Empirical Bayes

Will Penny

3rd March 2011

General Linear Model

The General Linear Model (GLM) is given by

$$y = Xw + e$$

where y are data, X is a design matrix, and e are zero mean Gaussian errors with covariance V. The above equation implicitly defines the likelihood function

$$p(y|w) = \mathrm{N}(y; Xw, V)$$

where the Normal density is given by

$$\mathrm{N}(x; \mu, C) = \frac{1}{(2\pi)^{N/2}|C|^{1/2}} \exp\left(-\frac{1}{2}(x-\mu)^T C^{-1}(x-\mu)\right)$$

Maximum Likelihood

If we know V then we can estimate w by maximising the likelihood or equivalently the log-likelihood

$$L = -\frac{N}{2}\log 2\pi - \frac{1}{2}\log|V| - \frac{1}{2}(y - Xw)^T V^{-1}(y - Xw)$$

We can compute the gradient with help from the Matrix Reference Manual

$$\frac{dL}{dw} = X^T V^{-1} y - X^T V^{-1} X w$$

to zero. This leads to the solution

$$\hat{w}_{ML} = (X^T V^{-1} X)^{-1} X^T V^{-1} y$$

This is often referred to as Weighted Least Squares (WLS), $\hat{w}_{ML} = \hat{w}_{WLS}$. For example, some observations may be more reliable than others (Penny et al, 2007).

fMRI analysis

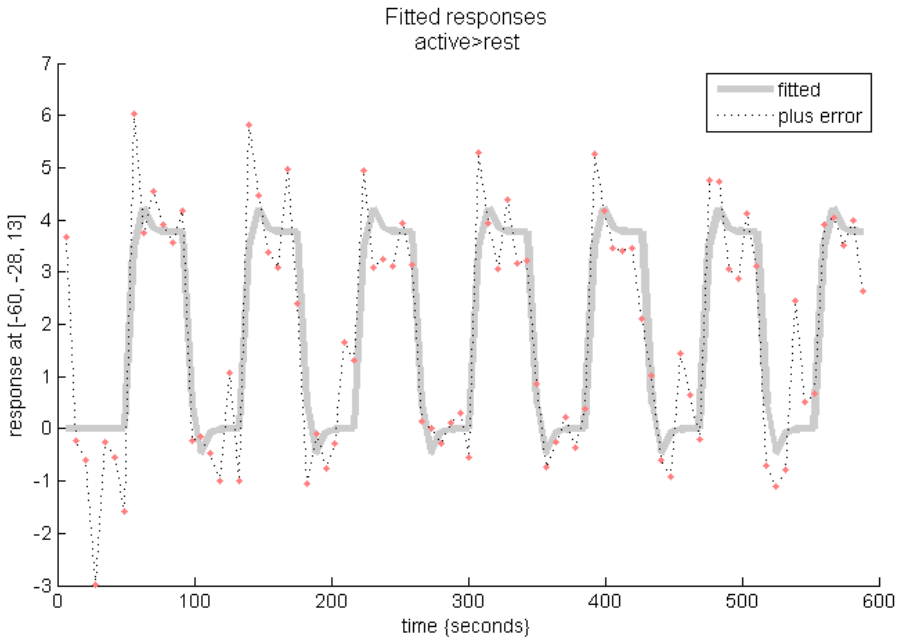
For fMRI time series analysis we have a linear model at each voxel i

$$y_i = Xw_i + e_i$$

$V_i = \text{Cov}(e_i)$ is estimated first (see later) and then the regression coefficients are computed using Maximum Likelihood (ML) estimation.

$$\hat{w}_i = (X^T V_i^{-1} X)^{-1} X^T V_i^{-1} y_i$$

The fitted responses are then $\hat{y}_i = X\hat{w}_i$ (SPM Manual)



fMRI analysis

The uncertainty in the ML estimates is given by

$$S = (X^T V_i^{-1} X)^{-1}$$

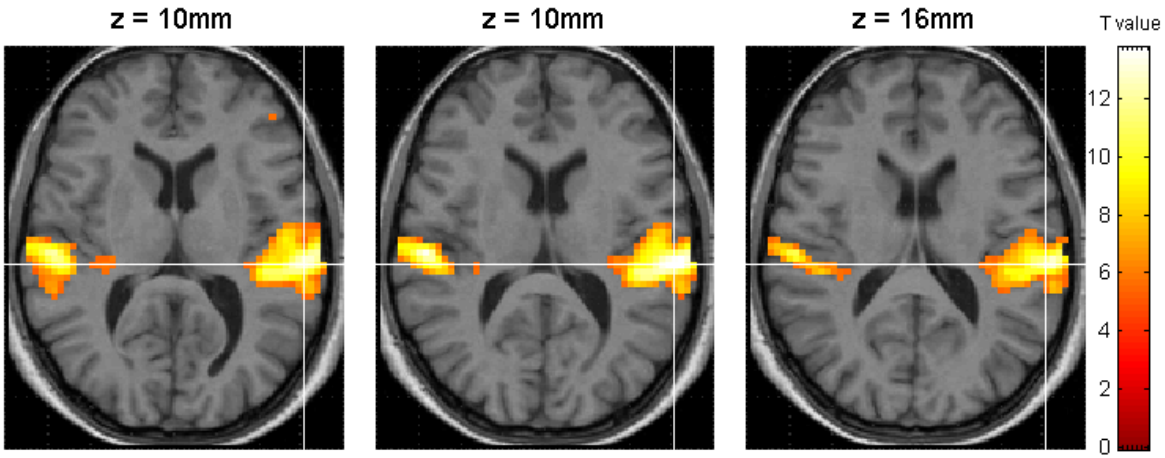
Contrast vectors c can then be used to test for specific effects

$$\mu_c = c^T \hat{w}_i$$

The uncertainty in the effect is then

$$\sigma_c^2 = c^T S c$$

and a t-score is then given by $t = \mu_c / \sigma_c$



Least Squares

For isotropic error covariance $V = \lambda I$, the normal equations are

$$\frac{dL}{dw} = \lambda X^T y - \lambda X^T X w$$

This leads to the Ordinary Least Squares (OLS) solution $\hat{w}_{ML} = \hat{w}_{OLS}$,

$$\hat{w}_{OLS} = (X^T X)^{-1} X^T y$$

Gradient Ascent

In gradient ascent approaches an objective function, L, is maximised by changing parameters w to follow the local gradient

$$\tau \frac{dw}{dt} = \frac{dL}{dw}$$

where τ is the time constant that defines the learning rate. In discrete time, parameters are then updated as

$$w_t = w_{t-1} + \frac{1}{\tau} \frac{dL}{dw_{t-1}}$$

Smaller time constants τ correspond to bigger updates at each step. That is, faster learning rates. In the *batch version* of gradient ascent the gradient is computed based on all pattern pairs x_n, y_n for $n = 1..N$. In the *sequential version* updates are based on gradients from individual patterns (see later).

Neural Implementations

Many 'neural implementations' or neural network models are derived by taking a standard statistical model eg. linear models, hierarchical linear models, (non-)linear dynamical systems, and then maximimising some cost function (eg the likelihood or posterior probability) using a *sequential gradient ascent* approach.

