual (ie. 'bad').*

- Select a "bandpass" filter with band 0.5 30 (Hz).

The progress bar will appear and the resulting filtered data will be saved in files `fdeMspm8_subject1.mat` and `fdeMspm8_subject1.dat`.

33.1.6 Artefacts

A number of different methods of artefact removal are implemented in SPM8. Here, we will demonstrate a simple thresholding method. However, before doing so, we will look at the data in the display:

- Choose "M/EEG" from the "Display" dropdown menu.
- Select the `fdeMspm8_subject1.mat` file.
- Click on the "EEG" tab.
- Press the "scalp" radio button.

The time-series for the first trial will then appear in the topographical layout shown in Figure 33.1.

You will see that Channel 14, second-row from bottom, left hemisphere, contains (slightly) higher variability data than the others . Right-click on the channel; this tells you that this channel is "A14". You will also see as an entry in this menu "bad: 0". Select this entry, and click the left button. This will make the menu disappear, but the channel now has a grey background. You have marked this channel as bad. Click on "save"in the top-right corner. This channel will then be ignored in subsequent processing. In fact this channel probably doesn't need removing, but we do so for teaching purposes only.

Now, click on ARTEFACTS and:

- Select the `fdeMspm8_subject1.mat` file.
- Answer "no" to "Read own artefact list ?".
- Answer "no" to "Robust average ?".
- Answer "yes" to "Threshold channels ?"
- Press return to choose the default for which channels you want to threshold, i.e., all 130 channels (128 EEG, 2 EOG).
- Choose as threshold 80 (μV).

The progress bar will appear and the resulting data will be saved to files `afdeMspm8_subject1.mat` and `afdeMspm8_subject1.dat`. Sixty five trials will have been rejected. There are also interactive artefact removal routines available from `Toolbox` → `MEEG tools` → `Fieldtrip visual artifact rejection`.

33.1.7 Averaging

To produce an ERP click on AVERAGING and select the `afdeMspm8_subject1.mat` file. The progress bar will appear and the resulting data will be saved to files `mafdeMspm8_subject1.mat` and `mafdeMspm8_subject1.dat`. The Graphics window will pop up and allow you to look at the averaged data. To look at the ERP, click on the EEG tab, and press the "scalp" radio button. Now hold the Shift button down on the keyboard whilst selecting trial 2 with the left mouse button in the upper right corner of the graphics window. This will overlay responses to standard and rare trials on the same figure axes.

Now press the 'plus' icon at the top of this graphics window and select channel C23 (seventh central channel down from the top) with a left mouse click. This will plot the ERPs shown in Figure 33.2. This completes the preprocessing step.

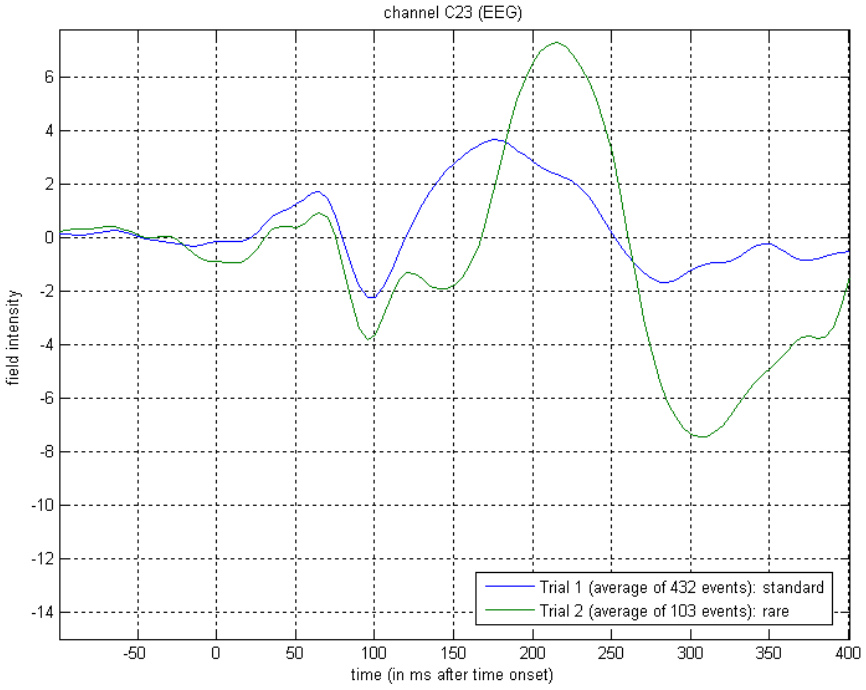


Figure 33.2: *ERPs at channel C23 (CZ) for standard and rare tones. The ERP curve for rare tones lies underneath that for standard tones between 100 and 180ms. This corresponds to the mismatch negativity signal in this subject.*

33.1.8 History

The evoked response file (and every other SPM MEEG data file) contains a history-entry which stores all of the above preprocessing steps. You can take this history and produce a script that will re-run the same analysis which you entered using the GUI. See the "history" tab in the "info" section when displaying the data. Chapter 12 provides more details on this.

33.2 Sensor space analysis

A useful feature of SPM is the ability to use Random Field Theory to correct for multiple statistical comparisons across N-dimensional spaces. For example, a 2D space representing the scalp data can be constructed by flattening the sensor locations and interpolating between them to create an image of MxM pixels (when M is user-specified, eg M=32). This would allow one to identify locations where, for example, the ERP amplitude in two conditions at a given timepoint differed reliably across subjects, having corrected for the multiple t-tests performed across pixels. That correction uses Random Field Theory, which takes into account the spatial correlation across pixels (i.e, that the tests are not independent). Here, we will consider a 3D example, where the third dimension is time, and test across trials within this single subject. We first create a 3D image for each trial of the two types, with dimensions M×M×S, where S=101 is the number of samples (time points). We then take these images into an unpaired t-test across trials (in a 2nd-level model) to compare "standard" and "rare" events. We can then use classical SPM to identify locations in space and time in which a reliable difference occurs, correcting across the multiple comparisons entailed. This would be appropriate if, for example, we had no a priori knowledge where or when the difference between standard and rare trials would emerge. The appropriate images are created as follows

- Select the "Convert to image" option from the "Other" pulldown menu.
- Select the `afdeMspm8_subject1.mat` file.

- For "output image dimensions" accept the default of 32 (leading to a 32x32 pixel space).
- For "interpolate' or "mask out" bad channels, select "interpolate".

SPM will take some time as it writes out a NIfTI image for each trial (except rejected trials), in a new directory called `afdeMspm8_subject1`, which will itself contain two subdirectories, one for each trialtype, called `type_rare` and `type_standard`. In each trialtype subdirectory there will be image and header files for each non-rejected trial of that type, e.g, trial0001.img/hdr. You can press "Display: images" to view one of these images - it will have dimensions $32 \times 32 \times 101$.

To perform statistics on these images:

- Create a new directory, eg. `mkdir XYTstats`.
- Press the "Specify 2nd level" button.
- Select "two-sample t-test" (unpaired t-test)
- Define the images for "Group 1" as all those in the subdirectory `type_standard` (using right mouse, and "select all") and the images for "Group 2" as all those in the subdirectory `type_rare`.
- Finally, specify the new `XYTstats` directory as the output directory.
- Press the "save" icon, top left, and save this design specification as `mmn_design.mat` and press "save".
- Press the green "Run" button to execute the job[3] This will produce the design matrix for a two-sample t-test.
- Now press "Estimate" in SPMs main window, and select the `SPM.mat` file from the `XYTstats` directory.

Now press "Results" and define a new F-contrast as [1 -1] (for help with these basic SPM functions, see eg. chapter 28). Keep the default contrast options, but threshold at $p < .05$ FWE corrected for the whole "image". Then press "volume", and the Graphics window should now look like that in Figure 33.3. This reveals a large fronto-central region within the 2D sensor space and within the time epoch in which standard and rare trials differ reliably, having corrected for multiple F-tests across pixels/time. An F-test is used because the sign of the difference reflects the polarity of the ERP difference, which is not of primary interest.

In the SPM results window, the time epoch (-100ms to 400ms) is referred in percentage units. The cursor in Figure 33.3 has been positioned by selecting the second cluster in the results table. This occurs at a time point 51% through the time epoch, which corresponds to 155ms post stimulus.

Now:

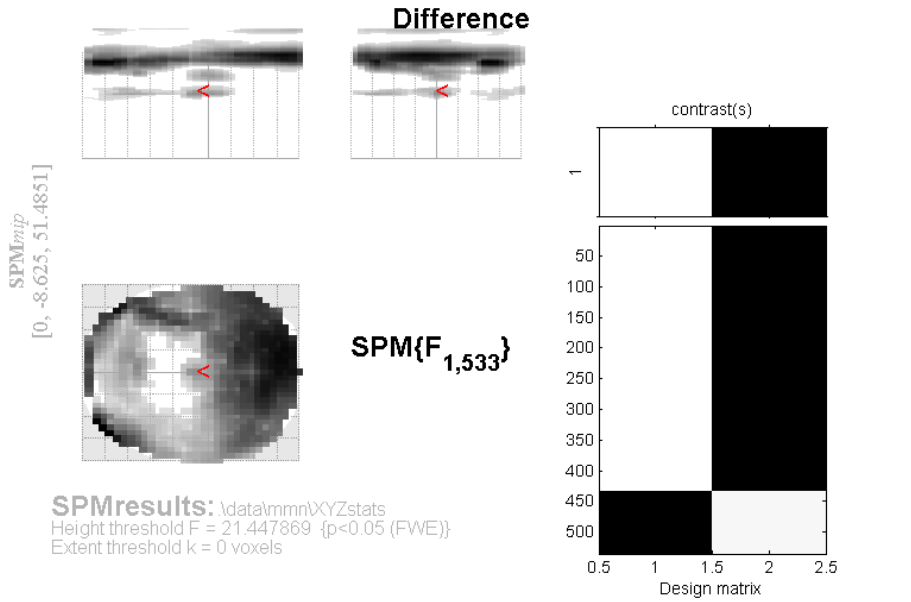
- Press the right mouse button in the MIP
- Select "display/hide" channels
- Select the `afdeMspm8_subject1.mat` file.

This links the `SPM.mat` file with the M/EEG file from which the EEG images were created. It is now possible to superimpose the channel labels onto the spatial SPM, and also to "goto the nearest channel" (using options provided after a right mouse click, when navigating the MIP).

We have demonstrated sensor space analysis for single-subject data. More usually, one would compute ERP images for each subject, smooth them, and then perform paired t-tests over subjects to look for condition differences. See [33] for a group analysis of MMN data.

Finally, if one had more constrained a priori knowledge about where and when the differences would appear, one could perform a Small Volume Correction (SVC) based on, for example, a box around fronto-central channels and between 100 and 200ms poststimulus. We also refer the reader to chapter 13 for further details on sensor space analysis.

[3] Note that we can use the default "nonsphericity" selections, i.e, that the two trial-types may have different variances, but are uncorrelated.



Statistics: *p-values adjusted for search volume*

set-level		cluster-level				peak-level					mm mm %		
p	*c*	$p_{FWE\text{-}corr}$	$q_{FDR\text{-}corr}$	k_E	p_{uncorr}	$p_{FWE\text{-}corr}$	$q_{FDR\text{-}corr}$	*F*	$(Z_{\equiv})$	p_{uncorr}			
0.000	**15**			**10349**		**0.000**	**0.000**	**93.46**	**Inf**	**0.000**	**43**	**-84**	**74**
						0.000	0.000	89.30	Inf	0.000	-9	61	80
						0.000	0.000	86.65	Inf	0.000	-47	-73	75
				198		**0.000**	**0.000**	**47.69**	**6.65**	**0.000**	**0**	**-9**	**51**
						0.000	0.000	44.24	6.41	0.000	4	2	51
				274		**0.000**	**0.000**	**43.51**	**6.36**	**0.000**	**0**	**2**	**64**
				432		**0.000**	**0.000**	**41.55**	**6.21**	**0.000**	**-51**	**-62**	**50**
						0.000	0.001	38.97	6.02	0.000	-38	-57	50
						0.000	0.001	38.65	5.99	0.000	-55	-52	50
				515		**0.000**	**0.003**	**35.71**	**5.76**	**0.000**	**-9**	**67**	**99**
						0.000	0.010	32.98	5.53	0.000	-34	61	99
						0.000	0.012	32.44	5.49	0.000	-26	45	99
				23		**0.000**	**0.011**	**32.72**	**5.51**	**0.000**	**38**	**-89**	**100**
						0.031	0.676	22.54	4.55	0.000	47	-79	96
				68		**0.001**	**0.032**	**30.08**	**5.28**	**0.000**	**68**	**-3**	**53**
				193		**0.002**	**0.044**	**29.25**	**5.21**	**0.000**	**55**	**-57**	**50**
						0.010	0.231	25.20	4.82	0.000	47	-79	47
						0.013	0.295	24.60	4.76	0.000	38	-62	50
				3		**0.008**	**0.183**	**25.82**	**4.88**	**0.000**	**-47**	**-73**	**100**

table shows 3 local maxima more than 8.0mm apart

Height threshold: F = 21.45, p = 0.000 (0.050)
Extent threshold: k = 0 voxels, p = 1.000 (0.050)
Expected voxels per cluster, <k> = 8.246
Expected number of clusters, <c> = 0.05
FWEp: 21.448, FDRp: 28.961

Degrees of freedom = [1.0, 533.0]
FWHM = 28.7 34.9 5.4 mm mm %; 6.7 6.5 5.4 {voxels}
Volume: 1886894 = 83426 voxels = 320.1 resels
Voxel size: 4.3 5.4 1.0 mm mm %; (resel = 238.37 voxels)
Page 1

Figure 33.3: *In this SPM the time axis is reflected in the two MIP windows in the top row, with time proceeding from the bottom to the top of the page. The cursor has been positioned by selecting the second cluster in the results table. This occurs at a time point 51% through the time epoch, which corresponds to 155ms post stimulus. The design matrix on the right hand side comprises two columns, the first for standard trials and the second for rare ones.*

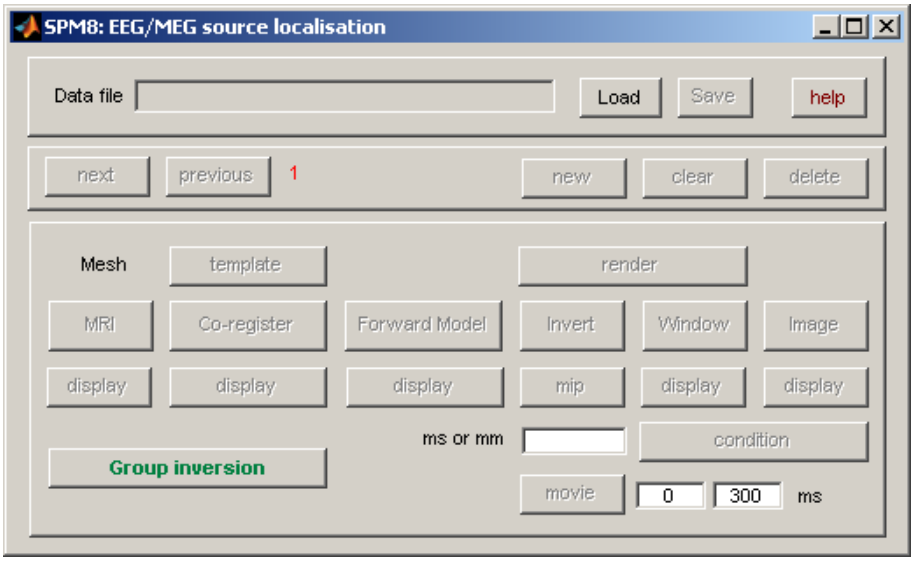


Figure 33.4: *Graphical user interface for 3D source localisation. A complete localisation comprises the following steps (i) creation of a cortical mesh, (ii) co-registration of the mesh with M/EEG data, (iii) creation of a forward model, and (iv) results interrogation. As each of these steps is completed the relevant part of the GUI becomes highlighted (text appears more solid).*

33.3 Source reconstruction

Source reconstruction comprises forward modeling and inverse modeling steps and is implemented by pressing the 3D source reconstruction button in SPM's top-left window. This brings up the source localisation GUI shown in Figure 33.4. The following subsections detail each of the steps in a source reconstruction analysis. We also advise the reader to consult the reference material in chapter 14.

