Bayesian analysis of single-subject fMRI data: SPM implementation

William D. Penny, Nelson Trujillo-Bareto, Guillaume Flandin and Karl J. Friston

Wellcome Department of Imaging Neuroscience, UCL, London UK.

Corresponding author:

William D. Penny,

Wellcome Department of Imaging Neuroscience,

12 Queen Square, London WC1N 3BG, UK.

Key Words: Variational Bayes, fMRI, spatial priors, effect size, general linear model, autoregressive model, Laplacian, smoothing, spatial regularisation

Introduction

This document describes how Bayesian analysis of single subject fMRI is implemented in SPM.

Cross covariance formulae

The formulae in the appendix of (Penny et al., 2003) for computing the quantities $\{\tilde{\mathbf{A}}, \tilde{\mathbf{b}}, \tilde{\mathbf{C}}, \tilde{\mathbf{d}}, \tilde{g}\}$ (equations 63, 64, 50 and 77 respectively) though correct (apart from a few missing transpose operators here and there !) are computationally inefficient. This is because of the sums over time. For the lengths of time series typical in fMRI (eg. 300-400 scans) this creates a bottleneck when implementing the algorithm in MATLAB. The equations, which contain terms only up to second order (ie. quadratic), can however be re-arranged to isolate the sums over time so that they can be pre-computed. This effectively amounts to computing the following cross-covariances. This first set of terms depends on the design matrix only and therefore can be pre-computed for the whole volume

$$\mathbf{G}_x = \mathbf{X}^T \mathbf{X}$$
$$[k \times k]$$
$$\mathbf{S} = Cov(\tilde{\mathbf{X}}_t, \tilde{\mathbf{X}}_t)$$
$$[k^2 \times p^2]$$
$$\mathbf{R}_1 = Cov(\tilde{\mathbf{X}}_t, \mathbf{x}_t)$$
$$[k^2 \times p]$$

where the *Cov()* operator only involves terms from $p+1$ to T where p is the order of the AR model and T is the number of time points, $\mathbf{X}$ is the design matrix, and $\tilde{\mathbf{X}}$ is the embedded design matrix. Underneath each equation is the dimension into which the result is re-shaped into (where necessary). The following terms depend on the design matrix and the data and therefore can be pre-computed for each slice

$$\mathbf{R}_{xy} = Cov(\mathbf{d}_t, \mathbf{x}_t)$$
$$[p \times k]$$
$$\mathbf{D} = Cov(\mathbf{d}_t, \tilde{\mathbf{X}}_t)$$
$$[k \times p^2]$$
$$\mathbf{G}_{xy} = Cov(\mathbf{y}_t, \tilde{\mathbf{X}}_t)$$
$$[p \times k]$$
$$\mathbf{G}_y = Cov(\mathbf{d}_t, \mathbf{d}_t^T)$$
$$[p \times p]$$
$$\mathbf{g}_y = Cov(\mathbf{y}_t, \mathbf{d}_t)$$
$$[p \times 1]$$
$$\mathbf{g}_{xy} = Cov(\mathbf{X}_t^T, \mathbf{y}_t)$$
$$[k \times 1]$$

where $\mathbf{y}$ is the fMRI time series and $\mathbf{d}$ is the embedded time series. The new update formulae can then be related to these quantities. For T=400, the new formulae are about 200 times quicker.

Approximation Posterior Covariance Matrices

In the SPM implementation of this algorithm we do not store full covariance matrices for each voxel as this would require too much disk space. Instead we we store the posterior standard deviations of parameter estimates

$$d_n = \sqrt{diag(\hat{\Sigma}_n)} \tag{1}$$

(effectively images of error bars), AR coefficents, a_n, and noise standard deviation, $\sqrt{1/\lambda_n}$. These are stored in the files `SDbeta_000k.img`, `AR_000p.img` and `SDerror.img`. The approximate posterior covariances are then formed using a Taylor series expansion as follows. For each slice, *s*, we first compute the averages

$$\begin{aligned} \lambda_s &= \langle \lambda_n \rangle \\ a_s &= \langle a_n \rangle \\ V_s &= \langle V_n \rangle \\ b_s &= \langle B_{nn} \rangle \end{aligned} \tag{2}$$

where V_n is the posterior covariance of AR coefficients, and B_{nn} is the spatial precision, and from these compute a slice-specific error covariance matrix, $\hat{\Sigma}_s$. This is then normalised to produce a slice-specific error correlation matrix, R_s. The error correlation matrix at voxel *n* is then approximated using

$$R_n = R_s + \frac{dR_s}{d\lambda_s}\left(\lambda_n - \lambda_s\right) + \sum_{p=1}^{P} \frac{dR_s}{da_s(i)} \left(a_n(i) - a_s(i)\right) \tag{3}$$

where Jacobian matrices are stored for each slice. In SPM the slice-specific information is held in `SPM.PPM.slice(z).mean`. Finally the approximate covariance at voxel *n* is formed using

$$C_n = (d_n d_n^T).*R_n \tag{4}$$

This approximate covariance is used when inferences are made about contrasts of parameter estimates. Figure X plots the resulting approximate variance versus the true variance for two contrasts from the face fMRI data sets.