When the same model is applied to, for example, neuroimaging data more sophisticated optimisation methods eg. Newton Methods (see later) are used.

Online Learning - Sequential Gradient Ascent

In some situations observations may be made sequentially. For independent observations we have

$$p(y|w) = \prod_{n=1}^{N} p(y_n|w)$$

where

$$\begin{aligned} p(y_n|w) &= \mathrm{N}(y_n; x_n w, \lambda^{-1}) \\ &= \frac{1}{Z} \exp\left(-\frac{\lambda}{2}(y_n - x_n w)^2\right) \end{aligned}$$

and x_n is the nth row of X. Now take logs to give

$$\begin{aligned} L_n &= \log p(y_n|w) \\ &= -\frac{\lambda}{2}(y_n - x_n w)^2 - \log Z \end{aligned}$$

Predictions with smaller error have higher likelihood. Online learning then proceeds by following the gradients based on individual patterns.

Online Learning

For the linear model the learning rule for the ith coefficient is

$$\begin{aligned} \tau \frac{dw_i}{dt} &= \frac{dL_n}{dw_i} \\ &= \lambda x_n(i)(y_n - x_n w) \end{aligned}$$

Learning is faster for high precision observations, larger inputs and bigger prediction errors. One can use this in signal processing applications such as Real-Time fMRI.

Delta Rule

If λ is the same for all observations it can be absorbed into the learning rate. The above expression then reduces to the Delta Rule (Widrow and Hoff, 1960).

$$\tau \frac{dw_i}{dt} = x_n(i)(y_n - x_n w)$$

If observations have different precisions then

$$\tau \frac{dw_i}{dt} = \lambda_n x_n(i)(y_n - x_n w)$$

Example - Linear Regression

For the linear model

$$Y = Xw + e$$

with $Cov(e) = \lambda^{-1} I$ the log-likelihood is

$$L(w) = -\frac{\lambda}{2}(y - Xw)^T(y - Xw)$$

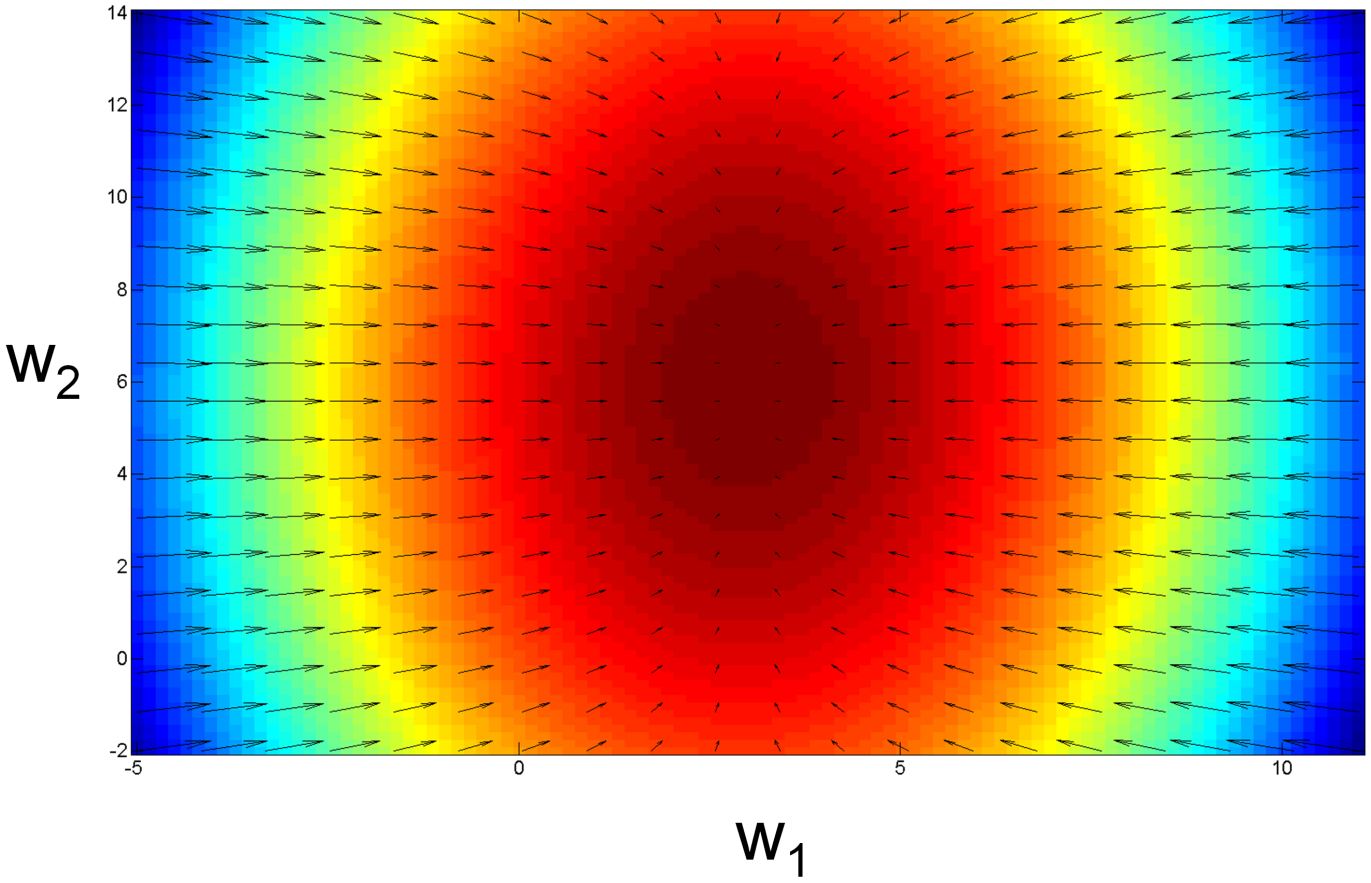
The gradient is

$$\begin{aligned} j(w) &= \frac{dL}{dw} \\ &= \lambda X^T y - \lambda X^T X w \\ &= \lambda X^T (y - Xw) \end{aligned}$$

Following this gradient corresponds to the Delta rule.

Example

For the log-likelihood $L(w)$



the local gradient does not always point in the direction of the optimum ($\hat{w}_{ML} = [3, 6]^T$). And convergence is slower for w_2 than w_1. This is because regressors did not have the same variance. They were also correlated.

The Problem with Gradient Ascent

A problem with (the batch version of) gradient descent is that large learning rates (big steps) will lead to instabilities.

This is because for many optimisation functions the local gradient does not point in the direction of the optimum.

Conversely, small learning rates lead to very slow convergence (in terms of the number of discrete steps).

Newton Method

This can be remedied with the Newton Method in which information about the curvature of the error surface is also used (Press, 1988; from 2nd-order Taylor expansion)

$$w_t = w_{t-1} - H_w^{-1} j_w$$

and

$$\begin{aligned} j_w(i) &= \frac{dL}{dw(i)} \\ H_w(i,j) &= \frac{d^2 L}{dw(i)dw(j)} \end{aligned}$$

where j_w is the gradient vector and H_w is the curvature matrix, also referred to as the Hessian.

As maximum is approached the gradient gets smaller, hence the curvature is negative (hence minus sign above).