33.3.1 Mesh

The first step is to load the data and create a cortical mesh upon which M/EEG data will be projected:

- Press the "Load" button in the souce localisation GUI and select the file `mafdeMspm8_subject1.mat`.
- Enter "Standard" under "Comment/Label for this analysis" and press OK.
- Now press the "template" button.
- For "Cortical mesh", select "normal".

SPM will then form the "standard" or "canonical" cortical mesh shown in the Graphics window which, after rotation, should look like Figure 33.5

33.3.2 Coregister

Now press the "Co-register" button. This will create further output in the Graphics window, the upper panel of which should like like Figure 33.6.

In this coregister step we were not required to enter any further parameters. However, if you are not using the template (or "canonical" mesh) or if at the "prepare" stage above you loaded your own (non-standard) sensor positions then you will be asked for the locations in MNI coordinates of the fiducial positions.

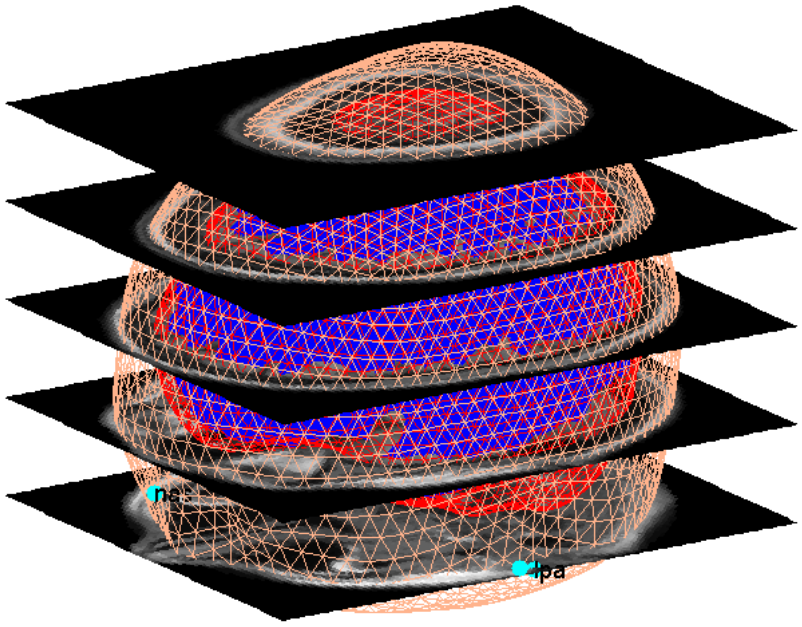


Figure 33.5: *The figure shows the canonical cortical mesh (blue), inner skull surface (red) and scalp surface (light brown). The hardwired fiducials are shown in light blue. Transverse slices of canonical MRI are also shown in black, with gray scale inlays showing anatomical detail.*

33.3.3 Forward model

Now press the "Forward model" button. Then select "EEG-BEM" in response to the "Which EEG head model?" question. SPM will then use a Boundary Element Method (BEM) which will take approximately 10 minutes to run. Upon completion SPM will write the `single_subj_T1_EEG_BEM.mat` file into the canonical subdirectory of your SPM distribution. The Graphics window should now appear as in Figure 33.7. The next time you wish to use an EEG-BEM solution based on the template mesh, SPM will simply use the date from the `single_subj_T1_EEG_BEM.mat` file (so this step will be much quicker the next time you do it). The same principle applies to EEG-BEM solutions computed from meshes based on subjects individual MRIs.

33.3.4 Invert

Now press the Invert button and

- Select an "Imaging" reconstruction.
- Select "Yes" for "All conditions or trials".
- Select "Standard" for Model.

SPM will now compute a leadfield matrix and save it in the file `SPMgainmatrix.mat` placed in `DATA_DIR`. This file can be replaced with one computed using other methods for computing the lead field (eg methods external to SPM). The forward model will then be inverted using the Multiple Sparse Priors (MSP) algorithm (the progress of which is outputted to the MATLAB command window). SPM will produce, in the Graphics window, (i) a Maximum Intensity Projection (MIP) of activity in source space (lower panel) and (ii) a time series of activity for (upper panel) each condition.

The "ms or mm" window has three functionalities (i) if you enter a single number this will be interpreted as ms, (ii) if you enter two numbers this will be interpreted as a time window

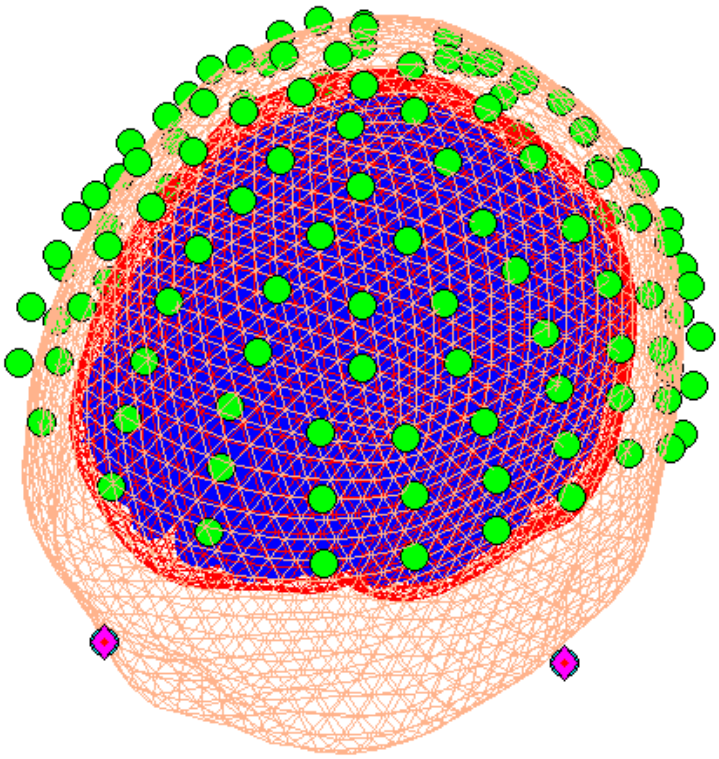


Figure 33.6: *The figure shows the MRI fiducials (pink), the sensor fiducials (blue) and the locations of sensors (green) in addition the the canonical cortical mesh (blue), inner skull surface (red) and scalp surface (light brown).*

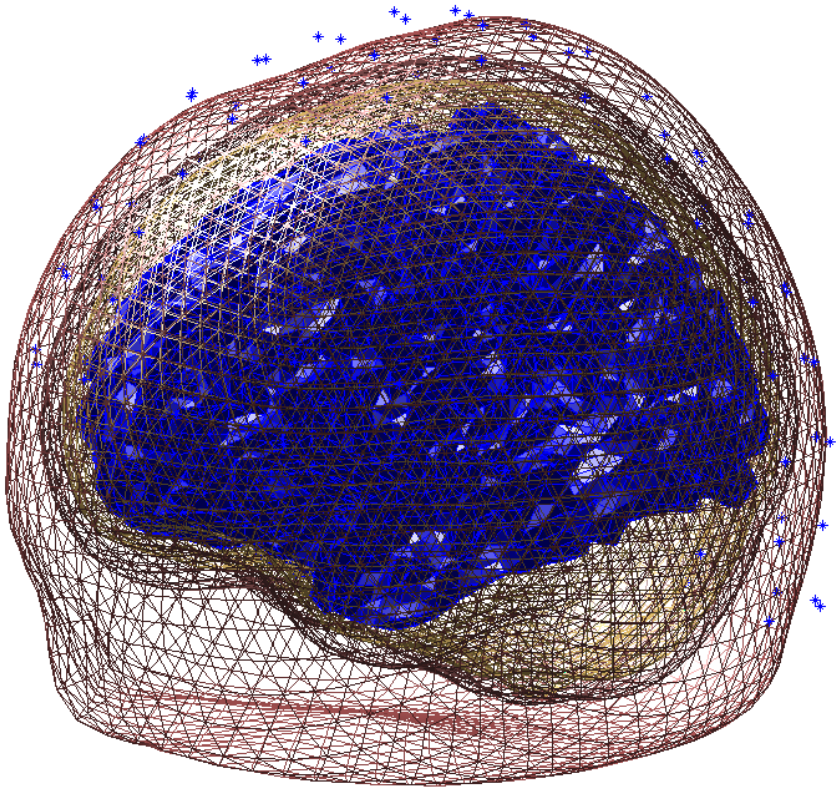


Figure 33.7: *The figure shows the cortical mesh (blue), brain, skull and scalp surfaces. Electrode positions are marked with asterisks.*

for plotting movies (see below), (iii) if you enter 3 numbers this will be interpreted as MNI coordinates for a time series plot.

Now enter "150" for "ms or mm" and press the MIP button, to see a MIP of activity in source space at 150ms post-stimulus, and the time series of activities (top panel) at the position with largest magnitude signal. The corresponding graphic is shown in Figure 33.8. By toggling the "Condition" button, and pressing MIP each time, you can view the spatial distribution of activity for the different conditions (at the selected time point).

33.4 Dynamic Causal Modeling

Many of the functionalities of DCM for M/EEG are described in more detail in the reference chapter 16. In this chapter we demonstrate only the "DCM for ERP" model. Users are strongly encouraged to read the accompanying theoretical papers [20, 48]. Briefly, DCM for ERP fits a neural network model to M/EEG data, in which activity in source regions are described using differential equations based on neural mass models. Activity in each region comprises three populations of cells; pyramidal, local excitatory and local inhibitory. Fitting the model will then allow you to plot estimated activity in each cell population in each region. It will also provide estimates of the long range connections between regions, and show how these values are changed by experimental manipulation (eg. rare versus standard trial types).

In the `example_scripts` folder of the SPM distribution, we also provide an example script that will run a DCM-for-ERP analysis of this data. This can be edited to implement your own analysis.

Pressing the "DCM" button will open up the DCM GUI shown in Figure 33.9. We will now complete the three model specification entries shown in Figure 33.10:

- Press the "new data" button and select the `mafdeMspm8_subject1.mat` file.
- Enter the "between-trial effects" and design matrix information shown in Figure 33.10(a).
- Press the "Display" button.

This completes the data specification stage. Now:

- Press the right hand arrow to move on to the specification of the electromagnetic model.
- Instead of "IMG" select "ECD" for the spatial characteristics of the sources.
- Now enter the names and (prior mean) locations of the sources shown in Figure 33.10(b).
- Pressing the "dipoles" button will create an interactive display in the graphics window showing the prior source positions.

This completes the specification of the electromagnetic model. Now:

- Press the right hand arrow (next to the dipoles button) to move on to specification of the neuronal model.
- Highlight the connectivity matrix radio buttons so that they correspond to those shown in Figure 33.10(c).
- Press the (top left) 'save' button and accept the default file name.
- Press 'Invert DCM'

SPM will plot the progess of the model estimation in the MATLAB command window. Plots of data and the progressing model fit will be shown in SPM's graphics window. The algorithm should converge after five to ten minutes (in 51 iterations). Now select the "ERPs (sources)" option from the pull down menu to the right of the "Estimated" button. This will produce the plot shown in Figure 33.11. The values of the connections between areas can be outputted by selecting eg "Coupling(A)" from the pull-down menu in the DCM GUI. This will allow you to interrogate the posterior distribution of model parameters. It is also possible to fit multiple models, eg. with different numbers of regions and different structures, and to compare them using Bayesian Model Comparison. This is implemented by pressing the BMS button (bottom right hand corner of the DCM window).

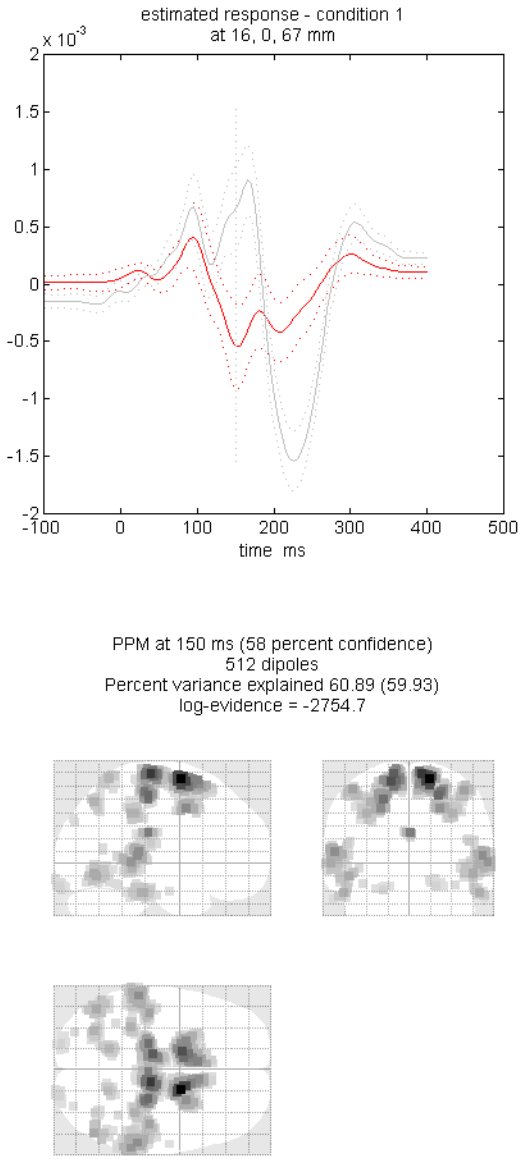


Figure 33.8: *Source reconstructed activity at 150ms post-stimulus. The upper trace shows responses to Condition 1 (Standards) with the red curve, and to Condition 2 (Rare) in gray.*

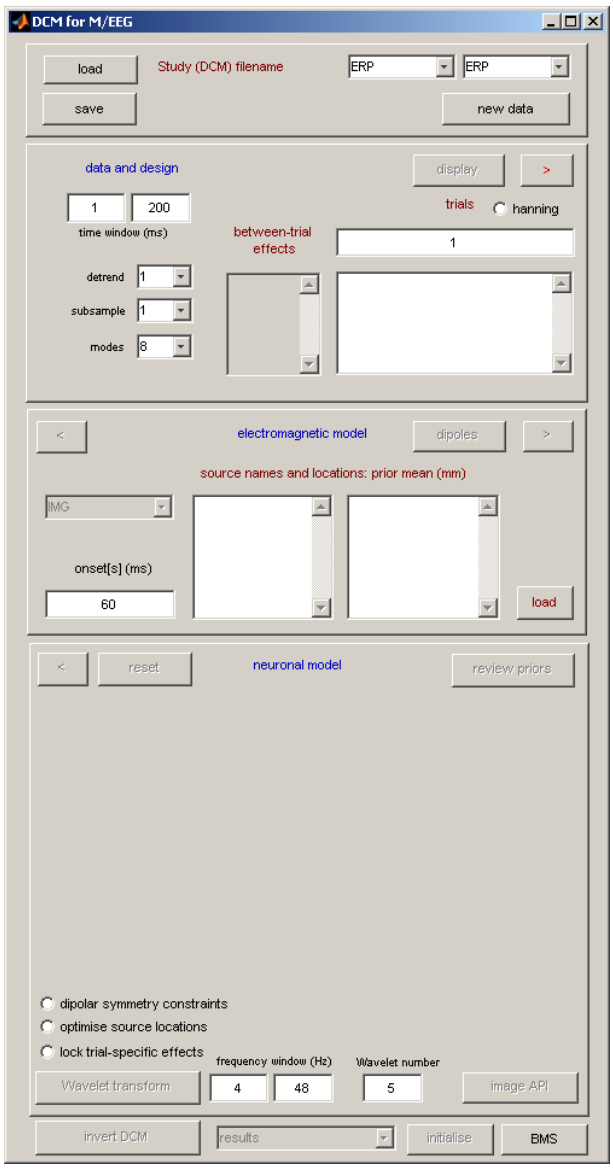


Figure 33.9: *The Dynamic Causal Modeling GUI splits model specification into three reversible phases (i) data and design, (ii) electromagnetic model and (iii) neuronal model. One can move forwards and backwards in the model specification using the left and right arrow buttons (these become highlighted when sufficient information has been entered to proceed to the next step).*