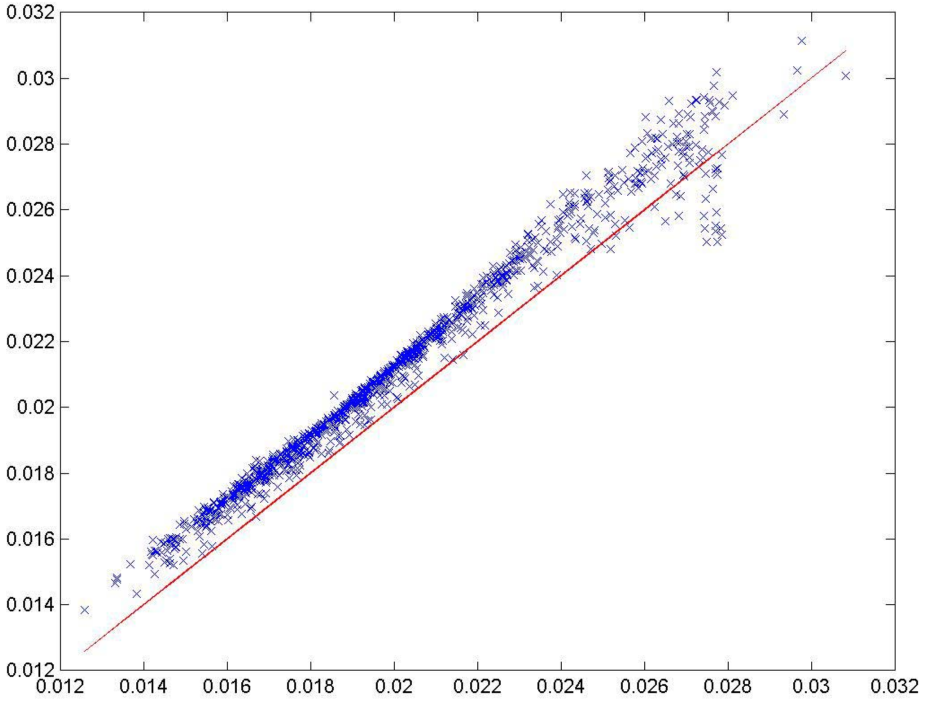
Acknowledgements

Will Penny is supported by the Wellcome Trust. The authors would also like to thank Rik Henson for commenting on the manuscript and for his advice on analysing the face processing data.

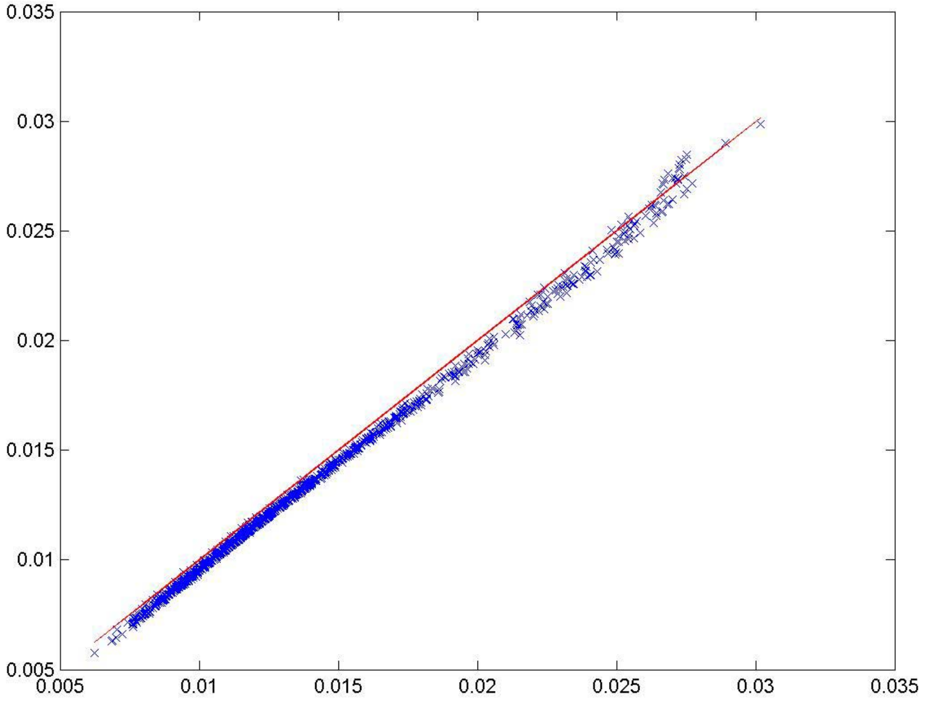
Figure X. **Approximating the Posterior Covariance**. Plots of approximate variance versus true variance for the face fMRI data and contrasts (a) main effect of faces and (b) main effect of fame. Crosses mark values at each of the voxels in slice *z=10* and the straight line shows *y=x*. The second contrast, being a differential contrast, shows smaller error.

Reference List

1. Penny,W., Kiebel,S., and Friston,K. (2003). Variational Bayesian inference for fMRI time series. Neuroimage. *19*, 727-741.


0.032
0.03
0.028
0.026
0.024
0.022
0.02
0.018
0.016
0.014
0.012
0.012 0.014 0.016 0.018 0.02 0.022 0.024 0.026 0.028 0.03 0.032

(a)


0.035
0.03
0.025
0.02
0.015
0.01
0.005
0.005 0.01 0.015 0.02 0.025 0.03 0.035

(b)