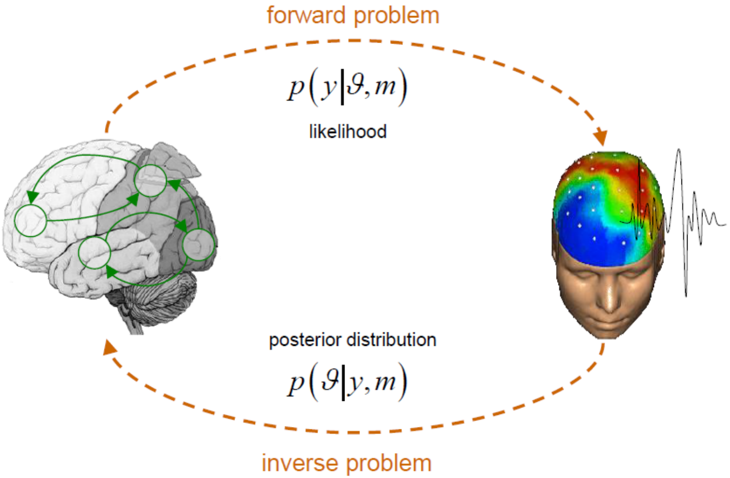
Bayesian Inference for Brain Connectivity Modelling

Will Penny, UCL.

Sussex University, January 25th, 2016.

Introduction



forward problem

$$p(y|\vartheta, m)$$

likelihood

posterior distribution

$$p(\vartheta|y, m)$$

inverse problem

Cortical Units

Jansen and Rit (Biol Cybernetics, 1995), building on the work of Lopes Da Sliva and others, developed a biologically inspired model of EEG activity using Neural Masses.

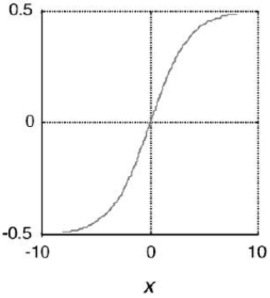
It models a cortical unit with three subpopulations of cells

- Stellate cells with average membrane potential v_s and current c_s.
- Pyramidal cells with average membrane potential v_p and current c_p.
- Inhibitory interneurons with average membrane potential v_i and current c_i.

Firing Rate Curves

Membrane potentials are transformed into firing rates via sigmoidal functions

$$s(x) = \frac{1}{1+\exp(-rx)} - \frac{1}{2}$$



Alpha Function Synapses

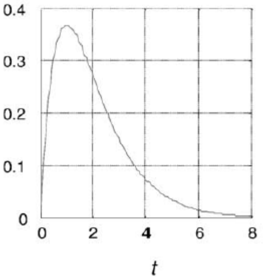
Firing rates cause postsynaptic potentials via convolutions with alpha function synaptic kernels

$$v_{out}(t) = h_e(t) \otimes s(v_{in})$$

where

$$h_e(t) = \frac{H_e}{\tau_e} t \exp(-t/\tau_e)$$

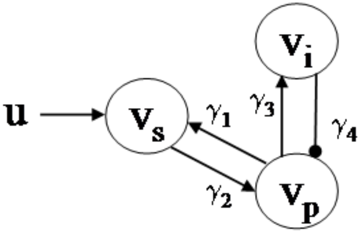
Similarly for inhibitory synapses with $h_i(t)$, H_i, τ_i.



Inhibitory Interneurons

The inhibitory interneurons receive excitatory input from the pyramidal cells

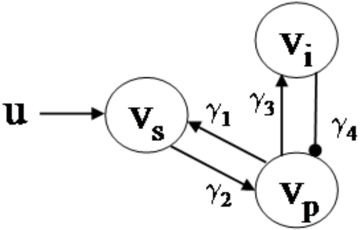
$$v_i = \gamma_3 s(v_p) \otimes h_e$$



Stellate Cells

The stellate cells receive external input from thalamus or other cortical regions and excitatory feedback from pyramidal cells

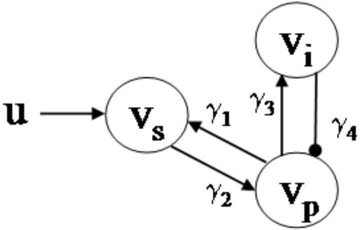
$$v_s = (s(u) + \gamma_1 s(v_p)) \otimes h_e$$