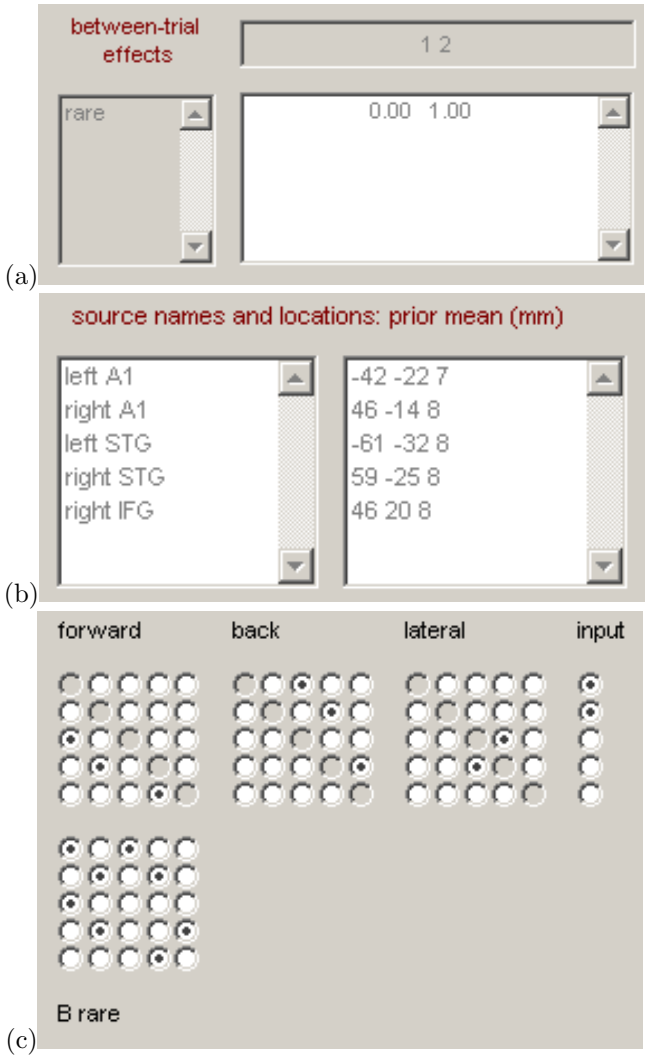


Figure 33.10: *Specification of DCM for ERP model (a) Data and design, (b) electromagnetic model and (c) neuronal model.*

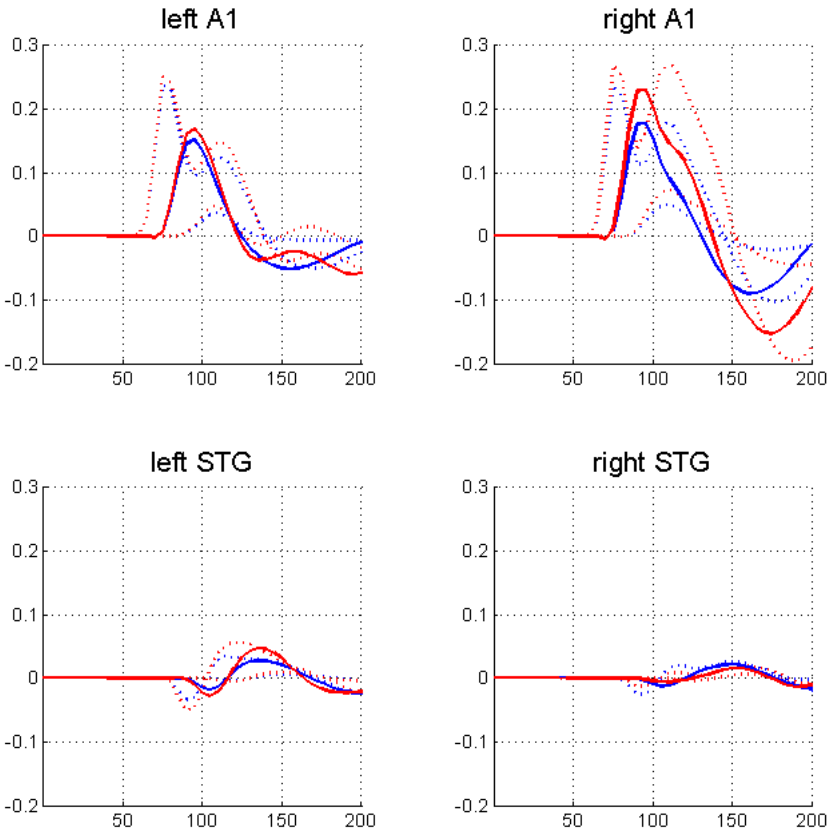


Figure 33.11: *Activity plots for three neuronal populations (solid lines for pyramidal cells, dotted lines for others), in four areas (fifth not shown in this figure), for standard (blue) and rare (red) trial types.*

Chapter 34

Using DARTEL

DARTEL[1] is a suite of tools for achieving more accurate inter-subject registration of brain images. It consists of several thousand lines of code. Because it would be a shame if this effort was wasted, this guide was written to help encourage its widespread use. Experience at the FIL would suggest that it offers very definite improvements for VBM studies – both in terms of localisation[2] and increased sensitivity[3].

34.1 Using DARTEL for VBM

The following procedures could be specified one at a time, but it is easier to use the batching system. The sequence of jobs (use the *TASKS* pull-down from the *Graphics* window to select *BATCH*) would be:

- **Module List**
  - **SPM→Spatial→Segment**: To obtain *_seg_sn.mat files for "importing" the data into a form that DARTEL can use for registering the subject's scans.
  - **SPM→Tools→DARTEL Tools→Initial Import**: Uses the *_seg_sn.mat files to generate roughly (via a rigid-body) aligned grey and white matter images of the subjects.
  - **SPM→Tools→DARTEL Tools→Run DARTEL (create Template)**: Determine the nonlinear deformations for warping all the grey and white matter images so that they match each other.
  - **SPM→Tools→DARTEL Tools→Normalise to MNI Space**: Actually generate the smoothed "modulated" warped grey and white matter images.

Alternatively, the **New Segment** procedure could be used. Although still at the slightly experimental stages of development, this procedure has been found to be generally more robust than the implementation of "Unified Segmentation" from SPM→Spatial→Segment (which is the version from the Segment button - the same as that in SPM5). The new segmentation procedure includes the option to generate DARTEL "imported" data, so the **Initial Import** step is skipped.

- **Module List**
  - **SPM→Tools→New Segment**: To generate the roughly (via a rigid-body) aligned grey and white matter images of the subjects.
  - **SPM→Tools→DARTEL Tools→Run DARTEL (create Template)**: Determine the nonlinear deformations for warping all the grey and white matter images so that they match each other.

[1] DARTEL stands for "Diffeomorphic Anatomical Registration Through Exponentiated Lie algebra". It may not use a true Lie Algebra, but the acronym is a nice one.

[2] Less smoothing is needed, and there are fewer problems relating to how to interpret the differences.

[3] More sensitivity could mean that fewer subjects are needed, which should save shed-loads of time and money.

  - **SPM→Tools→DARTEL Tools→Normalise to MNI Space**: Actually generate the smoothed "modulated" warped grey and white matter images.

The segmentation and importing steps of these two alternative processing streams are described next.

34.1.1 Using Spatial→Segment and DARTEL Tools→Initial Import

The first step is to classify T1-weighted scans[4] of a number of subjects into different tissue types via the Segmentation routine in SPM. The *SPM→Spatial→Segment* pull-down can be used here:

- **Segment**
  - **Data**: Select all the T1-weighted images, one per subject. It is usually a good idea to have roughly aligned them to MNI space first. The *Display* button can be used to reorient the data so that the *mm* coordinate of the AC is within about 3cm from $[0, 0, 0]$, and the orientation is within about 15^o of MNI space. The *Check Reg* button can be used to see how well aligned a number of images are.
  - **Output Files**: It is suggested that *Native Space* grey (and possibly white) matter images are created. These are c1*.img and c2*.img. The Segmentation produces a *_seg_sn.mat and a *_seg_inv_sn.mat for each image. It is the *_seg_sn.mat files that are needed for the next step.
  - **Custom**: Default settings can usually be used here.

The resulting *_seg_sn.mat files encode various parameters that allow the data to be "imported" into a form that can be used by the main DARTEL algorithm. In particular, *Procrustes* aligned maps of grey and white matter can be generated. Select *SPM→Tools→DARTEL Tools→Initial Import*:

- **Initial Import**
  - **Parameter Files**: Select all the *_seg_sn.mat files generated by the previous step. The T1-weighted scans need not be selected, as the import routine will try to find them. If the image files have not been moved since the segmentation, then their location can be determined by the contents of the *_seg_sn.mat files. If they have been moved, then the routine looks for the files in the current directory, or the output directory.
  - **Output Directory**: Specify where the imported data should be written.
  - **Bounding box**: This is the bounding box for the imported data. If the values are not finite (eg, if they are $[NaN, NaN, NaN; NaN, NaN, NaN]$) then the bounding box for the tissue probability maps, used as priors for the segmentation, will be assumed. Note that the deformations that DARTEL estimates will wrap around at the boundaries, so it is usually a good idea to ensure that the whole brain is easily enclosed within the bounding box.
  - **Voxel size**: These specify the resolution of the imported data. $[1.5, 1.5, 1.5]$ are reasonable values here. If the resolution is finer than this, then you may encounter memory problems during the actual DARTEL registration. If you do want to try working at a higher resolution, then consider changing the bounding box (but allow for the strange behaviour at the edges).
  - **Image option**: No imported version of the image is needed - usually only the grey and white matter tissue classes are used (ie choose *None*).
  - **Grey Matter**: Yes, you need this.
  - **White Matter**: Yes, you also need this.
  - **CSF**: The CSF is not usually segmented very reliably because the segmentation only has tissue probability maps for GM WM an CSF. Because there are no maps for bone and other non-brain tissue, it is difficult for the segmentation algorithm to achieve a good CSF segmentation. Because of the poor CSF segmentation, it is not a good idea to use this tissue class for the subsequent DARTEL registration.

[4] Other types of scan may also work, but this would need some empirical exploration.

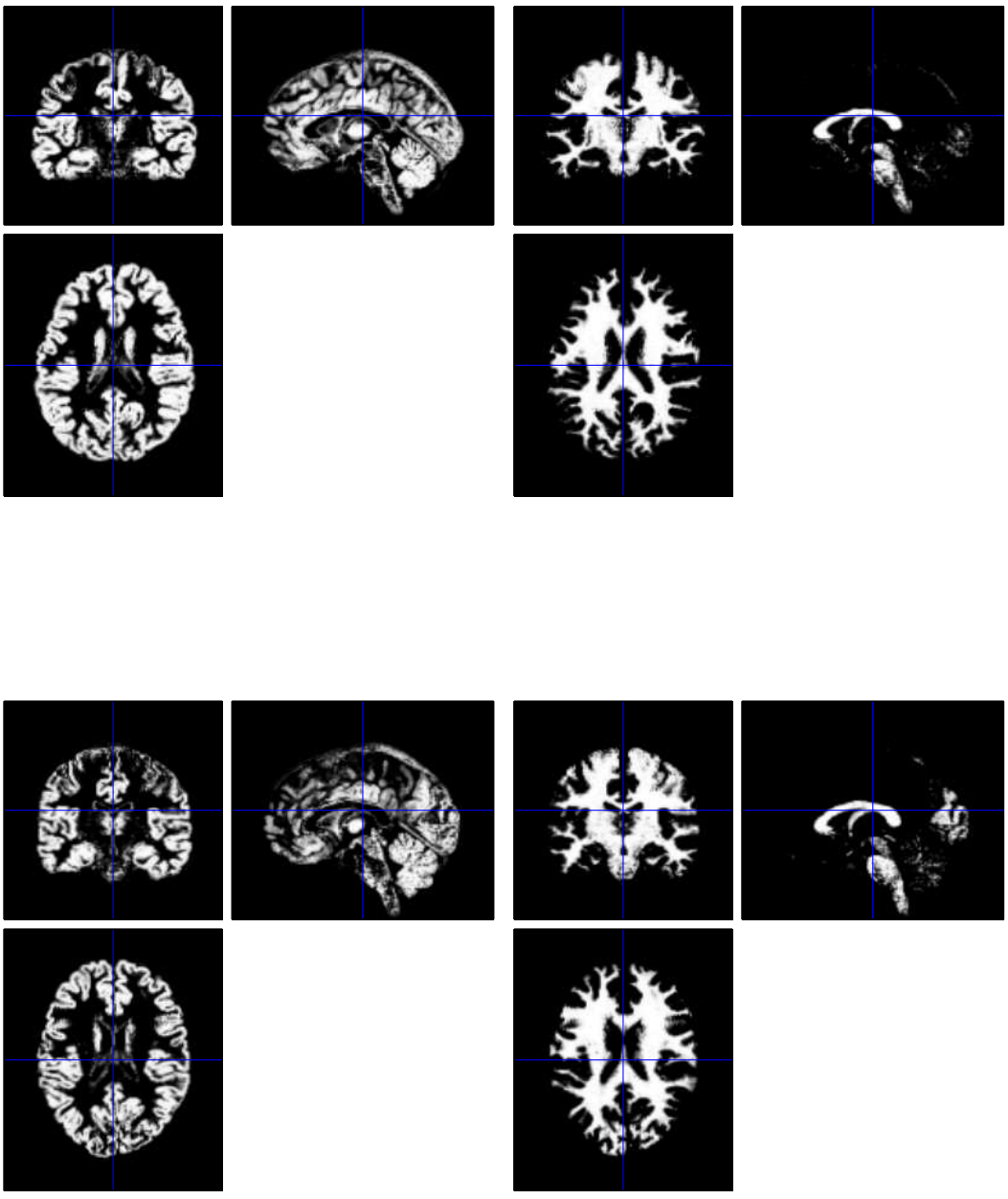


Figure 34.1: Imported data for two subjects (A and B). Top row: rc1A.nii and rc2A.nii. Bottom row: rc1B.nii and rc2B.nii.

34.1.2 Using Tools→New Segment

Note: This subsection will be elaborated on later.

There is a new segmentation option in SPM8, which can be found within Tools→New Segment. If this option is used, then the "imported" tissue class images (usually rc1.nii and rc2.nii) would be generated directly and the initial import step is skipped. It is also suggested that *Native Space* versions of the tissues in which you are interested are also generated. For VBM, these are usually the c1*.nii files, as it is these images that will eventually be warped to MNI space. Both the imported and native tissue class image sets can be specified via the Native Space options of the user interface.