Example - Linear Regression

The gradient is

$$j(w) = \lambda X^T y - \lambda X^T X w$$

as before and the curvature is

$$H = -\lambda X^T X$$

The parameter update is therefore

$$w_t = w_{t-1} + (X^T X)^{-1} X^T (y - X w_{t-1})$$

Hence

$$\begin{aligned} w_1 &= w_0 + \hat{w}_{ML} - (X^T X)^{-1} X^T X w_0 \\ &= \hat{w}_{ML} \end{aligned}$$

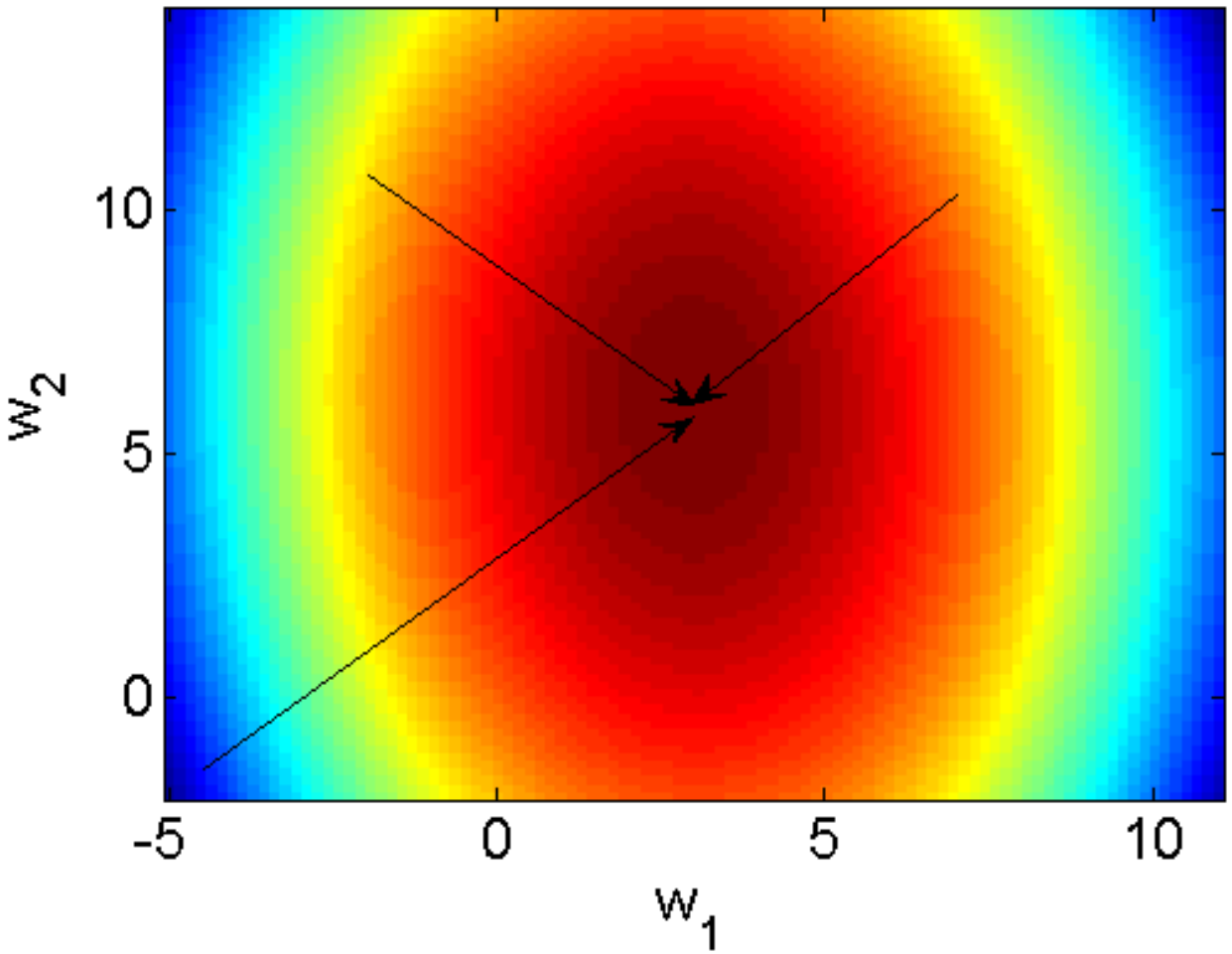
That is, learning in one step !

Example - Linear Regression

The Newton weight update is

$$w_1 = w_0 + (X^T X)^{-1} X^T (y - X w_0)$$

Learning in one step.



Bayesian GLM

A Bayesian GLM is defined as

$$\begin{aligned} y &= Xw + e_1 \\ w &= \mu_w + e_2 \end{aligned}$$

where the errors are zero mean Gaussian with covariances $\text{Cov}[e_1] = C_y$ and $\text{Cov}[e_2] = C_w$.

$$\begin{aligned} p(y|w) &\propto \exp\left(-\tfrac{1}{2}(y - Xw)^T C_y^{-1}(y - Xw)\right) \\ p(w) &\propto \exp\left(-\tfrac{1}{2}(w - \mu_w)^T C_w^{-1}(w - \mu_w)\right) \end{aligned}$$

Bayesian GLM

The posterior distribution is then

$$p(w|y) \propto p(y|w)p(w)$$

Taking logs and keeping only those terms that depend on w gives

$$\begin{aligned}
\log p(w|y) &= -\frac{1}{2}(y - Xw)^T C_y^{-1}(y - Xw) \\
&- \frac{1}{2}(w - \mu_w)^T C_w^{-1}(w - \mu_w) + .. \\
&= -\frac{1}{2} w^T (X^T C_y^{-1} X + C_w^{-1}) w \\
&+ w^T (X^T C_y^{-1} y + C_w^{-1} \mu_w) + ..
\end{aligned}$$

Bayesian GLM

If $p(x) = \mathrm{N}(x; m, S)$ then

$$p(x) \propto \exp\left(-\frac{1}{2}(x-m)^T S^{-1}(x-m)\right)$$

Taking logs of the Gaussian density $p(x)$ and keeping only those terms that depend on x gives

$$\log p(x) = -\frac{1}{2}x^T S^{-1} x + x^T S^{-1} m + ..$$

For our posterior we have

$$\begin{aligned} \log p(w|y) &= -\frac{1}{2}w^T(X^T C_y^{-1} X + C_w^{-1})w \\ &+ w^T(X^T C_y^{-1} y + C_w^{-1}\mu_w) + .. \end{aligned}$$

Equating terms gives

$$\begin{aligned} p(w|y) &= \mathrm{N}(m_w, S_w) \\ S_w^{-1} &= X^T C_y^{-1} X + C_w^{-1} \\ m_w &= S_w(X^T C_y^{-1} y + C_w^{-1}\mu_w) \end{aligned}$$