Pyramidal Cells

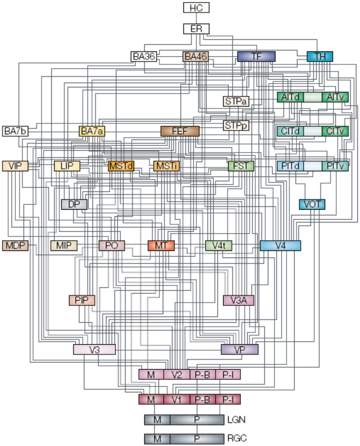
The pyramidal cells receive excitatory input from stellate cells and inhibitory input from interneurons. This produces both excitatory v_{pe} and inhibitory v_{pi} postsynaptic potentials.

$$
\begin{aligned}
v_{pe} &= \gamma_2 s(v_s) \otimes h_e \\
v_{pi} &= \gamma_4 s(v_i) \otimes h_i \\
v_p &= v_{pe} - v_{pi}
\end{aligned}
$$



Brain Connectivity

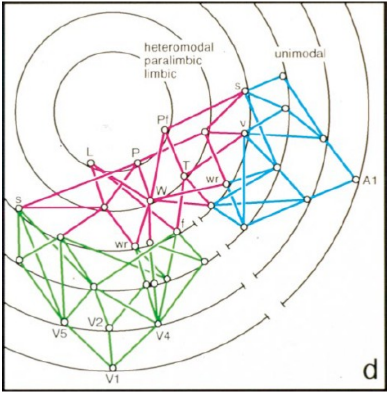
Cortex is organised hierarchically with higher level regions processing more abstract features and lower levels more concrete ones.



Felleman and Van Essen, Cerebral Cortex, 1991

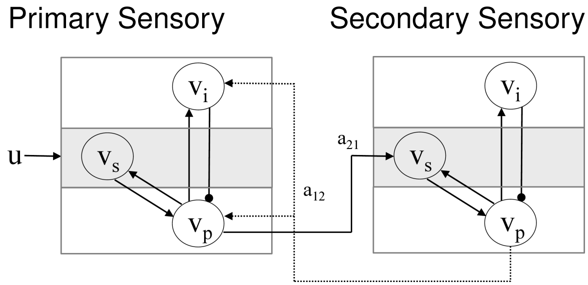
Brain Connectivity

Multiple, parallel, convergent hierarchies with information flow both towards and away from senses.



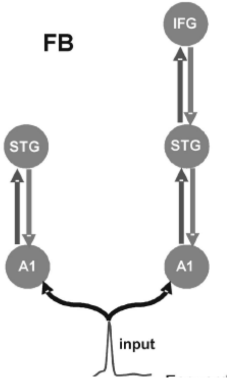
Mesulam, Brain, 1999

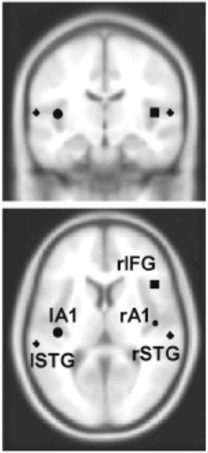
Connecting Cortical Units



David et al. Neuroimage, 2006 proposed connecting neural mass units together according to the Felleman and Van-Essen connection rules.

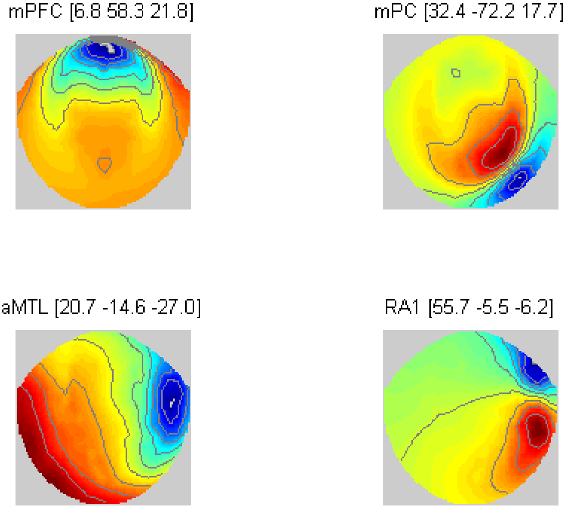
Brain Connectivity Model





Garrido et al. PNAS, 2007

MEG/EEG Forward Model



mPFC [6.8 58.3 21.8]