34.1.3 Using DARTEL Tools→Run DARTEL (create Template)

The output of the previous step(s) are a series of rigidly aligned tissue class images (grey matter is typically encoded by rc1*.nii and white matter by rc2*.nii – see Fig 34.1). The headers of these files encode two affine transform matrices, so the DARTEL tools are still able to relate their orientations to those of the original T1-weighted images. The next step is to estimate the nonlinear deformations that best align them all together. This is achieved by alternating between building a template, and registering the tissue class images with the template, and the whole procedure is very time consuming. Specify *SPM→Tools→Dartel Tools→Run DARTEL (create Template)*.

- **Run DARTEL (create Template)**
  - **Images**
    - **Images**: Select all the rc1*.nii files generated by the import step.
    - **Images**: Select all the rc2*.nii files, in the same subject order as the rc1*.nii files. The first rc1*.nii is assumed to correspond with the first rc2*.nii, the second with the second, and so on.
  - **Settings**: Default settings generally work well, although you could try changing them to see what happens. A series of templates are generated called Template_basename_0.nii, Template_basename_1.nii etc. If you run multiple DARTEL sessions, then it may be a good idea to have a unique template basename for each.

The procedure begins by computing an initial template from all the imported data. If u_rc1*.nii files exist for the images, then these are treated as starting estimates and used during the creation of the initial template. If any u_rc1*.nii files exist from previous attempts, then it is usually recommended that they are removed first (this sets all the starting estimates to zero). Template generation incorporates a smoothing procedure, which may take a while (several minutes). Once the original template has been generated, the algorithm will perform the first iteration of the registration on each of the subjects in turn. After the first round of registration, a new template is generated (incorporating the smoothing step), and the second round of registration begins. Note that the earlier iterations usually run faster than the later ones, because fewer "time-steps" are used to generate the deformations. The whole procedure takes (in the order of) about a week of processing time for 400 subjects.

The end result is a series of templates (see Fig 34.2), and a series of u_rc1*.nii files. The first template is based on the average[5] of the original imported data, where as the last is the average of the DARTEL registered data. The u_rc1*.nii files are flow fields that parameterise the deformations. Note that all the output usually contains multiple volumes per file. For the u_rc1*.nii files, only the first volume is visible using the Display or Check Reg tools in SPM. All volumes within the template images can be seen, but this requires the file selection to be changed to give the option of selecting more than just the first volume (in the file selector, the widget that says "1" should be changed to "1:2").

[5]They are actually more similar to weighted averages, where the weights are derived from the Jacobian determinants of the deformations. There is a further complication in that a smoothing procedure is built into the averaging.

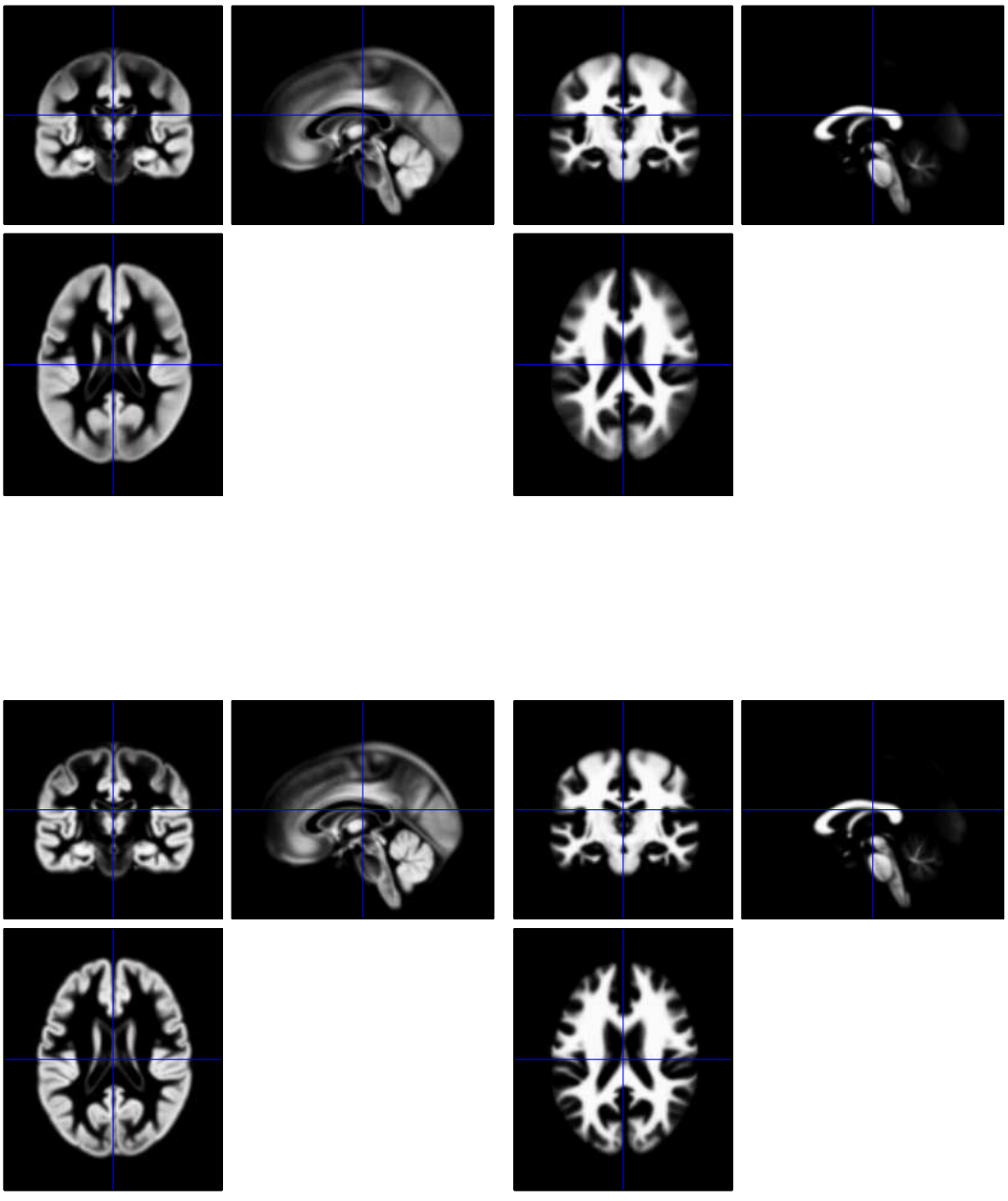


Figure 34.2: Different stages of template generation. Top row: an intermediate version of the template. Bottom row: the final template data.

34.1.4 Using DARTEL Tools→Normalise to MNI Space

The next step is to create the Jacobian scaled ("modulated") warped tissue class images, by selecting *SPM→Tools→DARTEL Tools→Normalise to MNI Space*. The option for spatially normalising to MNI space automatically incorporates an affine transform that maps from the population average (DARTEL Template space) to MNI space, as well as incorporating a spatial smoothing step.

- **Normalise to MNI Space**
  - **DARTEL Template**: Specify the last of the series of templates that was created by *Run DARTEL (create Template)*. This is usually called *Template_6.nii*. Note that the order of the N volumes in this template should match the order of the first N volumes of the *toolbox/DARTEL/TPM.nii* file.
  - **Select according to** either *Few Subjects* or *Many Subjects*. For VBM, the *Many Subjects* option would be selected.
    - **Flow Fields**: Specify the flow fields (u_rc1*.nii) generated by the nonlinear registration.
    - **Images**: You may add several different sets of images.
      - **Images**: Select the c1*.nii files for each subject, in the same order as the flow fields are selected.
      - **Images**: This is optional, but warped white matter images can also be generated by selecting the c2*.nii files.
  - **Voxel sizes**: Specify the desired voxel sizes for the spatially normalised images (NaN, NaN, NaN gives the same voxel sizes as the DARTEL template).
  - **Bounding box**: Specify the desired bounding box for the spatially normalised images (NaN, NaN, NaN; NaN NaN NaN gives the same bounding box as the DARTEL template).
  - **Preserve**: Here you have a choice of *Preserve Concentrations* (ie not Jacobian scaled) or *Preserve Amount* (Jacobian scaled). The *Preserve Amount* would be used for VBM, as it does something similar to Jacobian scaling (modulation).
  - **Gaussian FWHM**: Enter how much to blur the spatially normalised images, where the values denote the full width at half maximum of a Gaussian convolution kernel, in units of mm. Because the inter-subject registration should be more accurate than when done using other SPM tools, the FWHM can be smaller than would be otherwise used. A value of around 8mm (ie $[8, 8, 8]$) should be about right for VBM studies, although some empirical exploration may be needed. If there are fewer subjects in a study, then it may be advisable to smooth more.

The end result should be a bunch of smwc1*.nii files[6] (possibly with smwc2*.nii if white matter is also to be studied).

The final step is to perform the statistical analysis on the preprocessed data (smwc1*.nii files), which should be in MNI space. The next section says a little about how data from a small number of subjects could be warped to MNI space.

34.2 Spatially normalising functional data to MNI space

Providing it is possible to achieve good alignment between functional data from a particular subject and an anatomical image of the same subject (distortions in the fMRI may prevent

[6]The actual warping of the images is done slightly differently, with the aim that as much of the original signal is preserved as possible. This essentially involves pushing each voxel from its position in the original image, into the appropriate location in the new image - keeping a count of the number of voxels pushed into each new position. The procedure is to scan through the original image, and push each voxel in turn. The alternative (older way) was to scan through the spatially normalised image, filling in values from the original image (pulling the values from the original). The results of the pushing procedure are analogous to Jacobian scaled ("modulated") data. A minor disadvantage of this approach is that it can introduce aliasing artifacts (think stripy shirt on TV screen) if the original image is at a similar - or lower - resolution to the warped version. Usually, these effects are masked by the smoothing.

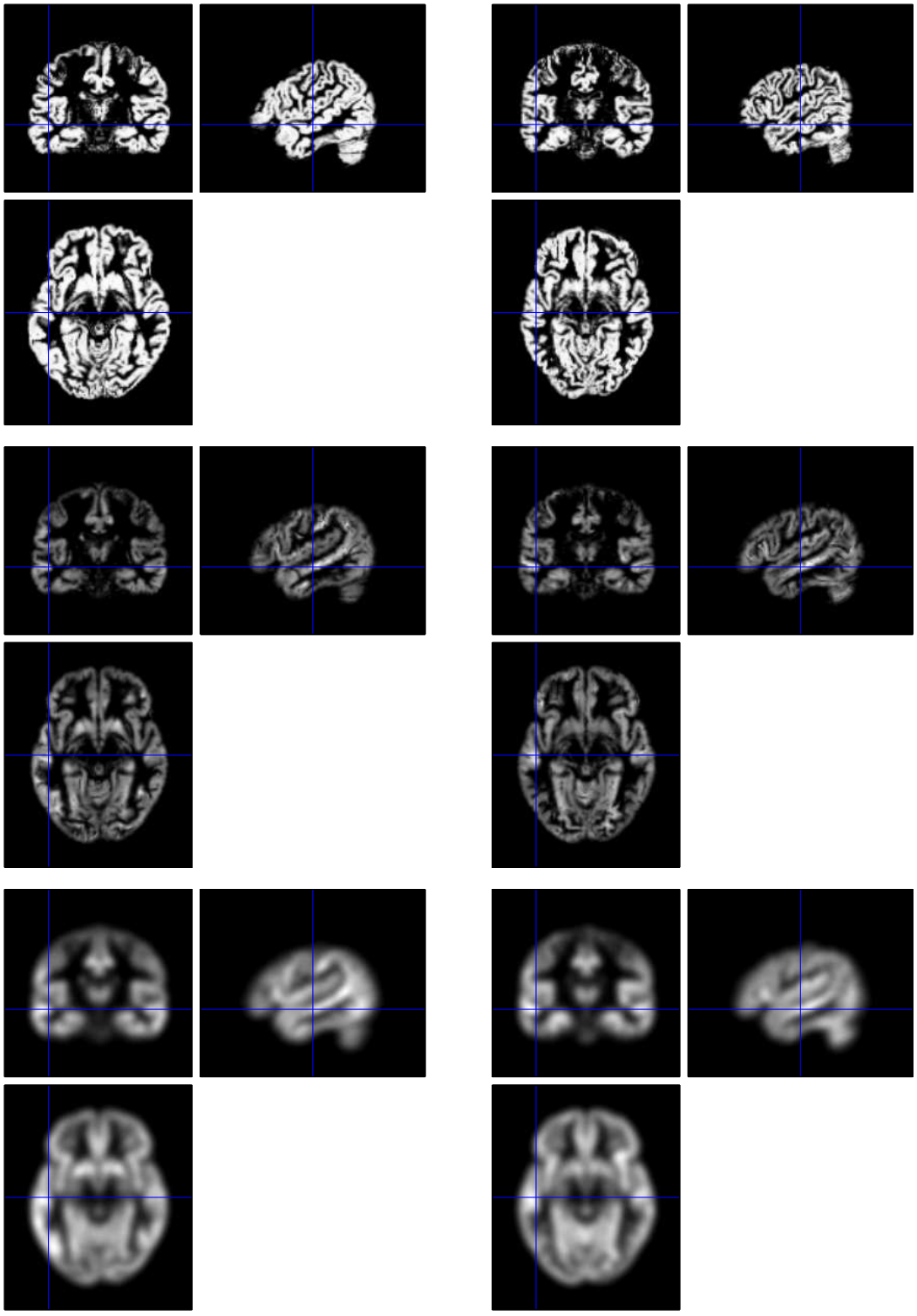


Figure 34.3: Pre-processing for VBM. Top row: Imported grey matter (rc1A.nii and rc1B.nii). Centre row: Warped with *Preserve Amount* option and zero smoothing ("modulated"). Bottom row: Warped with *Preserve Amount* option smoothing of 8mm (smwc1A.nii and smwc1B.nii).

accurate alignment), then it may be possible to achieve more accurate spatial normalisation of the fMRI data using DARTEL. There are several advantages of having more accurate spatial normalisation, especially in terms of achieving more significant activations and better localisation.

The objectives of spatial normalisation are:

- To transform scans of subjects into alignment with each other. DARTEL was developed to achieve better inter-subject alignment of data.
- To transform them to a standard anatomical space, so that activations can be reported within a standardised coordinate system. Extra steps are needed to achieve this aim.

The option for spatially normalising to MNI space automatically incorporates an affine transform that maps from the population average (DARTEL Template space) to MNI space. This transform is estimated by minimising the KL divergence between the final template image generated by DARTEL and tissue probability maps that are released as part of SPM (in the new segmentation toolbox). MNI space is defined according to affine matched images, so an affine transform of the DARTEL template to MNI space would appear to be a reasonable strategy.

For GLM analyses, we usually do not wish to work with Jacobian scaled data. For this reason, warping is now combined with smoothing, in a way that may be a bit more sensible than simply warping, followed by smoothing. The end result is essentially the same as that obtained by doing the following with the old way of warping

- Create spatially normalised and "modulated" (Jacobian scaled) functional data, and smooth.
- Create spatially normalised maps of Jacobian determinants, and smooth by the same amount.
- Divide one by the other, adding a small constant term to the denominator to prevent divisions by zero.

This should mean that signal is averaged in such a way that as little as possible is lost. It also assumes that the procedure does not have any nasty side effects for the GRF assumptions used for FWE corrections.