GLM posterior

The posterior density is

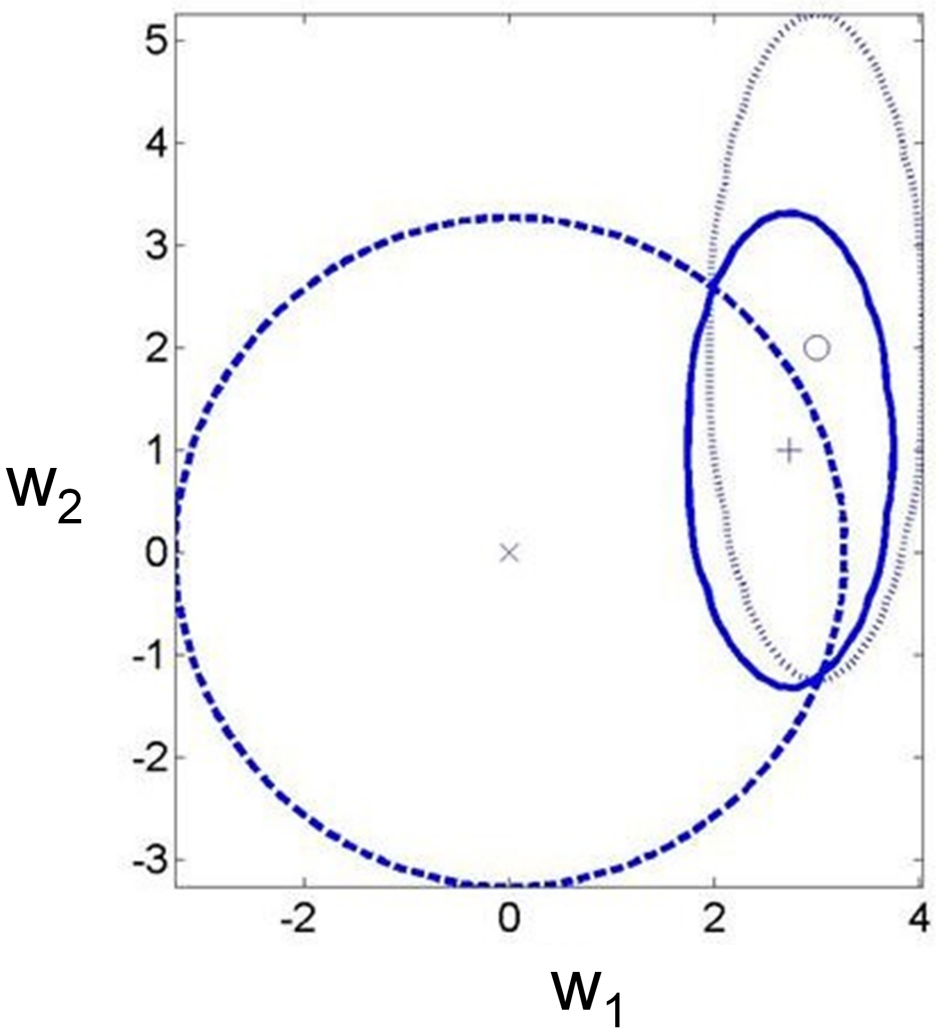
$$\begin{aligned} p(w|y) &= \mathrm{N}(m_w, S_w) \\ S_w^{-1} &= X^T C_y^{-1} X + C_w^{-1} \\ m_w &= S_w(X^T C_y^{-1} y + C_w^{-1} \mu_w) \end{aligned}$$

The posterior precision is the sum of the prior precision and the data precision.

The posterior mean is a relative precision weighted combination of the data mean and the prior mean.

If $\mu_w = 0$ we have a *shrinkage prior*.

Bayesian GLM with two parameters

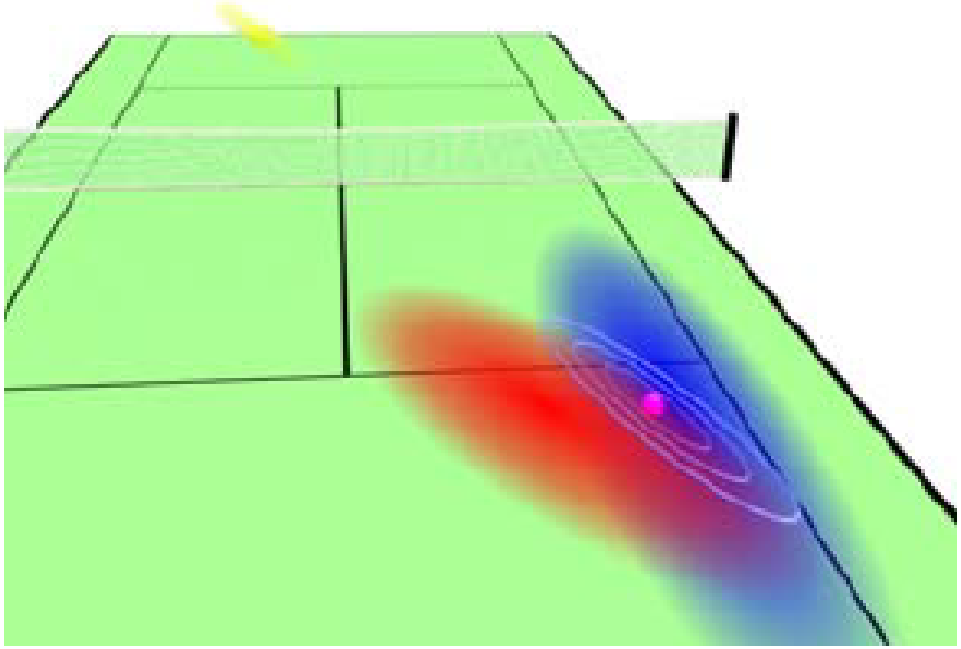


The prior (dashed line) has mean $\mu_w = [0, 0]^T$ (cross) and precision $C_w^{-1} = \text{diag}([1, 1])$. The likelihood (dotted line) has mean $X^T y = [3, 2]^T$ (circle) and precision $(X^T C_y^{-1} X)^{-1} = \text{diag}([10, 1])$. The posterior (solid line) has mean $m = [2.73, 1]^T$ (cross) and precision $S_w^{-1} = \text{diag}([11, 2])$.

In this example, the measurements are more informative about $w(1)$ than $w(2)$. This is reflected in the posterior distribution.

Tennis

From Wolpert and Ghahramani (2006)



$$
\begin{aligned}
p(w|y) &= \mathrm{N}(m_w, S_w) \\
S_w^{-1} &= X^T C_y^{-1} X + C_w^{-1} \\
m_w &= S_w (X^T C_y^{-1} y + C_w^{-1} \mu_w)
\end{aligned}
$$

MAP Learning

The posterior density is given by Bayes rule

$$p(w|y) = \frac{p(y|w)p(w)}{p(y)}$$

The Maximum A Posterior (MAP) estimate is given by

$$\hat{w} = \arg\max_{w} p(w|y)$$

Because the maxima of $\log[x]$ is the same as the maximum of x we can also write

$$\hat{w} = \arg\max_{w} L(y, w)$$

where

$$L = \log[p(y|w)p(w)]$$

is the joint log likelihood. For Linear Gaussian models MAP parameters are equivalent to the posterior mean.