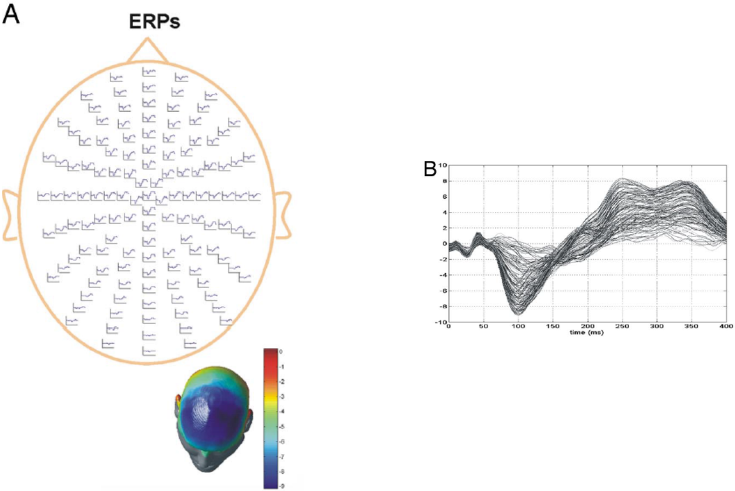
mPC [32.4 -72.2 17.7]

aMTL [20.7 -14.6 -27.0]

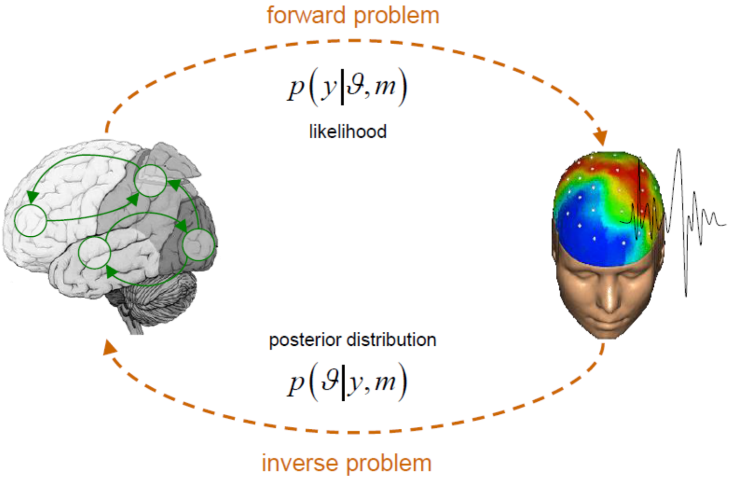
RA1 [55.7 -5.5 -6.2]

MEG/EEG data

Event-Related Fields/Potentials y.



Bayesian Inference



forward problem

$$p\left(y \middle| \vartheta, m\right)$$

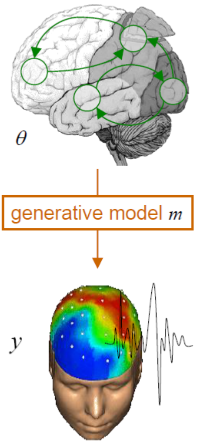
likelihood

posterior distribution

$$p\left(\vartheta \middle| y, m\right)$$

inverse problem

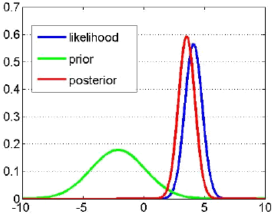
Bayesian Inference



Likelihood: $p(y|\theta,m)$

Prior: $p(\theta|m)$

Bayes rule: $p(\theta|y,m)=\dfrac{p(y|\theta,m)\,p(\theta|m)}{p(y|m)}$



Bayesian Inference

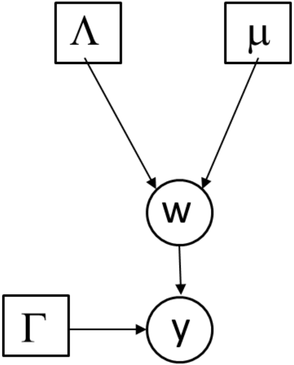
MEG/EEG data, y.