Prior to spatially normalising using DARTEL, the data should be processed as following:

- If possible, for each subject, use *SPM→Tools→FieldMap* to derive a distortion field that can be used for correcting the fMRI data. More accurate within-subject alignment between functional and anatomical scans should allow more of the benefits of DARTEL for inter-subject registration to be achieved.
- Use either *SPM→Spatial→Realign→Realign: Estimate Reslice* or *SPM→Spatial→Realign Unwarp*. If a field map is available, then use the *Realign Unwarp* option. The images need to have been realigned and resliced (or field-map distortion corrected) beforehand - otherwise things are not handled so well. The first reason for this is that there are no options to use different methods of interpolation, so rigid-body transforms (as estimated by Realign but without having resliced the images) may not be well modelled. Similarly, the spatial transforms do not incorporate any masking to reduce artifacts at the edge of the field of view.
- For each subject, register the anatomical scan with the functional data (using *SPM → Spatial → Coreg → Coreg: Estimate*). No reslicing of the anatomical image is needed. Use *SPM→Util→Check Registration* to assess the accuracy of the alignment. If this step is unsuccessful, then some pre-processing of the anatomical scan may be needed in order to skull-strip and bias correct it. Skull stripping can be achieved by segmenting the anatomical scan, and masking a bias corrected version (which can be generated by the segmentation option) by the estimated GM, WM and CSF. This masking can be done using *SPM→Util→Image Calculator* (*ImCalc* button), by selecting the bias corrected scan (m*.img), and the tissue class images (c1*.img, c2*.img and c3*.img) and evaluating "i1. × ((i2+i3+i4)>0.5)". If segmentation is done before coregistration, then the functional data should be moved so that they align with the anatomical data.

- Segment the anatomical data and generate "imported" grey and white matter images. If *SPM→Tools→New Segment* is used, then make sure that "imported" grey and white matter images are created. If *SPM→Spatial→Segment* (the SPM5 segmentation routine, which is the one under the *Segment* button), then an additional *SPM→Tools→DARTEL Tools→Initial Import* will be needed.
- To actually estimate the warps, use *SPM→Tools→DARTEL Tools→Run DARTEL (create Templates)* in order to generate a series of templates and a flow field for each subject.

In principle (for a random effects model), you could run the first level analysis using the native space data of each subject. All you need are the contrast images, which can be warped and smoothed. Alternatively, you could warp and smooth the reslices fMRI, and do the statistical analysis on the spatially normalised images. Either way, you would select *SPM→Tools→DARTEL Tools→Normalise to MNI Space*:

- **Normalise to MNI Space**
  - **DARTEL Template**: Template_6.nii,1 is usually the grey matter component of the final template of the series. An affine transform is determined using this image.
  - **Select according to** either *Few Subjects* or *Many Subjects*. For fMRI analyses, the *Few Subjects* option would be selected, which gives the option of selecting a flow field and list of images for each subject.
    - **Subject**
      - **Flow Field**: Specify the flow field ("u_c1*.nii") for this subject.
      - **Images**: Select the images for this subject that are to be transformed to MNI space.
  - **Voxel sizes**: Specify the desired voxel sizes for the spatially normalised images (NaN, NaN, NaN gives the same voxel sizes as the DARTEL template).
  - **Bounding box**: Specify the desired bounding box for the spatially normalised images (NaN, NaN, NaN; NaN NaN NaN gives the same bounding box as the DARTEL template).
  - **Preserve**: Here you have a choice of *Preserve Concentrations* (ie not Jacobian scaled) or *Preserve Amount* (Jacobian scaled). The *Preserve Concentrations* option would normally be used for fMRI data, whereas *Preserve Amount* would be used for VBM.
  - **Gaussian FWHM**: Enter how much to blur the spatially normalised images, where the values denote the full width at half maximum of a Gaussian convolution kernel, in units of mm.

An alternative approach is now presented, which does not attempt to make optimal use of the available signal.

34.2.1 An alternative approach for using DARTEL to spatially normalise to MNI Space

During spatial normalisation of a brain image, some regions need to expanded and other regions need to contract in order to match the template. If some structure is excessively shrunk by DARTEL (because it has the freedom to estimate quite large deformations), then this will lead to a systematic reduction in the amount of BOLD signal being detected from that brain region. For this reason, the normalise to MNI space option would generally be preferred when working with functional data that is to be smoothed.

Affine transform of DARTEL template to MNI space

DARTEL works with images that are of average size. When DARTEL is used to generate an average shaped template (represented by a series of tissue probability maps) from a group of scans of various individuals, the result is of average size. Brains normalised to MNI space are slightly larger than average. In order to spatially normalise to MNI space, the deformation that

maps from MNI space to the space of the group average is required. Because the MNI space was derived by affine registration of a number of subjects to a common coordinate system, in most cases it should be possible to achieve a reasonable match of the template generated by DARTEL using only an affine spatial normalisation. This can be achieved by matching the grey matter component of the template with a grey matter tissue probability map in MNI space. The spatial normalisation routine in SPM can be used to achieve this.

- **Normalise: Estimate**
  - **Data**
    - **Subject**
      - **Source Image**: Template_6.nii,1 is usually the grey matter component of the final template of the series.
      - **Source Weighting Image**: <None>
  - **Estimation Options**
    - **Template Image**: Should be the apriori/grey.nii file distributed in SPM.
    - **Template Weighting Image**: <None>
    - **Source Image Smoothing**: 8mm (the same as the apriori/grey.nii file has been smoothed).
    - **Template Image Smoothing**: 0mm (because the data in the apriori folder are already smoothed by 8mm.)
    - **Affine Regularisation**: Usually, you would specify "ICBM space template".
    - **Nonlinear Frequency Cutoff**: Set this to infinity (enter "Inf") for affine registration.
    - **Nonlinear Iterations**: Setting this to zero will also result in affine-only spatial normalisation.
    - **Nonlinear Regularisation**: Setting this to infinity is another way of doing affine-only spatial normalisation.

For some populations of subjects, an affine transform may not be adequate for achieving good registration of the average shape to MNI space. Nonlinear spatial normalisation may be more appropriate for these cases. As ever, determining which procedure is better would involve a degree of empirical exploration.

Combining deformations

Once you have the spatial transformation that maps from MNI space to the space of the DARTEL template, it is possible to combine this with the DEFORMATIONS estimated by DARTEL. Rather than warping the image data twice (introducing interpolation artifacts each time), the two spatial transforms can be combined by composing them together. The required deformation, for spatially normalising an individual to MNI space, is a mapping from MNI space to the individual image. This is because the spatially normalised images are generated by scanning through the (initially empty) voxels in the spatially normalised image, and figuring out which voxels in the original image to sample from (as opposed to scanning through the original image and putting the values into the right places in the spatially normalised version).

The desired mapping is from MNI space to DARTEL template to individual scan. If A is the mapping from MNI to template, and B is the mapping from template to individual, then this mapping is $B \circ A$, where "$\circ$" denotes the composition operation. Spatially normalising via the composed deformations can be achieved through the *Deformations* utility from the *TASKS* pull-down (it is in *Utils*).

- **Deformations**
  - **Composition**
    - **DARTEL flow**
      - **Flow field**: Specify the u_rc1*.nii flow field for that subject.

· **Forward/Backwards**: This should be set to "Backward" to indicate a mapping from template to individual.
· **Time Steps**: This is the number of time steps used by the final iterations of the DARTEL registration (usually 64).

* **Imported _sn.mat**
· **Parameter File**: Select the spatial normalisation parameters that would spatially normalise the Template_6.nii file.
· **Voxel sizes**: These are set to "NaN" (not a number) by default, which would take the voxel sizes for the apriori/grey.nii file. Alternatively, you could specify your favourite voxel sizes for spatially normalised images.
· **Bounding box**: Again, these are set to non-finite values by default, which results in the same bounding box as the apriori/grey.nii file. To specify your favourite bounding box, enter $[x_{min}, y_{min}, z_{min}; x_{max}, y_{max}, z_{max}]$ (in units of mm, relative to the AC).

– **Save as**: You can save the composed deformations as a file. This would be called y_*.nii, which contains three volumes that encode the x, y and z components of the mapping. Note that only the first (x) component can be visualised in SPM. These things were not really designed to be visualised as images anyway.
– **Apply to**: Specify the images for that subject that you would like spatially normalised. Note that the spatially normalised images are not masked (see the Chapter on Realignment for more information here). If realignment parameters are to be incorporated into the transformation, then this could cause problems at the edges. These can be avoided by reslicing after realignment (which is the default option if you "Realign Unwarp"). Alternatively, some form of additional masking could be applied to the spatially normalised images, prior to smoothing.
– **Interpolation**: Specify the form of interpolation.

The above procedure would be repeated for each subject in the study.

34.3 Warping Images to Existing Templates

If templates have already been created using DARTEL, then it is possible to align other images with such templates. The images would first be imported in order to generate rc1*.nii and rc2*.nii files. The procedure is relatively straight-forward, and requires the *SPM→Tools→DARTEL Tools→Run DARTEL (existing Template)* option to be specified. Generally, the procedure would begin by registering with a smoother template, and end with a sharper one, with various intermediate templates between.

• **Run DARTEL (existing Templates)**
– **Images**
* **Images**: Select the rc1*.nii files.
* **Images**: Select the corresponding rc2*.nii files.
– **Settings**: Most settings would be kept at the default values, except for the specification of the templates. These are specified in within each of the *Settings→Outer Iterations→Outer Iteration→Template* fields. If the templates are Template_*.nii, then enter them in the order of Template_1.nii, Template_2.nii, ... Template_6.nii.

Running this option is rather faster than *Run DARTEL (create Template)*, as templates are not created. The output is in the form of a series of flow fields (u_rc1*.nii).

34.4 Warping one individual to match another

Sometimes the aim is to deform an image of one subject to match the shape of another. This can be achieved by running DARTEL so that both images are matched with a common template, and

composing the resulting spatial transformations. This can be achieved by aligning them both with a pre-existing template, but it is also possible to use the *Run DARTEL (create Template)* option with the imported data of only two subjects. Once the flow fields (u_rc1*.nii files) have been estimated, then the resulting deformations can be composed using *SPM→Utils→Deformations*. If the objective is to warp A.nii to align with B.nii, then the procedure is set up by:

- **Deformations**
  - **Composition**
    - **DARTEL flow**
      - **Flow field**: Specify the u_rc1A_Template.nii flow field.
      - **Forward/Backwards**: Backward.
      - **Time Steps**: Usually 64.
    - **DARTEL flow**
      - **Flow Field**: Specify the u_rc1B_Template.nii flow field.
      - **Forward/Backwards**: Forward.
      - **Time Steps**: Usually 64.
    - **Identity**
      - **Image to base Id on**: Specify B.nii in order to have the deformed image(s) written out at this resolution, and with the same orientations etc (ie so there is a voxel-for-voxel alignment, rather than having the images only aligned according to their "voxel-to-world" mappings).
  - **Save as**: You can save the composed deformations as a file. This would be called y_*.nii, which contains three volumes that encode the x, y and z components of the mapping.
  - **Apply to**: Specify A.nii, and any other images for that subject that you would like warped to match B.nii. Note that these other images must be in alignment according to *Check Reg*.
  - **Interpolation**: Specify the form of interpolation.

Suppose the image of one subject has been manually labelled, then this option is useful for transferring the labels on to images of other subjects.

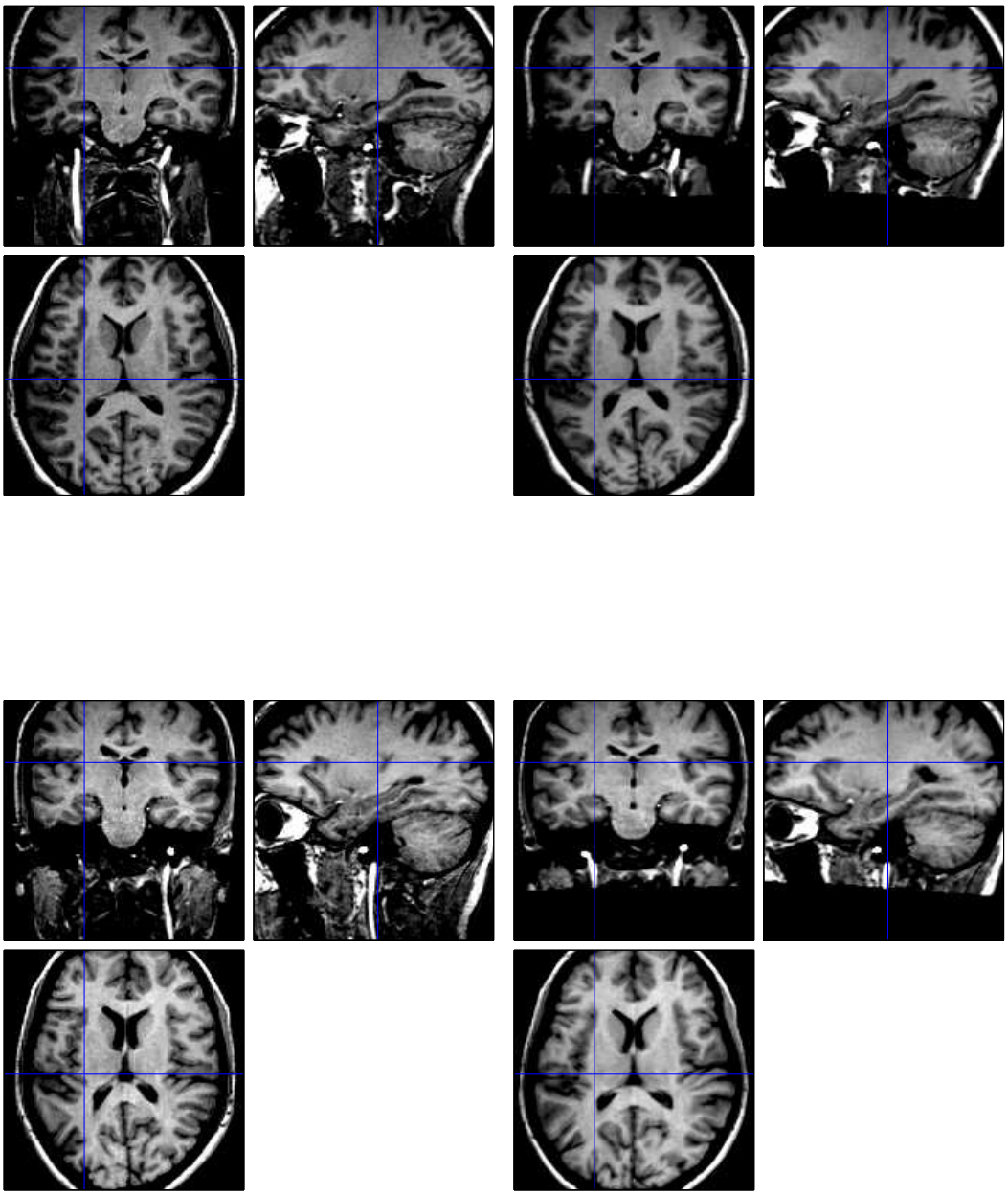


Figure 34.4: Composition of deformations to warp one individual to match another. Top-left: Original A.nii. Top-right: A.nii warped to match B.nii. Bottom-left: Original B.nii. Bottom-right: B.nii warped to match A.nii.

Part VIII

Batch Interface

Chapter 35

Batch interface

Details about the algorithms used for data processing are given in the other sections of this manual. This section explains how a sequence of processing steps can be run at once without MATLAB programming. SPM8 includes `matlabbatch`[1] which has been derived from the SPM5 batch system, but is also available as a separate package.

In `matlabbatch`, each data processing step is called "module". There are e.g. modules for spatial processing of MRI data (realignment, normalisation, smoothing), statistics (fMRI or factorial design specification, model estimation, contrast specification). A batch describes which modules should be run on what kind of data and how these modules depend on each other.