MAP Learning

Online MAP learning follows the gradient of the joint log likelihood

$$\tau \frac{dw}{dt} = \frac{dL}{dw}$$

This splits into two derivatives - one for the likelihood (shown earlier) and one for the prior. For prior mean μ_w and isotropic prior covariance $C_w = \lambda_w I_p$ we have

$$\log p(w) = -\frac{\lambda_w}{2}(w - \mu_w)^T(w - \mu_w) - \log Z$$

Hence

$$\frac{d \log p(w)}{dw} = \lambda_w(\mu - w)$$

MAP Learning

The overall MAP learning rule is

$$\tau \frac{dw}{dt} = \lambda_w(\mu_w - w_i) + \lambda_n x_n^T(y_n - x_n w)$$

For $\mu = 0$ we have the ML update plus a decay term

$$\tau \frac{dw_i}{dt} = -\lambda_w w_i + \lambda_n x_n(i)(y_n - x_n w)$$

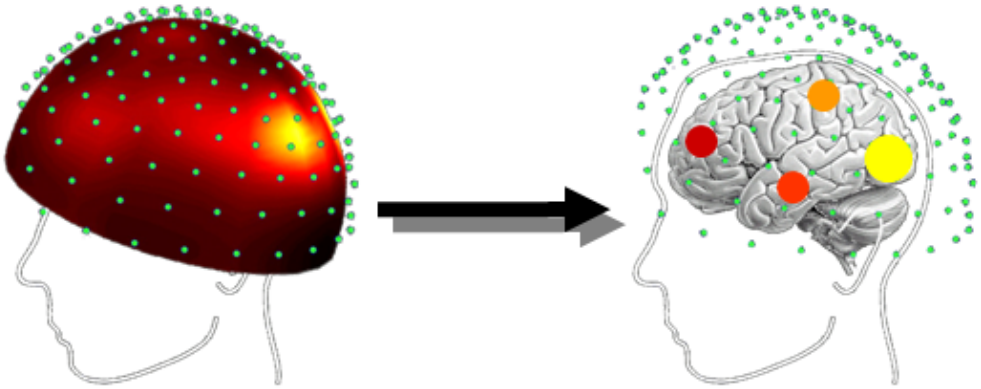
MEG Source Reconstruction

MEG Source Reconstruction is achieved through inversion of the linear model

$$y = Xw + e$$

$$(d \times 1) = (d \times p)(p \times 1) + (d \times 1)$$

for MEG data, y with d sensors and p potential sources, w, lying perpendicular to the cortical surface. The lead field matrix is specified by X. For our example we have $d = 274$ and $p = 8192$.



The above equation is for a single time point.

Generative Models

Likelihood

$$p(y|w) = \mathrm{N}(y; Xw, C_y)$$

Prior

$$p(w) = \mathrm{N}(w; 0, C_w)$$

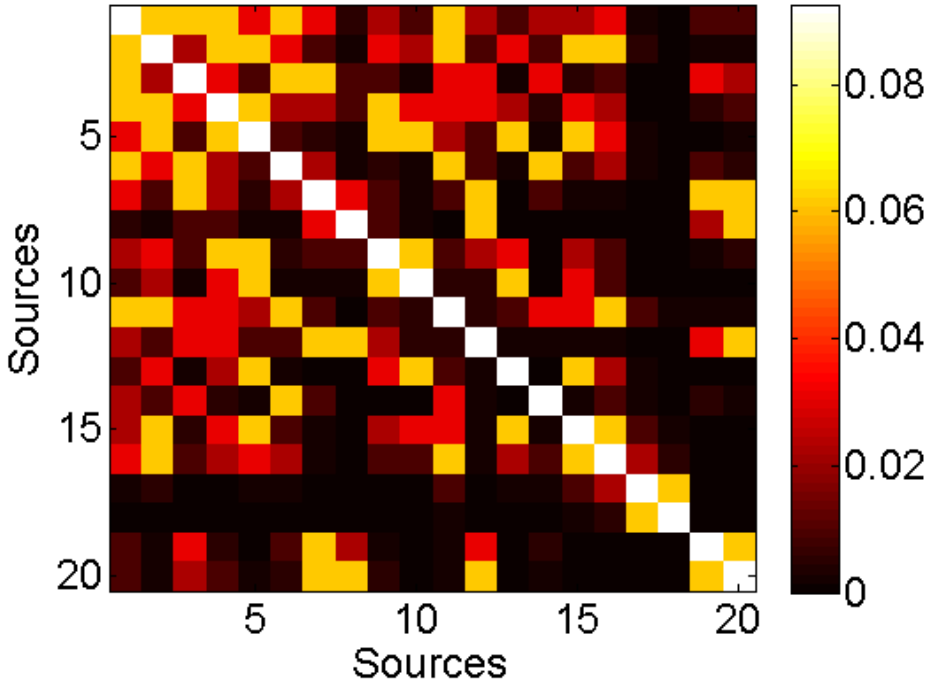
We let

$$\begin{aligned} C_y &= \lambda_1 Q_1 \\ C_w &= \lambda_2 Q_2 \end{aligned}$$

For shrinkage priors $Q_2 = I_p$, MAP estimation results in the minimum norm method of source reconstruction. This is implemented in SPM as the 'IID' option

Smoothness Priors

For smoothness priors $Q_2 = KK^T$ corresponding to the operation of a Gaussian smoothing kernel, MAP estimation results something similar to the Low Resolution Tomography (LORETA) method.



This is implemented in SPM as the 'COH' option. Note, these are not location priors.

Posterior Density

From earlier we have

$$\begin{aligned} S_w^{-1} &= X^T C_y^{-1} X + C_w^{-1} \\ m_w &= S_w X^T C_y^{-1} y \end{aligned}$$

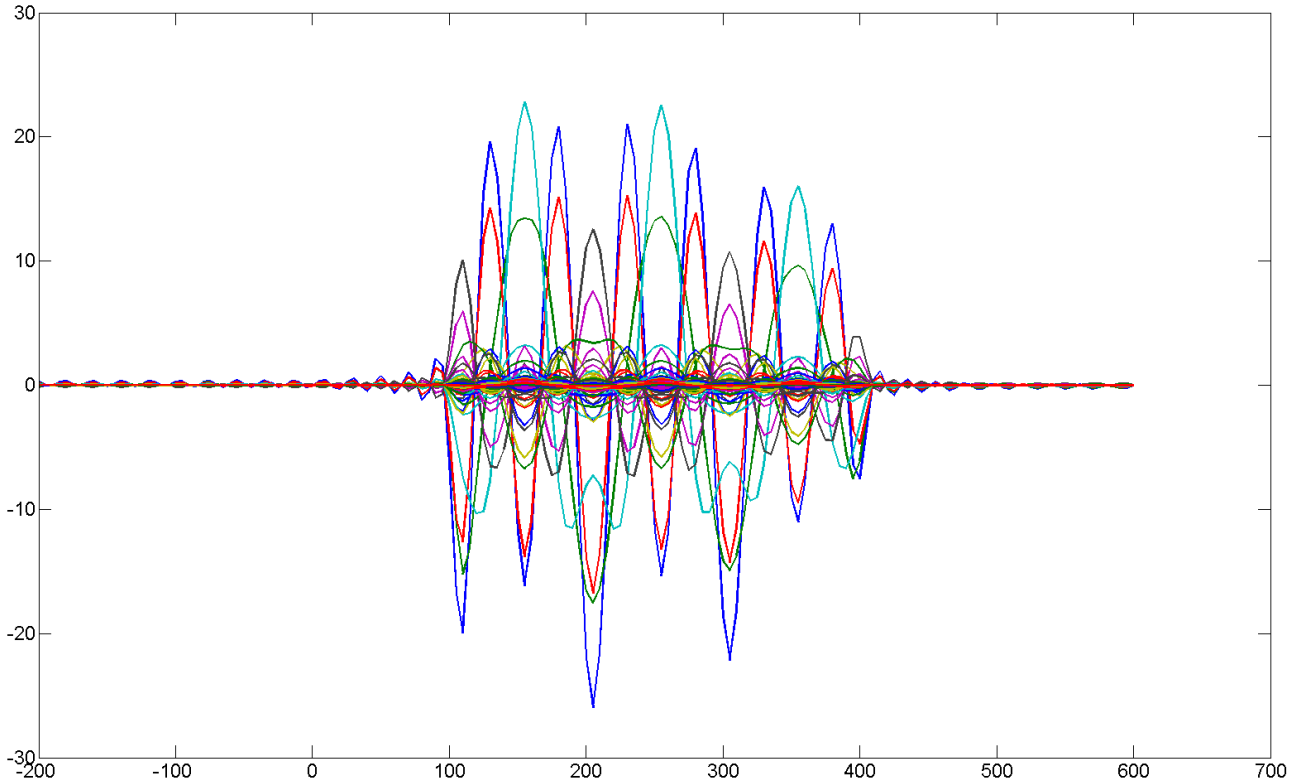
However, S_w is $p \times p$ with $p = 8192$ so cannot be inverted easily. But we can use the matrix inversion lemma, also known as the Woodbury identity (Bishop, 2006)