Likelihood $p(y|w,\Gamma)$ where Γ is a covariance matrix specifying observation noise.

We have a Gaussian prior over model parameters

$$p(w|\mu,\Lambda) = N(w;\mu,\Lambda)$$

with known prior mean μ and precision Λ. This captures our prior knowledge about likely range of synaptic time constants.



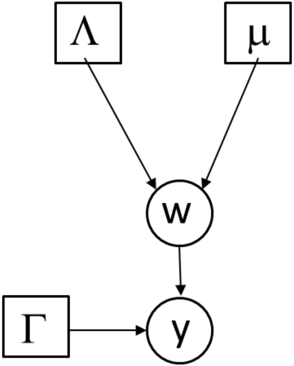
Bayesian Inference

Want **posterior distribution** over parameters $p(w|y)$ to make inferences about connection strengths.

Want **model evidence** $p(y|m)$ so we can use Bayes rule over models

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

to find out e.g. how many cortical sources there are, what is the best model of a cortical unit, what is the connectivity structure of the network.



" Although this may seem a paradox, all exact science is dominated by the idea of approximation" - Bertrand Russell.

Variational Inference

Assume an approximate posterior distribution that factorises among chosen grouping of unknown parameters. For example

$$q(w, \Gamma | y) = q(w|y) q(\Gamma | y)$$

assumes posterior independence between connectivity parameters and observation noise parameters.

Minimise Kullback-Liebler (KL) divergence between approximate $q(w, \Gamma|y)$ and true posterior $p(w, \Gamma|y)$.

This is equivalent to maximising a lower bound (F, the negative variational free energy) on the model evidence $p(y|m)$, where m indexes model assumptions.

M Beal, PhD Thesis, UCL, 2003

Variational Laplace

Additionally assume that each factorised density is a Gaussian

$$\begin{aligned} q(w|y) &= N(w; m_w, S_w) \\ q(\Gamma|y) &= \prod_i N(\log \Gamma_{ii}; m_\Gamma(i), S_\Gamma(i)) \end{aligned}$$

Minimise KL divergence by finding the moments of the approximate posterior density (m_w, S_w, m_Γ, S_Γ) that maximise F.

We can also use F as a model selection criterion.

Friston et al. Neuroimage, 2007.

Variational Laplace

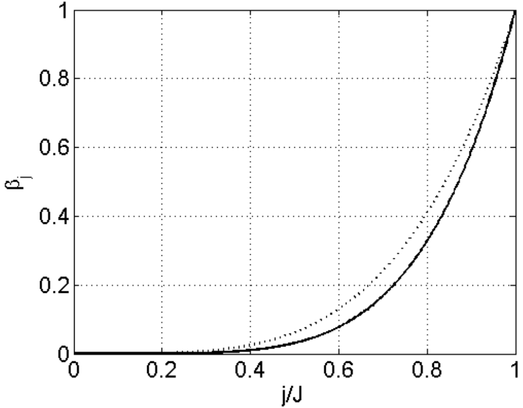
In practice F is maximised using a local gradient-based search method making VL very fast.

- Local optimisation based on gradients and curvatures
- Posterior assumed Gaussian
- Provides model evidence estimate
- Very fast

Annealed Importance Sampling

For the *j*th temperature the algorithm produces a sample from

$$f_j(w) = p(y|w)^{\beta_j} p(w)$$



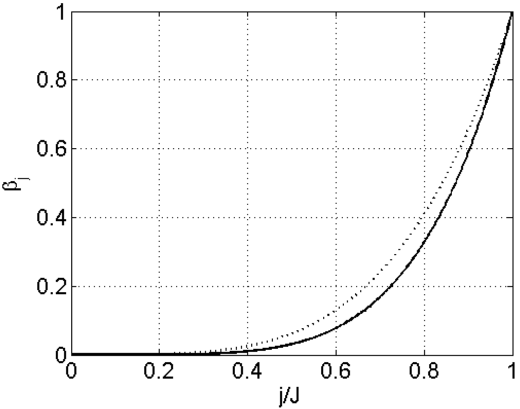
Sample from prior at $\beta = 0$ and posterior at $\beta = 1$.