Compared to running each processing step interactively, batches have a number of advantages:

Documentation Each batch can be saved as a MATLAB script. All parameters (including default settings) are included in this script. Thus, a saved batch contains a full description of the sequence of processing steps and the parameter settings used.

Reproducibility Batches can be saved, even if not all parameters have been set. For a multi-subject study, this allows to create template batches. These templates contain all settings which do not vary across subjects. For each subject, they can be loaded and only subject-specific parts need to be completed.

Unattended execution Instead of waiting for a processing step to complete before entering the results in the next one, all processing steps can be run in the specified order without any user interaction.

Multiple batches Multiple batches can be loaded and executed together.

35.1 Batch tutorial - single subject

In this tutorial we will develop a batch for spatial processing and fMRI statistics of a single subject of the "Face" example dataset (see chapter 29). To follow this tutorial, it is not necessary to download the example dataset, except for the last step (entering subject dependent data).

To create a batch which can be re-used for multiple subjects in this study, it is necessary to collect/define

- study specific input data (e.g. MRI measurement parameters, time constants of the functional experiment, number of sessions),
- necessary processing steps,
- data flow between processing steps.

Subject specific input data (original functional and structural MRI data, subject specific experiment parameters) should be entered after the batch template has been saved.

[1] http://sourceforge.net/projects/matlabbatch

35.1.1 Study specific input data

This dataset consists of fMRI data acquired in a single session and a structural MRI. See section 35.2 to learn how to deal efficiently with multi-session data. MRI parameters and experiment details are described in chapter 29.

35.1.2 Necessary processing steps

Helper modules

Some SPM modules produce graphics output which is captured in a PostScript file in the current working directory. Also, a new directory needs to be created for statistics. The "BasicIO" menu provides a collection of modules which are useful to organise a batch. We will need the following modules:

- Named directory selector
- Change directory
- Make directory

SPM processing

For a classical SPM analysis, the following processing steps are necessary:

- Realignment
- Slice timing correction
- Coregistration
- Segmentation
- Normalisation
- Smoothing
- fMRI design
- Model estimation
- Contrasts
- Results report

35.1.3 Add modules to the batch

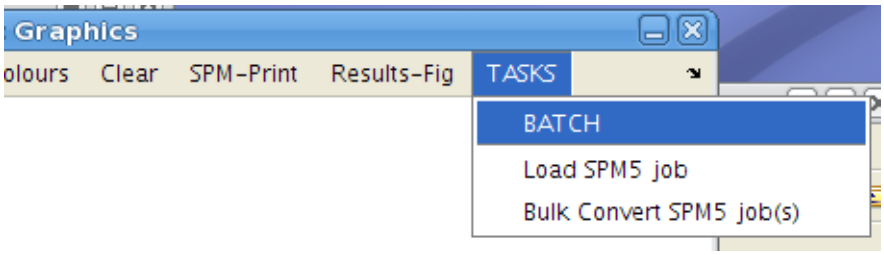


Figure 35.1: TASKS menu in "Graphics" window

The helper modules and the SPM processing modules can be assembled using the GUI. Locate the Graphics window and open the batch editor by selecting "BATCH" from the "TASKS" menu (fig. 35.1). First, add the helper modules, followed by the SPM modules in the order listed above. Do not configure any details until you have selected all modules.

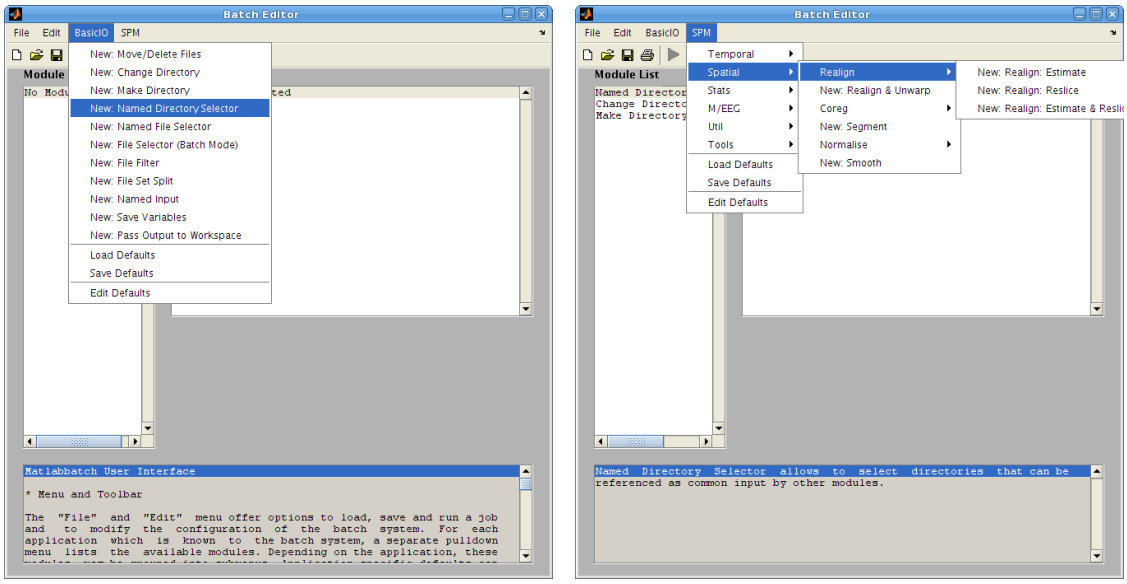


Figure 35.2: BasicIO and SPM application menus

35.1.4 Configure subject-independent data

Now, go through the batch and configure all settings that are subject-independent (e.g. the name of the analysis directory, slice timing parameters) as described in chapter 29. Do not enter any data that is specific for a certain subject. The values that need to be entered are not repeated here, please refer to the corresponding sections in chapter 29.

The file `man/batch/face_single_subject_template_nodeps.m` contains the batch after all modules have been added and subject-independent data has been entered.

Named Directory Selector

Input Name Give this selection a name (e.g. "Subject directory") - this name will be shown in the dependency list of this batch.

Directories Add a new directory selector, but do not enter a directory itself.

Change Directory

Nothing to enter now.

Make Directory

New Directory Name "categorical" - the name of the analysis directory. This directory will be created at batch run-time in the subject directory.

Realign: Estimate & Reslice

Data Add a new "Session" item. Do not enter any files for this session now.

Slice Timing

Data Add a new "Session" item. Do not enter any files for this session now.

Timing options Enter data for "Number of slices", "TR", "TA", "Slice order", "Reference slice".

Coreg: Estimate

Nothing to enter now.

Segment

Nothing to enter now.

Normalise: Write

Data Add a new "Subject". Do not enter any files now.

Writing Options Adjust bounding box, voxel sizes, interpolation

Smooth

FWHM Enter FWHM

fMRI model specification

Enter all data which is constant across subjects.

Timing parameters Enter values for "Units for design", "Interscan interval", "Microtime resolution", "Microtime onset"

Data & Design Add a new "Session" item. Do not enter scans, conditions or regressors yet. They will be added as dependencies or subject specific inputs. If you want to make sure to remember this, you can highlight "Multiple conditions" and select "Clear Value" from the "Edit" menu. Do the same for "Multiple regressors". This will mark both items with an `<-X`, indicating that something must be entered there.

Factorial design Enter the specification for both factors.

Basis functions Select the basis function and options you want to use.

Model estimation

Nothing to be entered yet for classical estimation.

Contrast manager

If you have selected the "Factorial design" option as described above, SPM will automatically create some contrasts for you. Here, you can create additional T- or F-contrasts. As an example, we will add an "Effects of interest" F-contrast.

Contrast session Add a new "F-contrast" item.

Name Enter a name for this contrast, e.g. "Effects of interest".

Contrast vectors Add a new "Contrast vector" item. F-contrasts can have multiple rows. You can either enter a contrast matrix in an "F contrast vector" entry, or enter them row by row. To test for the effects of interest (1 basis function and 2 derivatives for each of the four conditions) enter `eye(12)` as F contrast vector.

Replicate over sessions This design does not have multiple sessions, so it is safe to say "No" here.

Results report

Reviewing individual results for a large number of subjects can be very time consuming. Results report will print results from selected contrasts to a PostScript file.

Contrast(s) Enter `Inf` to print a report for each of the defined contrasts.

35.1.5 Data flow

In chapter 29, each processing step was performed on its own. In most cases, output data was simply passed on from one module to the next. This scheme is illustrated in figure 35.3. Only the coloured items at the top of the flow chart are subject specific and need to be entered in the final batch. All arrow connections are subject-independent and can be specified in the batch template.

Add dependencies

Based on the data flow in figure 35.3, modules in the batch can now be connected. The batch containing all dependencies can be found in `man/batch/face_single_subject_template.m`.

Again, start editing at the top of the batch:

Named Directory Selector

Nothing to enter now.

Change Directory

Directory Press "Dependency" and select "Subject directory(1)". At run time, SPM will change to this directory before batch processing continues.

Make Directory

Parent Directory Press "Dependency" and select "Subject directory(1)". The "categorial" directory will be created in this directory.

Realign: Estimate & Reslice

Nothing to enter now.

Slice Timing

Session Press "Dependency" and select "Resliced Images (Sess 1)".

Coreg: Estimate

Reference Image Press "Dependency" and select "Mean Image".

Segment

Data Press "Dependency" and select "Coregistered Images". At run time, this will resolve to the coregistered anatomical image.

Normalise: Write

Parameter File Press "Dependency" and select "Norm Params File Subj→MNI (Subj 1)".

Images to Write Press "Dependency" and select "Slice Timing Corr. Images (Sess 1)".

Smooth

Images to Smooth Press "Dependency" and select "Normalised Images (Subj 1)"

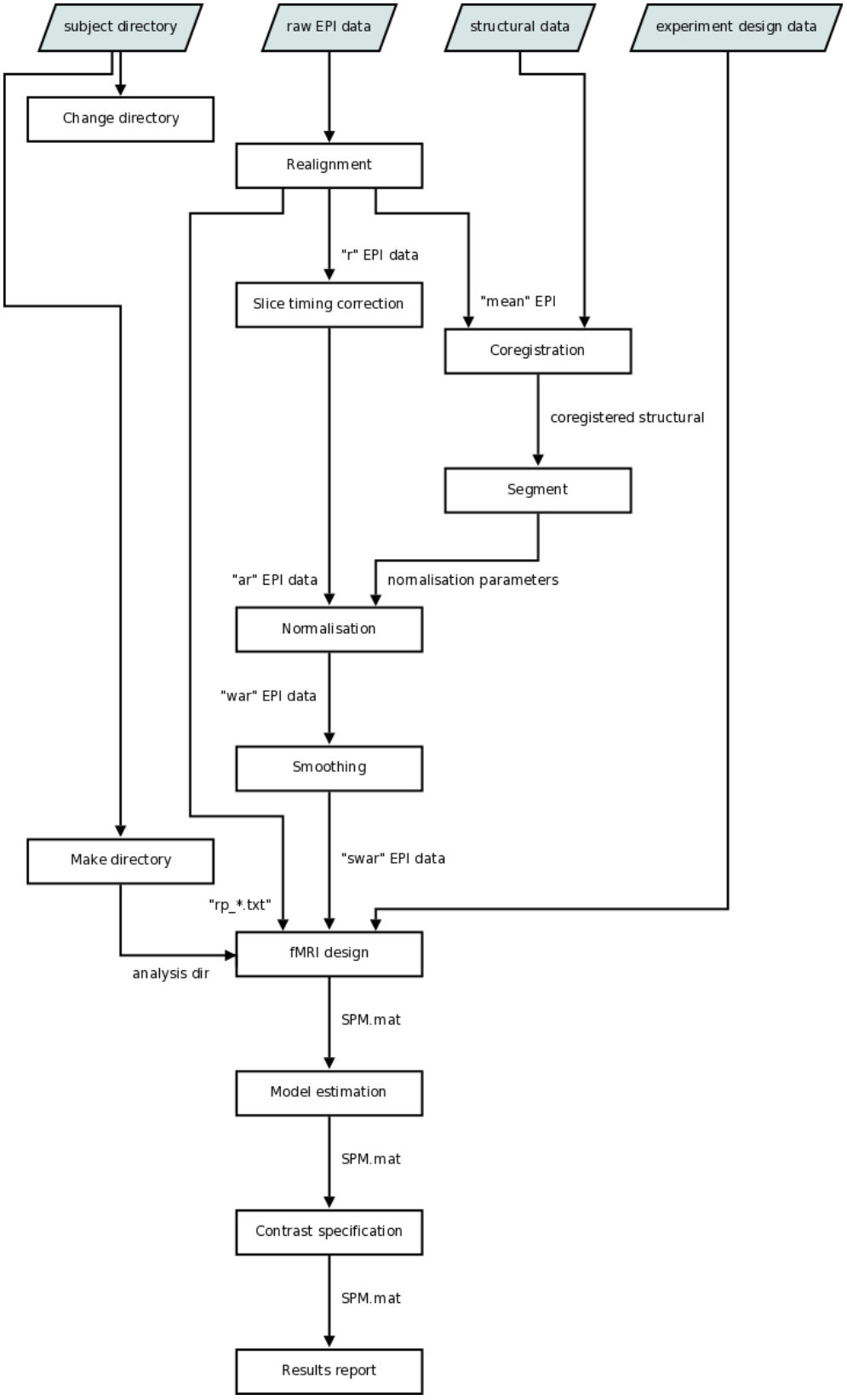


Figure 35.3: Flow chart for batch

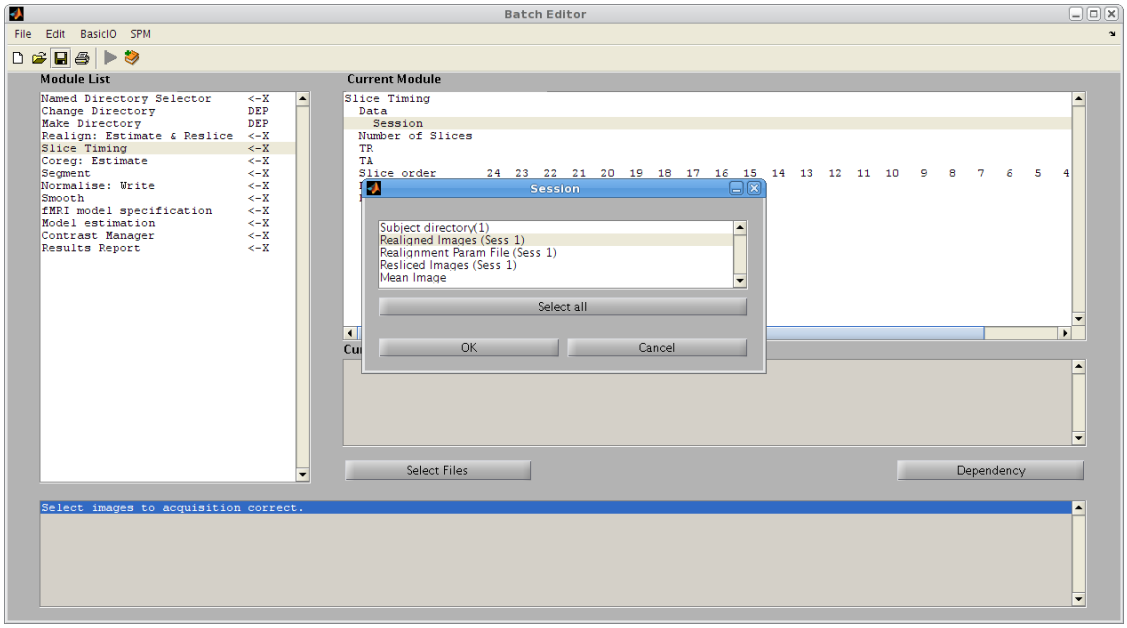


Figure 35.4: Dependency selection

fMRI model specification

Directory Press "Dependency" and select "Make Directory 'categorical' "

Scans Press "Dependency" and select "Smoothed Images". Note: this works because there is only one session in our experiment. In a multisession experiments, images from each session may be normalised and smoothed using the same parameters, but the smoothed images need to be split into sessions again. See section 35.2 how this can be done.