$$(A + BCD)^{-1} = A^{-1} - A^{-1}B(C^{-1} + DA^{-1}B)^{-1}DA^{-1}$$

to ensure that only $d \times d$ matrices need inverting.

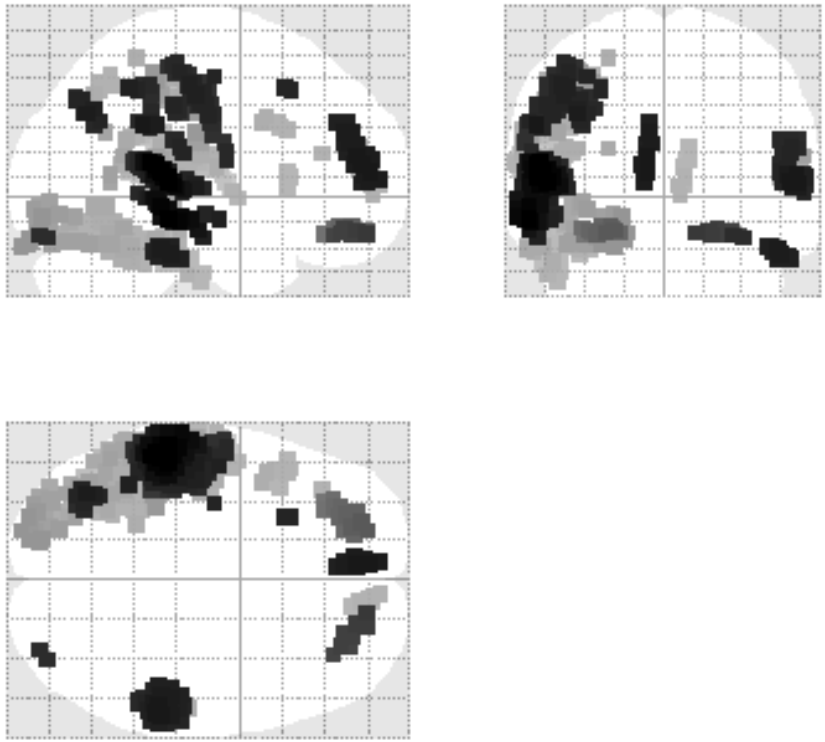
Simulation

Two sinusoidal sources were placed in bilateral auditory cortex and produced this MEG data (Barnes, 2010), comprising $d = 274$ time series (butterfly plot)



LORETA

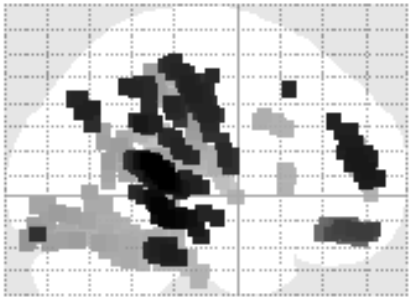
We fix $\lambda_1 = 1$. Here we set $\lambda_2 = 0.01$.

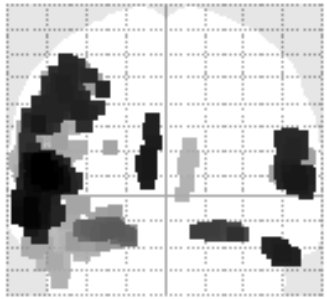


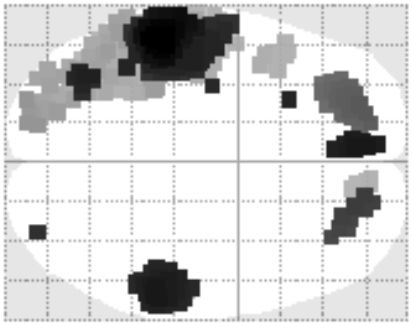
This shows the posterior mean activity for the 500 dipoles with the greatest power (over peristimulus time)

LORETA

We fix $\lambda_1 = 1$. Here we set $\lambda_2 = 0.1$.

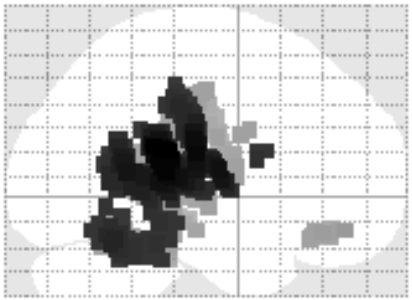


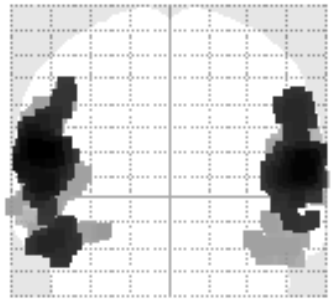


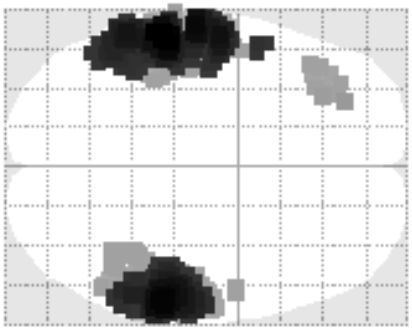


LORETA

We fix $\lambda_1 = 1$. Here we set $\lambda_2 = 1$.







Empirical Bayes

Hyperparameters, λ, can be estimated so as to maximise the model evidence. This forms the basis of Empirical Bayes.

The marginal likelihood or model evidence is given by

$$\begin{aligned} p(y|\lambda) &= \int p(y, w, \lambda) dw \\ &= \int p(y|w, \lambda) p(w|\lambda) dw \end{aligned}$$

The log model evidence is

$$L(\lambda) = \log p(y|\lambda)$$

For linear models this can be derived as in Bishop (2006) or as in my Maths for Brain Imaging notes.