Neal, Statistics and Computing, 2001.

Annealed Importance Sampling

For the *j*th temperature the algorithm produces a sample from

$$f_j(w) = p(y|w)^{\beta_j} p(w)$$



Inverse temperatures β_j with $j = 0..J$, $\beta_0 = 0$ and $\beta_J = 1$.
Geometric schedule $\beta_j = (j/J)^5$ (solid), $\beta_j = (j/J)^4$ (dotted).

Annealed Importance Sampling

An independent sample $w^{(i)}$ from the posterior density is produced by generating a sequence of points $w_1, w_2, ...w_J$ as follows

- Generate w_1 from $p(w)$
- Generate w_2 from w_1 using $T_1(w_2|w_1)$
- ...
- Generate w_j from w_{j-1} using $T_{j-1}(w_j|w_{j-1})$
- ...
- Generate w_J from w_{J-1} using $T_{J-1}(w_J|w_{J-1})$

and then let $w^{(i)} = w_J$. We refer to the process of producing a single independent sample as a ‘trajectory’.

We are using Langevin Monte Carlo (LMC) for the T_j’s.

Langevin Monte Carlo (LMC)

Given log joint and its gradient as a function of w

$$\begin{aligned} f_j(w) &= p(y|w,\Gamma)^{\beta_j} p(w|\mu,\Lambda) \\ L_j(w) &= \beta_j \log p(y|w,\Gamma) + \log p(w|\mu,\Lambda) \\ g_j(w) &= \frac{dL_j(w)}{dw} \end{aligned}$$

the LMC Proposal is drawn as

$$\begin{aligned} w_j^* &\sim p(w_j^*|w_{j-1}) \\ p(w_j^*|w_{j-1}) &= \mathcal{N}(w_j^*; m_j, C_j) \\ m_j &= w_{j-1} + \frac{1}{2} C_j g_j(w_{j-1}) \\ C_j &= h^2 \left(\Lambda + \beta_j S^T \Gamma S\right)^{-1} \end{aligned}$$

where S is a sensitivity matrix

$$S(i,k) = \frac{dy(i)}{dw_s(k)}$$

Langevin Monte Carlo

The proposal is accepted using the standard Metropolis-Hastings probability

$$a = \frac{f_j(w_j^*)}{f_j(w_{j-1})} \frac{p(w_{j-1}|w_j^*)}{p(w_j^*|w_{j-1})}$$

The proposal is always accepted if $a > 1$.

If the step is accepted we set $w_j = w_j^*$. If it is rejected we set $w_j = w_{j-1}$.

The second term above ensures reversibility, and in principle that we visit all of parameter space in proportion to its (posterior) probability.

Girolami and Calderhead, J Roy Stat Soc B, 2011.

Annealed Importance Sampling

The above process is repeated $i = 1..I$ times to produce I independent samples from the posterior density.

Because the samples are produced independently, without interaction among trajectories, the AIS algorithm is amenable to ‘embarrassing parallelization’

We need not concern ourself with within-trajectory correlation (as e.g. Hamiltonian Monte Carlo does) as we’re only taking one sample from each

Effectively, AIS is a multistart algorithm, that has a principled way of combining information from multiple starts/trajectories

Annealed Importance Sampling

Each sample is also accompanied by an importance weight

$$v^{(i)} = \frac{f_1(w_1)}{f_0(w_1)} \frac{f_2(w_2)}{f_1(w_2)} \frac{f_3(w_3)}{f_2(w_3)} \cdots \frac{f_J(w_J)}{f_{J-1}(w_J)}$$

which can be evaluated as

$$\log v^{(i)} = \sum_{j=1}^{J} (\beta_j - \beta_{j-1}) \log p(y|w_j)$$

The importance weights, or average of them, provide an approximation to the model evidence.

AIS is highly efficient as every sample contributes to the model evidence estimate.