Multiple regressors Press "Dependency" and select "Realignment Param File (Sess 1)".

Model estimation

Select SPM.mat Press "Dependency" and select "SPM.mat File (fMRI Design&Data)".

Contrast manager

Select SPM.mat Press "Dependency" and select "SPM.mat File (Estimation)".

Results report

Select SPM.mat Press "Dependency" and select "SPM.mat File (Contrasts)".

35.1.6 Entering subject-specific data

Now, only 4 modules should have open inputs left (marked with `<-X`). These inputs correspond to data which vary over the subjects in your study:

Named Directory Selector Subject directory

Realign: Estimate & Reslice Raw EPI data for the fMRT session

Coreg: Estimate Anatomical image to be coregistered to mean EPI

fMRI model specification Names, conditions and onsets of your experimental conditions, specified in a multiple conditions .mat file.

Using the GUI, you can now perform these steps for each subject:

1. load the template batch
2. enter subject-specific data
3. save batch under a subject specific name.

After that, all batches for all subjects can be loaded and run at once.

This process can be automated using some basic MATLAB scripting. See section 35.2.3 for details.

35.2 Advanced features

35.2.1 Multiple sessions

If an fMRI experiment has multiple sessions, some processing steps need to take this into account (slice timing correction, realignment, fMRI design), while others can work on all sessions at once (normalisation, smoothing).

Two modules in BasicIO help to solve this problem:

Named File Selector Files can be entered here session by session. Note that this file selector selects all files (not restricted to images) by default. To select only images, set the filter string to something like `.*nii$` or `.*img$`.

File Set Split This module splits a list of files based on an index vector. Named file selector provides such an index vector to split the concatenation of all selected images into individual sessions again.

35.2.2 Processing multiple subjects in GUI

There are different ways to process multiple subjects in the batch editor:

- Add the necessary processing steps when creating the job.
- Create a per-subject template, save it and load it multiple times (i.e. in the file selector, add the same file multiple times to the list of selected files).
- Use "Run Batch Jobs" from "BasicIO"

In all cases, the data for all subjects has to be entered through the GUI, and computation will be done for all subjects at once after all data is entered. There is an example job `face_multi_subject_template.m` that demonstrates the usage of "Run Batch Jobs" to run the single subject template job described above. Note that the order and type of inputs in the single subject template is important. Also, consistency checks are limited. If inconsistent data is entered, the job will fail to execute and return an error message.

To run this job for multiple subjects, simply repeat the "Runs" item as many times as necessary and fill in the required data.

35.2.3 Command line interface

The command line interface is especially useful to run multiple jobs at once without user interaction, e.g. to process multiple subjects or to combine separate processing steps. There is a "high-level" interface using `spm_jobman`, which combines "low-level" callbacks to `cfg_util`.

Complete and run a pre-specified job

```
spm_jobman('serial', job[,'', input1, input2 ...])
```

This interface is called the "serial" interface. It takes a job, and asks for the input to any open configuration items one after another. If a list of inputs is supplied, these will be filled in (if they are appropriate). After all inputs are filled, the job will be run. Note that only items without

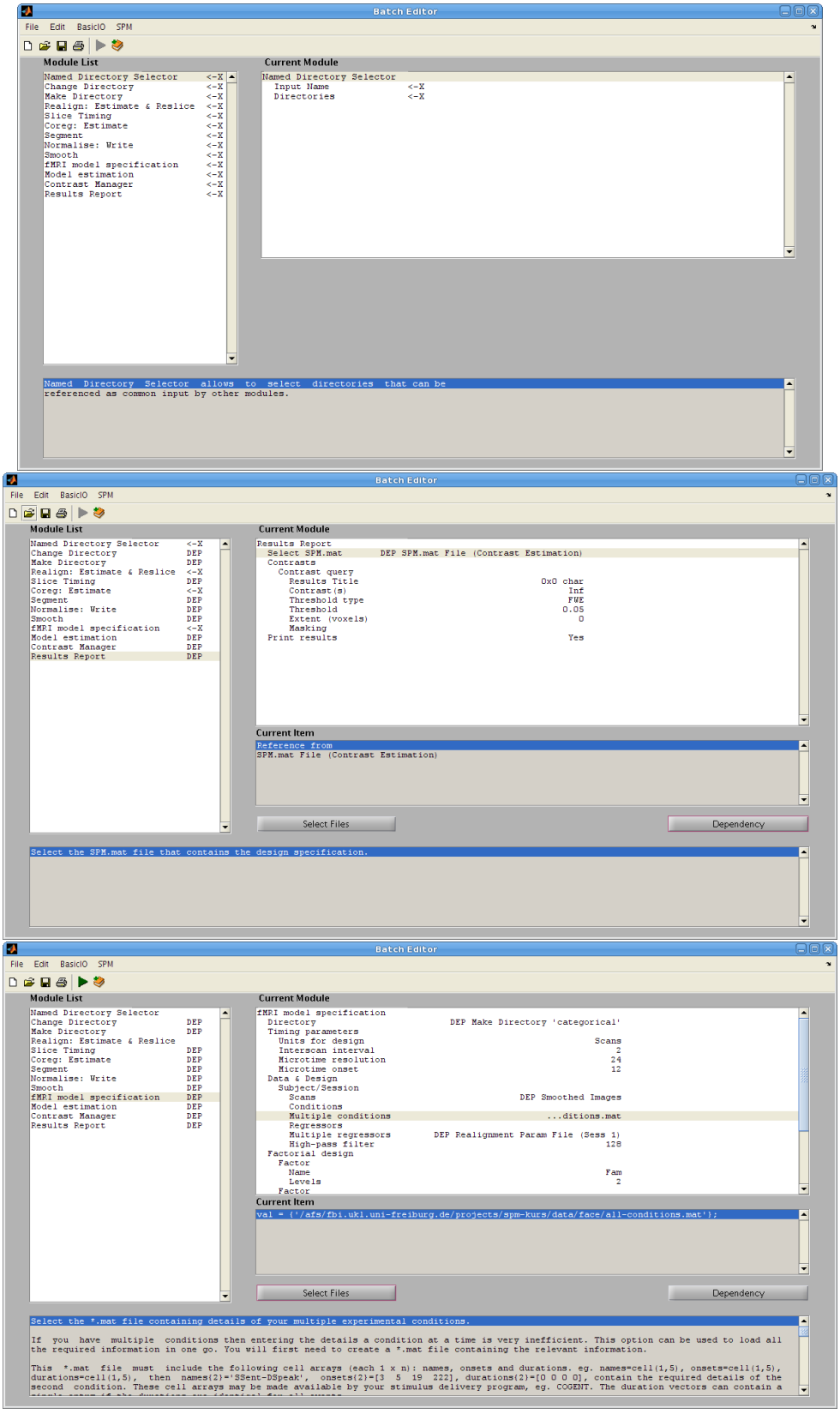


Figure 35.5: All stages of batch entry

a pre-set value will be filled (marked with <-X in the GUI). To force a item to to be filled, use "Edit:Clear Value" in the GUI or set its value to '<UNDEFINED>' in the harvested job.

The job argument is very flexible, it can e.g. be a job variable, the name of a script creating a job variable, even a cell list of any mixture of variables and scripts. All job snippets found will be concatenated into a single job, the missing inputs will be filled and the resulting job will be run.

The following MATLAB code snippet can be used to fill in and run the face_single_subject_template.m batch. Instead of using spm_select as file selector, files could be read from e.g. a MATLAB variable:

```
% Collect missing inputs
subjdir = cellstr(spm_select([1 1],'dir','Subject Dir'));
subjepi = cellstr(spm_select([1 inf],'image','Raw EPI images'));
subjana = cellstr(spm_select([1 1],'image','Anatomy image'));
subjcon = cellstr(spm_select([1 1],'mat','Multiple conditions'));
% Run batch, assuming face_single_subject_template.m is in
% MATLAB path or working directory.
% If it is not, then a full path and file name can be used instead.
spm_jobman('serial', 'face_single_subject_template.m', '', ...
           subjdir, subjepi, subjana, subjcon);
```

To run the same job for multiple subjects, this code could be modified to:

```
% Collect missing inputs
subj1dir = cellstr(spm_select([1 1],'dir','Subject Dir'));
subj1epi = cellstr(spm_select([1 inf],'image','Raw EPI images'));
subj1ana = cellstr(spm_select([1 1],'image','Anatomy image'));
subj1con = cellstr(spm_select([1 1],'mat','Multiple conditions'));
subj2dir = cellstr(spm_select([1 1],'dir','Subject Dir'));
subj2epi = cellstr(spm_select([1 inf],'image','Raw EPI images'));
subj2ana = cellstr(spm_select([1 1],'image','Anatomy image'));
subj2con = cellstr(spm_select([1 1],'mat','Multiple conditions'));
% Run batch, assuming face_single_subject_template.m is in your
% MATLAB path or working directory
% If it is not, then a full path and file name can be used instead.
spm_jobman('serial', ...
    {'face_single_subject_template.m', 'face_single_subject_template.m'}, ...
    '', subj1dir, subj1epi, subj1ana, subj1con, ...
    subj2dir, subj2epi, subj2ana, subj2con);
```

Here the job argument has been replaced by a cell array containing the same job twice, and the argument list for the second subject has been added.

35.2.4 Modifying a saved job

In some cases, instead of using the serial interface it may be more appropriate to modify the fields of a saved or harvested job. By default, jobs are saved as MATLAB .mat files, but they can also be saved as .m files. These files contain a number of MATLAB commands, which will create a variable matlabbatch. The commands can be modified to set different values, add or remove options.

35.3 SPM5 to matlabbatch transition guide

This is a short overview to describe code organisation and interfaces between SPM and the batch system.

35.3.1 Code Reorganisation

The following paths have changed:

- `fullfile(spm('dir'),'matlabbatch')` Core batch system.
- `fullfile(spm('dir'),'config')` New SPM config files.
- `fullfile(spm('dir'),'oldconfig')` Old SPM config files (unused)
- `spm_jobman.m` and `spm_select.m` replaced with compatibility code
- `spm_Menu.fig` Callbacks adapted

Configuration code has been generated automatically from the existing SPM configuration using `cfg_struct2cfg` and `gencode`. This sometimes results in redundant/duplicate code. Also, conditional constructs like `if`, `case` may not have been considered.

Some assignments to configuration items are guarded by validity checks. Usually, there will be a warning issued if a wrong value is supplied. Special care needs to be taken for `.prog`, `.vfiles`, `.vout`, `.check` functions or function handles. The functions referenced here must be on MATLAB path before they are assigned to one of these fields. For toolboxes, this implies that toolbox paths must be added at the top of the configuration file.

For details, see section 35.4.

35.3.2 Interfaces between SPM and Matlabbatch

Unchanged harvested job structure.

Changed Top-level node in SPM config now called `spmjobs` instead of `jobs`. New overall top-level node `matlabbatch`. `spm_jobman` will convert and load SPM5 style batch jobs into the new batch system.

Changed Configuration file syntax - instead of structs, configuration items are now objects. Structs of type `<type>` are now represented as objects of class `cfg_<type>`. Existing SPM5 configuration can be imported using `cfg_struct2cfg`. There is a new class `cfg_exbranch` which is used for branches that have a `.prog` field.

Deprecated Virtual files have been replaced by dependencies. These require computations to return a single output argument (e.g. a cell, struct). Parts of this output argument can be passed on to new inputs at run-time. Virtual files are treated as a special output argument.

Added Interface to the batch system

- `cfg_util` Configuration management, job management, job execution
- `cfg_serial` A utility to fill missing inputs and run a job (optionally with a GUI input function)
- `cfg_ui` GUI - inspired by `spm_jobman`, but modified to work around some MATLAB GUI "features" (like input widgets loosing focus before editing has finished).

35.4 Configuration Code Details

Configuration code has been split into two files per configuration:

spm_cfg_*.m Configuration classes, `.check`, `.vout` subfunctions

spm_run_*.m Run-time code, takes job structure as input and returns output structure as specified in `.vout`.

In a few cases (where there was no additional magic in the code), run-time code has been integrated into the main SPM code. This may be useful to run test batches without using the configuration/batch system.

35.4.1 Virtual Outputs

Virtual outputs are described by arrays of `cfg_dep` objects. These objects contain a "source" and a "target" part. Functions may have more than one virtual output (e.g. one output per session, a collection of results variables). One `cfg_dep` object has to be created for each output.

Only two fields in the "source" part need to be set in a `.vout` callback:

sname A display name for this output. This will appear in the dependencies list and should describe the contents of this dependency.

src_output A subscript reference that can be used to address this output in the variable returned at run-time.

tgt_spec (optional) A description on what kind of inputs this output should be displayed as dependency. This is not very convenient yet, the `match` and `cfg_findspec` methods are very restrictive in the kind of expressions that are allowed.

The `.vout` callback will be evaluated once the configuration system thinks that enough information about the *structure* of the outputs is available. This condition is met, once all in-tree nodes `cfg_(ex)branch`, `cfg_choice`, `cfg_repeat` have the required number of child nodes.

The `.vout` callback is called with a job structure as input, but its code *should not rely* on the evaluation of any contents of this structure (or at least provide a fallback). The contents of the leaf nodes may not be set or may contain a dependency object instead of a value during evalution of `.vout`.

The "target" part will be filled by the configuration classes, the `src_exbranch` field is set in `cfg_util`.

35.4.2 SPM Startup

The top level configuration file for SPM is `spm_cfg.m`. It collects SPM core configuration files and does toolbox autodetection. If a toolbox directory contains `*_cfg_*.m` files, they will be loaded. Otherwise, if there are only SPM5-style `*_config_*.m` files, the configuration will be converted at run-time using `cfg_struct2cfg`.

35.4.3 Defaults Settings

In Matlabbatch, there are different ways to set defaults:

1. in the configuration file itself,
2. in a defaults file, which has a structure similar to a harvested job,
3. using a `.def` field for leaf items.

Defaults set using option 1 or 2 will only be updated at SPM/matlabbatch startup. Defaults set using option 3 will be set once a new job is started. These defaults take precedence over the other defaults.

In core SPM, these defaults refer to `spm_get_defaults`, which accesses `spm_defaults`. Toolboxes may use their own callback functions.

35.4.4 Toolbox Migration

In the `fullfile(spm('dir'),'toolbox')` folder there exists a migration utility `spm_tbx_config2cfg.m`. This utility will create a `*_cfg_*.m` and a `*_def_*.m` file based on the configuration tree given as input argument.

Toolboxes should set their defaults using the `.def` fields, using a mechanism similar to `spm_get_defaults`. This allows for flexibility without interfering with SPMs own defaults.

35.5 Utilities

35.5.1 Batch Utilities

Matlabbatch is designed to support multiple applications. A standard application "BasicIO" is enabled by default. Among other options, it contains file/file selection manipulation utilities which can be used as as dependency source if multiple functions require the same set of files as input argument. For debugging purposes, "Pass Output to Workspace" can be used to assign outputs of a computation to a workspace variable.

The `cfg_confgui` folder contains an application which describes all configuration items in terms of configuration items. It is not enabled by default, but can be added to the batch system using `cfg_util('addapp'...)`. This utility can be used generate a batch configuration file with the batch system itself.