In this formulation λ are not treated as random variables. There is no prior on them.

Model Evidence

The model evidence is composed of sum squared precision weighted prediction errors and Occam factors

$$\begin{aligned} L(\lambda) &= -\frac{1}{2} e_y^T C_y^{-1} e_y - \frac{1}{2} \log |C_y| - \frac{d}{2} \log 2\pi \\ &- \frac{1}{2} e_w^T C_w^{-1} e_w - \frac{1}{2} \log \frac{|C_w|}{|S_w|} \end{aligned}$$

where λ is a vector of hyperparameters that parameterise the covariances C_w and C_y. The prediction errors are the difference between what is expected and what is observed

$$\begin{aligned} e_y &= y - X m_w \\ e_w &= m_w - \mu_w \end{aligned}$$

Empirical Bayes

We iterate between finding the parameters w and hyperparameters λ. For linear Gaussian models this corresponds to computing the posterior over w

$$\begin{aligned} S_w^{-1} &= X^T C_y^{-1} X + C_w^{-1} \\ m_w &= S_w (X^T C_y^{-1} y + C_w^{-1} \mu_w) \end{aligned}$$

and then setting λ to maximise the model evidence.

$$\hat{\lambda} = \arg\max_{\lambda} L(\lambda)$$

These two steps are then iterated and can be thought of as E and M steps in an EM optimisation algorithm.

Isotropic Covariances

For a Bayesian GLM

$$\begin{aligned} y &= Xw + e_1 \\ w &= \mu_w + e_2 \end{aligned}$$

with isotropic covariances

$$\begin{aligned} C_y &= \lambda_y I_N \\ C_w &= \lambda_w I_p \end{aligned}$$

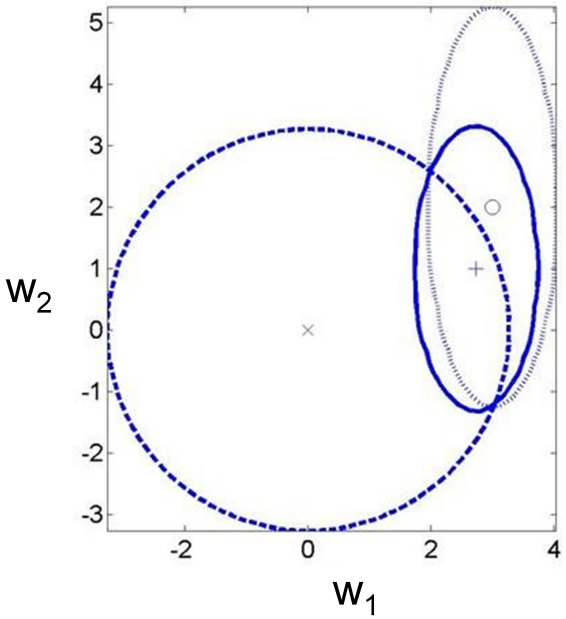
and d data points and p parameters. The equations for updating λ can be derived as shown in Chapter 10 of Bishop (2005).

Well-determined parameters

Define

$$\gamma = \sum_{j=1}^{p} \frac{\alpha_j}{\alpha_j + \hat{\lambda}_w}$$

where α_j are eigenvalues of the data precision term $X^T C_y^{-1} X$. If $\alpha_j >> \hat{\lambda}_w$ for all j then $\gamma = p$. Parameters have all been determined by the data. So γ is equivalent to number of well-determined parameters.



M-Step

Then

$$\frac{1}{\hat{\lambda}_w} = \frac{e_w^T e_w}{\gamma}$$
$$\frac{1}{\hat{\lambda}_y} = \frac{e_y^T e_y}{d-\gamma}$$

where the prediction errors are

$$e_y = y - Xm_w$$
$$e_w = m_w - \mu_w$$

This effectively partitions the degrees of freedom in the data into those for estimating the prior and the likelihood.

Setting λ to maximise the *marginal* likelihood produces unbiased estimates of variances whereas ML estimation produces biased estimates.

Linear Covariances

For a Bayesian GLM

$$\begin{aligned} y &= Xw + e_1 \\ w &= \mu_w + e_2 \end{aligned}$$

with covariances

$$\begin{aligned} C_y &= \sum_i \lambda_i Q_i \\ C_w &= \sum_{i'} \lambda_{i'} Q_{i'} \end{aligned}$$

where Q are known covariance basis functions. The M-step is

$$\hat{\lambda} = \arg\max_{\lambda} L(\lambda)$$

Gradient Ascent

This maximisation is effected by first computing the gradient and curvature of $L(\lambda)$ at the current parameter estimate, λ^{old}

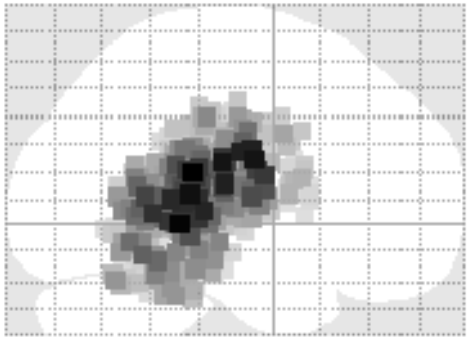
$$
\begin{aligned}
j_\lambda(i) &= \frac{dL(\lambda)}{d\lambda(i)} \\
H_\lambda(i,j) &= \frac{d^2L(\lambda)}{d\lambda(i)d\lambda(j)}
\end{aligned}
$$

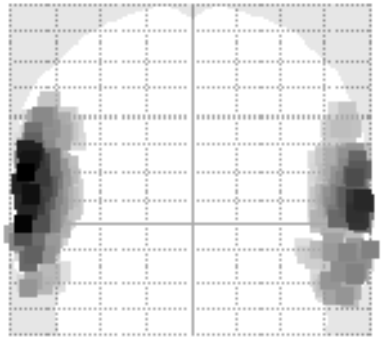
where i and j index the ith and jth parameters, j_λ is the gradient vector and H_λ is the curvature matrix. The new estimate is then given by

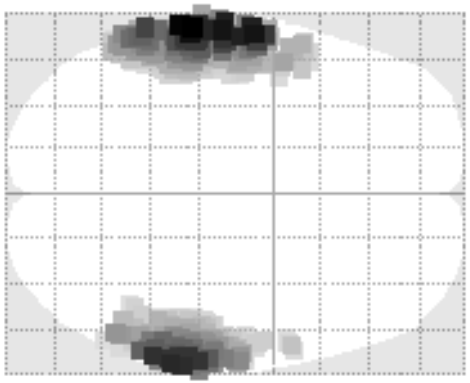
$$\lambda^{new} = \lambda^{old} - H_\lambda^{-1} j_\lambda$$

MEG Source Reconstruction

Hyperparameters set using Empirical Bayes.