35.5.2 Matlab Code Generation

The `gencode` utility generates MATLAB `.m` file code for any kind of MATLAB variable. This is used to save batch files as well as to generate configuration code.

35.5.3 Configuration Management

The backend utility to manage the configuration data is `cfg_util`. It provides callbacks to add application configurations, and to load, modify, save or run jobs. These callbacks are used by two frontends: `cfg_ui` is a MATLAB GUI, while `cfg_serial` can be used both as a GUI and in script mode. In script mode, it will fill in job inputs from an argument list. This allows to run predefined jobs with e.g. subject dependent inputs without knowing the exact details of the job structure.

Part IX

Bibliography

Bibliography

[1] J. Andersson, J. Ashburner, and K.J. Friston. A global estimator unbiased by local changes. *NeuroImage*, 13(6):1193–1206, 2001.

[2] J. Andersson, C. Hutton, J. Ashburner, R. Turner, and K.J. Friston. Modelling geometric deformations in EPI time series. *NeuroImage*, 13(5):903–919, 2001.

[3] J. Ashburner. A fast diffeomorphic image registration algorithm. *NeuroImage*, 38(1):95–113, 2007.

[4] J. Ashburner, J. Andersson, and K.J. Friston. Image registration using a symmetric prior - in three-dimensions. *Human Brain Mapping*, 9(4):212–225, 2000.

[5] J. Ashburner and K.J. Friston. The role of registration and spatial normalization in detecting activations in functional imaging. *Clinical MRI/Developments in MR*, 7(1):26–28, 1997.

[6] J. Ashburner and K.J. Friston. Nonlinear spatial normalization using basis functions. *Human Brain Mapping*, 7(4):254–266, 1999.

[7] J. Ashburner and K.J. Friston. Voxel-based morphometry – the methods. *NeuroImage*, 11:805–821, 2000.

[8] J. Ashburner and K.J. Friston. Why voxel-based morphometry should be used. *NeuroImage*, 14(6):1238–1243, 2001.

[9] J. Ashburner and K.J. Friston. Unified segmentation. *NeuroImage*, 26:839–851, 2005.

[10] J. Ashburner, P. Neelin, D. L. Collins, A. C. Evans, and K.J. Friston. Incorporating prior knowledge into image registration. *NeuroImage*, 6:344–352, 1997.

[11] S. Baillet, J.C. Mosher, and R.M. Leahy. Electromagnetic brain mapping. *IEEE Sign. Proc. Mag.*, 18:14–30, 2001.

[12] P.J. Besl and N.D. McKay. A method for registration of 3-d shapes. *IEEE Trans. Pat. Anal. and Mach. Intel.*, 142:239–256, 1992.

[13] C. Buchel and K.J. Friston. Modulation of connectivity in visual pathways by attention: Cortical interactions evaluated with structural equation modelling and fMRI. *Cerebral Cortex*, 7:768–778, 1997.

[14] C. Buchel, A.P. Holmes, G. Rees, and K.J. Friston. Characterizing stimulus-response functions using nonlinear regressors in parametric fMRI experiments. *NeuroImage*, 8:140–148, 1998.

[15] C. Buchel, R.J.S. Wise, C.J. Mummery, J.B. Poline, and K.J. Friston. Nonlinear regression in parametric activation studies. *NeuroImage*, 4:60–66, 1996.

[16] C.C. Chen, S.J. Kiebel, and K.J. Friston. Dynamic causal modelling of induced responses. *NeuroImage*, 41(4):1293–1312, 2008.

[17] A. Collignon, F. Maes, D. Delaere, D. Vandermeulen, P. Suetens, and G. Marchal. Automated multi-modality image registration based on information theory. In Y. Bizais, C. Barillot, and R. Di Paola, editors, *Proc. Information Processing in Medical Imaging*, pages 263–274, Dordrecht, The Netherlands, 1995. Kluwer Academic Publishers.

[18] R. W. Cox and A. Jesmanowicz. Real-time 3D image registration for functional MRI. *Magnetic Resonance In Medicine*, 42:1014–1018, 1999.

[19] A.M. Dale and M. Sereno. Improved localization of cortical activity by combining EEG and MEG with MRI surface reconstruction: a linear approach. *J. Cognit. Neurosci.*, 5:162–176, 1993.

[20] O. David, S.J. Kiebel, L. Harrison, J. Mattout, J. Kilner, and K.J. Friston. Dynamic causal modelling of evoked responses in EEG and MEG. *NeuroImage*, 30:1255–1272, 2006.

[21] W. F. Eddy, M. Fitzgerald, and D. C. Noll. Improved image registration by using Fourier interpolation. *Magnetic Resonance in Medicine*, 36:923–931, 1996.

[22] Matthias Fastenrath, K.J. Friston, and S.J. Kiebel. Dynamical causal modelling for m/eeg: Spatial and temporal symmetry constraints. *NeuroImage*, 2008.

[23] K.J. Friston, J. Ashburner, S.J. Kiebel, T.E. Nichols, and W.D. Penny, editors. *Statistical Parametric Mapping: The Analysis of Functional Brain Images*. Academic Press, 2007.

[24] K.J. Friston, C. Frith, R.S.J. Frackowiak, and R. Turner. Characterizing dynamic brain responses with fMRI. *NeuroImage*, 2:166–172, 1995.

[25] K.J. Friston, D.E. Glaser, R.N.A. Henson, S.J. Kiebel, C. Phillips, and J. Ashburner. Classical and Bayesian inference in neuroimaging: Applications. *NeuroImage*, 16:484–512, 2002.

[26] K.J. Friston, L. Harrison, J. Daunizeau, S.J. Kiebel, C. Phillips, N. Trujillo-Bareto, R.N.A. Henson, G. Flandin, and J. Mattout. Multiple sparse priors for the m/eeg inverse problem. *NeuroImage*, 39(3):1104–1120, 2008.

[27] K.J. Friston, L. Harrison, and W.D. Penny. Dynamic Causal Modelling. *NeuroImage*, 19(4):1273–1302, 2003.

[28] K.J. Friston, R.N.A. Henson, C. Phillips, and J. Mattout. Bayesian estimation of evoked and induced responses. *Human Brain Mapping*, 27:722–735, 2005.

[29] K.J. Friston, A. Mechelli, R. Turner, and C.J. Price. Nonlinear responses in fMRI: The Balloon model, Volterra kernels and other hemodynamics. *NeuroImage*, 12:466–477, 2000.

[30] K.J. Friston and W.D. Penny. Posterior probability maps and SPMs. *NeuroImage*, 19(3):1240–1249, 2003.

[31] K.J. Friston, W.D. Penny, and D.E. Glaser. Conjunction revisited. *NeuroImage*, 25(3):661–667, 2005.

[32] K.J. Friston, W.D. Penny, C. Phillips, S.J. Kiebel, G. Hinton, and J. Ashburner. Classical and Bayesian inference in neuroimaging: Theory. *NeuroImage*, 16:465–483, 2002.

[33] M.I. Garrido, K.J. Friston, K.E. Stephan, S.J. Kiebel, T. Baldeweg, and J. Kilner. The functional anatomy of the MMN: A DCM study of the roving paradigm. *NeuroImage*, 42(2):936–944, 2008.

[34] M.I. Garrido, J. Kilner, S.J. Kiebel, and K.J. Friston. Evoked brain responses are generated by feedback loops. *PNAS*, 104(52):20961–20966, 2007.

[35] M.I. Garrido, J. Kilner, S.J. Kiebel, K.E. Stephan, and K.J. Friston. Dynamic causal modelling of evoked potentials: A reproducibility study. *NeuroImage*, 36:571–580, 2007.

[36] D.E. Glaser. Variance Components. In R.S.J. Frackowiak, K.J. Friston, C. Frith, R. Dolan, K.J. Friston, C.J. Price, S. Zeki, J. Ashburner, and W.D. Penny, editors, *Human Brain Function*. Academic Press, 2nd edition, 2003.

[37] R.N.A. Henson. Analysis of fMRI time series. In R.S.J. Frackowiak, K.J. Friston, C. Frith, R. Dolan, K.J. Friston, C.J. Price, S. Zeki, J. Ashburner, and W.D. Penny, editors, *Human Brain Function*. Academic Press, 2nd edition, 2003.

[38] R.N.A. Henson and W.D. Penny. ANOVAs and SPM. Technical report, Wellcome Department of Imaging Neuroscience, 2003.

[39] R.N.A. Henson, C.J. Price, M.D. Rugg, R. Turner, and K.J. Friston. Detecting latency differences in event-related bold responses: application to words versus non-words and initial versus repeated face presentations. *NeuroImage*, 15(1):83–97, 2002.

[40] R.N.A. Henson, M.D. Rugg, and K.J. Friston. The choice of basis functions in event-related fMRI. *NeuroImage*, 13(6):149, June 2001. Supplement 1.

[41] R.N.A. Henson, T. Shallice, M.L. Gorno-Tempini, and R. Dolan. Face repetition effects in implicit and explicit memory tests as measured by fMRI. *Cerebral Cortex*, 12:178–186, 2002.

[42] C. Hutton, A. Bork, O. Josephs, R. Deichmann, J. Ashburner, and R. Turner. Image distortion correction in fMRI: A quantitative evaluation. *NeuroImage*, 16:217–240, 2002.

[43] C. Hutton, R. Deichmann, R. Turner, and J. L. R. Andersson. Combined correction for geometric distortion and its interaction with head motion in fMRI. In *Proc. ISMRM 12*, Kyoto, Japan, 2004.

[44] M. Jenkinson. Fast, automated, N-dimensional phase-unwrapping algorithm. *Magnetic Resonance in Medicine*, 49:193–197, 2003.

[45] P. Jezzard and R. S. Balaban. Correction for geometric distortions in echoplanar images from B0 field variations. *Magnetic Resonance in Medicine*, 34:65–73, 1995.

[46] S.J. Kiebel. The general linear model. In R.S.J. Frackowiak, K.J. Friston, C. Frith, R. Dolan, K.J. Friston, C.J. Price, S. Zeki, J. Ashburner, and W.D. Penny, editors, *Human Brain Function*. Academic Press, 2nd edition, 2003.

[47] S.J. Kiebel, J. Daunizeau, C. Phillips, and K.J. Friston. Variational bayesian inversion of the equivalent current dipole model in eeg/meg. *NeuroImage*, 39(2):728–741, 2008.

[48] S.J. Kiebel, O. David, and K.J. Friston. Dynamic causal modelling of evoked responses in EEG/MEG with lead-field parameterization. *NeuroImage*, 30:1273–1284, 2006.

[49] S.J. Kiebel, M.I. Garrido, and K.J. Friston. Dynamic causal modelling of evoked responses: The role of intrinsic connections. *NeuroImage*, 36:332–345, 2007.

[50] S.J. Kiebel, M.I. Garrido, R. Moran, and K.J. Friston. Dynamic causal modelling for eeg and meg. *Cognitive Neurodynamics*, 2(2):121–136, 2008.

[51] S.J. Kiebel, S. Klppel, N. Weiskopf, and K.J. Friston. Dynamic causal modeling: A generative model of slice timing in fMRI. *NeuroImage*, 34:1487–1496, 2007.

[52] V. Litvak and K.J. Friston. Electromagnetic source reconstruction for group studies. *NeuroImage*, 2008.

[53] F. Maes, A. Collignon, D. Vandermeulen, G. Marchal, and P. Seutens. Multimodality image registration by maximisation of mutual information. *IEEE Transactions on Medical Imaging*, 16:187–197, 1997.

[54] A.C. Marreiros, J. Daunizeau, S.J. Kiebel, and K.J. Friston. Population dynamics: Variance and the sigmoid activation function. *NeuroImage*, 42(1):147–157, 2008.

[55] J. Mattout, C. Phillips, W.D. Penny, M. Rugg, and K.J. Friston. Meg source localization under multiple constraints: an extended Bayesian framework. *NeuroImage*, 30(3):753–767, 2006.

[56] A. Mechelli, C.J. Price, K.J. Friston, and J. Ashburner. Voxel-based morphometry of the human brain: Methods and applications. *Current Medical Imaging Reviews*, pages 105–113, 2005.

[57] R. Moran, S.J. Kiebel, N. Rombach, W.T. O'Connor, K.J. Murphy, R.B. Reilly, and K.J. Friston. Bayesian estimation of synaptic physiology from the spectral responses of neural masses. *NeuroImage*, 42(1):272–284, 2008.

[58] R. Moran, S.J. Kiebel, K.E. Stephan, R.B. Reilly, J. Daunizeau, and K.J. Friston. A neural mass model of spectral responses in electrophysiology. *NeuroImage*, 37(3):706–720, 2007.

[59] T.E. Nichols, M. Brett, J. Andersson, T. Wager, and J.B. Poline. Valid conjunction inference with the minimum statistic. *NeuroImage*, 25:653–660, 2005.

[60] W.D. Penny and G. Flandin. Bayesian analysis of single-subject fMRI: SPM implementation. Technical report, Wellcome Department of Imaging Neuroscience, 2005.

[61] W.D. Penny, G. Flandin, and N. Trujillo-Bareto. Bayesian Comparison of Spatially Regularised General Linear Models. *Human Brain Mapping*, 28:275–293, 2005.

[62] W.D. Penny, A.P. Holmes, and K.J. Friston. Random effects analysis. In R.S.J. Frackowiak, K.J. Friston, C. Frith, R. Dolan, K.J. Friston, C.J. Price, S. Zeki, J. Ashburner, and W.D. Penny, editors, *Human Brain Function*. Academic Press, 2nd edition, 2003.

[63] W.D. Penny, S.J. Kiebel, and K.J. Friston. Variational Bayesian Inference for fMRI time series. *NeuroImage*, 19(3):727–741, 2003.

[64] W.D. Penny, K.E. Stephan, A. Mechelli, and K.J. Friston. Comparing Dynamic Causal Models. *NeuroImage*, 22(3):1157–1172, 2004.

[65] W.D. Penny, N. Trujillo-Bareto, and K.J. Friston. Bayesian fMRI time series analysis with spatial priors. *NeuroImage*, 24(2):350–362, 2005.

[66] C. Phillips, J. Mattout, M.D. Rugg, P. Maquet, and K.J. Friston. An empirical Bayesian solution to the source reconstruction problem in EEG. *NeuroImage*, 24:997–1011, 2005.

[67] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipes in C (Second Edition)*. Cambridge, Cambridge, 1992.

[68] C. Studholme, D. L. G. Hill, and D. J. Hawkes. An overlap invariant entropy measure of 3D medical image alignment. *Pattern Recognition*, 32:71–86, 1999.

[69] P. Thévenaz, T. Blu, and M. Unser. Interpolation revisited. *IEEE Transactions on Medical Imaging*, 19(7):739–758, 2000.

[70] M. Unser, A. Aldroubi, and M. Eden. B-spline signal processing: Part I – theory. *IEEE Transactions on Signal Processing*, 41(2):821–833, 1993.

[71] M. Unser, A. Aldroubi, and M. Eden. B-spline signal processing: Part II – efficient design and applications. *IEEE Transactions on Signal Processing*, 41(2):834–848, 1993.

[72] W. M. Wells III, P. Viola, H. Atsumi, S. Nakajima, and R. Kikinis. Multi-modal volume registration by maximisation of mutual information. *Medical Image Analysis*, 1(1):35–51, 1996.

[73] I.C. Wright, P.K. McGuire, J.B. Poline, J.M. Travere, R.M. Murray, C. Frith, R.S.J. Frackowiak, and K.J. Friston. A voxel-based method for the statistical analysis of gray and white matter density applied to schizophrenia. *NeuroImage*, 2:244–252, 1995.