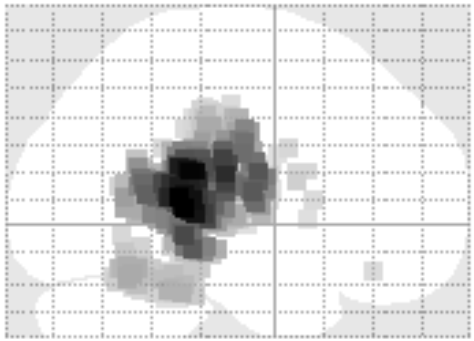


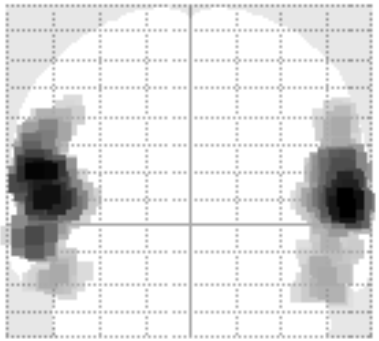


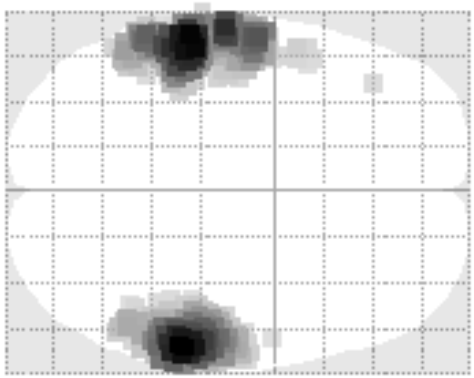
The *minimum norm* method, also implemented in SPM as the IID option.

Smoothness Priors

Hyperparameters set using Empirical Bayes.

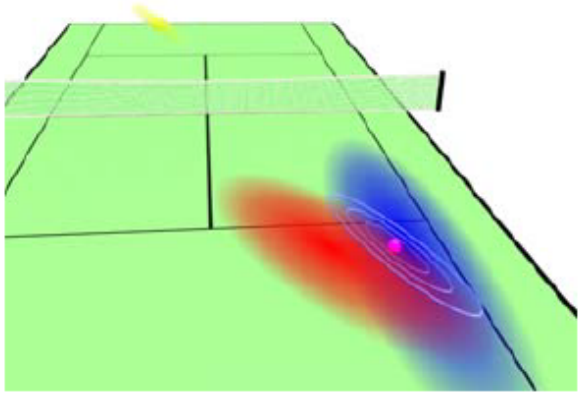






This is similar to the LORETA method, implemented in SPM as the COH option.

Restricted Maximum Likelihood



The posterior over w

$$
\begin{aligned}
S_w^{-1} &= X^T C_y^{-1} X + C_w^{-1} \\
m_w &= S_w (X^T C_y^{-1} y + C_w^{-1} \mu_w)
\end{aligned}
$$

can also be written in a more compact form.

Augmented Form

This compact form is

$$\begin{aligned} S_w^{-1} &= \bar{X}^T V^{-1} \bar{X} \\ m_w &= S_w(\bar{X}^T V^{-1} \bar{y}) \end{aligned}$$

where

$$\begin{aligned} \bar{X} &= \begin{bmatrix} X \\ I_p \end{bmatrix} \\ V &= \begin{bmatrix} C_y & 0 \\ 0 & C_w \end{bmatrix} \\ \bar{y} &= \begin{bmatrix} y \\ \mu_w \end{bmatrix} \end{aligned}$$

where we've augmented the data matrix with prior expectations; $\bar{y}$ is $(d+p) \times 1$ and $\bar{X}$ is $(d+p) \times p$.

Augmented Form

Estimation in a Bayesian GLM is therefore equivalent to Maximum Likelihood estimation (ie. for IID covariances this is the same as Weighted Least Squares) with *augmented* data.

$$m_w = (\bar{X}^T V^{-1} \bar{X})^{-1} \bar{X}^T V^{-1} \bar{y}$$

Prior beliefs can be thought of as extra data points.

Model Evidence

The previous expression for the model evidence

$$\begin{aligned} L(\lambda) &= -\frac{1}{2} e_y^T C_y^{-1} e_y - \frac{1}{2} \log |C_y| - \frac{N_y}{2} \log 2\pi \\ &- \frac{1}{2} e_w^T C_w^{-1} e_w - \frac{1}{2} \log \frac{|C_w|}{|S_w|} \end{aligned}$$

can now be written more compactly

$$\begin{aligned} L(\lambda) &= -\frac{1}{2} \bar{e}^T V^{-1} \bar{e} - \frac{1}{2} \log |V| - \frac{N_y}{2} \log 2\pi \\ &+ \frac{1}{2} \log |S_w| \end{aligned}$$

where the overall prediction errors are

$$\bar{e}^T = [e_y^T, e_w^T]$$

Restricted Maximum Likelihood

If we eliminate m_w and S_w from the model evidence equation we end up with the Restricted Maximum Likelihood (ReML) objective function.
Substituting for S_w gives

$$\begin{aligned} L(\lambda) &= -\frac{1}{2}\bar{e}^T V^{-1}\bar{e} - \frac{1}{2}\log|V| - \frac{N_y}{2}\log 2\pi \\ &- \frac{1}{2}\log|\bar{X}^T V^{-1}\bar{X}| \end{aligned}$$

where

$$\bar{e} = \bar{y} - \bar{X}m_w$$

Restricted Maximum Likelihood

$$\begin{aligned}\bar{e} &= \bar{y} - \bar{X} m_w \\ &= \bar{y} - \bar{X} S_w \bar{X}^T V^{-1} \bar{y} \\ &= \bar{y} - \bar{X} (\bar{X}^T V^{-1} \bar{X})^{-1} \bar{X}^T V^{-1} \bar{y} \\ &= R\bar{y}\end{aligned}$$

where R is called the residual-forming matrix

$$R = I - \bar{X} (\bar{X}^T V^{-1} \bar{X})^{-1} \bar{X}^T V^{-1}$$

Hence

$$\begin{aligned}\bar{e}^T V^{-1} \bar{e} &= \bar{y}^T R^T V^{-1} R \bar{y} \\ &= \mathrm{Tr}(V^{-1} R \bar{y} \bar{y}^T R^T)\end{aligned}$$

Restricted Maximum Likelihood

The Restricted Maximum Likelihood (ReML) objective function is therefore

$$\begin{aligned} L(\lambda) &= -\frac{1}{2}\mathrm{Tr}(V^{-1}R\bar{y}\bar{y}^T R^T) - \frac{1}{2}\log|V| - \frac{N_y}{2}\log 2\pi \\ &- \frac{1}{2}\log|\bar{X}^T V^{-1}\bar{X}| \end{aligned}$$

This only depends on $\bar{X}$, V and $\bar{y}\bar{y}^T$. This can also be used for nonaugmented matrices. This function is optimised in SPM's ReML function (Friston et al, 2002)

References

G. Barnes (2010) MEG Source Localisation, SPM Manual, Chapter 35

C. Bishop (1995) Neural Networks for Pattern Recognition. OUP.

K. Friston et al. (2002) Neuroimage (16), 465-483

W. Penny, J Kilner and F.Blankenburg (2007) Neuroimage 36, 661-671.

W. Press et al (1988) Numerical Recipes. Cambridge.

SPM Manual. http://www.fil.ion.ucl.ac.uk/spm/doc/

B. Widrow and M. Hoff (1960) IRE WESCON Convention Record, 96-104, New York.

D. Wolpert and Z. Ghahramani (2004) In Gregory RL (ed) Oxford Companion to the Mind, OUP.