VL versus AIS

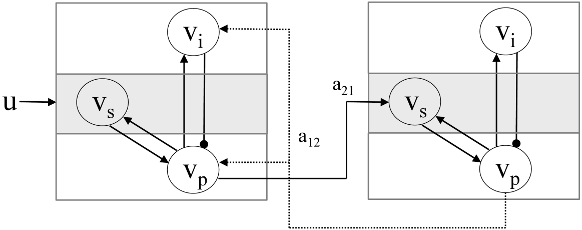
Variational Laplace (VL):

- Local optimisation based on grad and curve
- Provides model evidence estimate
- Posterior assumed Gaussian
- Will not avoid local maxima
- Very fast

Annealed Importance Sampling (AIS):

- LMC uses grad and curve for proposals
- Provides model evidence estimate
- Posterior not assumed Gaussian
- More likely to avoid local maxima
- Slow, but maps perfectly onto multi-core
- Test parametric assumptions of VL

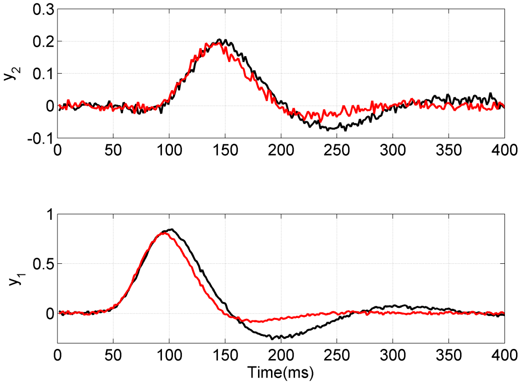
Neural Masses



We estimate a 10-dimensional parameter vector w. These are between-region connex, a_{12}, a_{21}, between region delays δ_{12}, δ_{21}, within-region connex $\gamma_{1..4}$ and parameters of firing rate function r_1, r_2.

Two region model

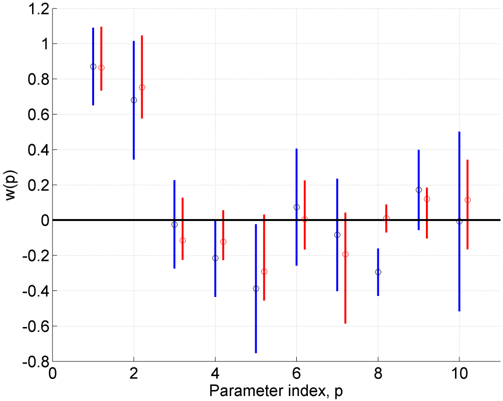
Observed time series, y_2, is pyramidal cell activity in higher-level region.



Impulse of activity, u, at $t = 0$ produces observed time series, y_1, being pyramidal cell activity in lower-level (sensory) region.

Parameter Estimates

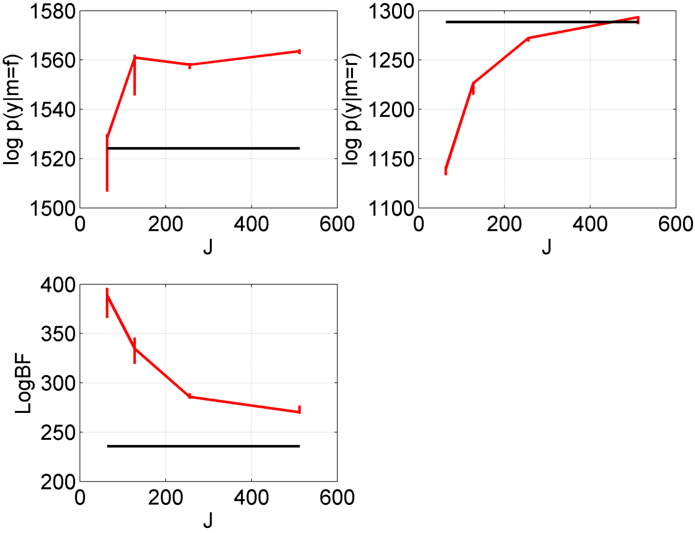
With 95% confidence intervals. AIS (red) VL (blue).



True parameters are all zero except first two.

Model Evidence

Vary resolution of annealing schedule



AIS (red), VL (black)

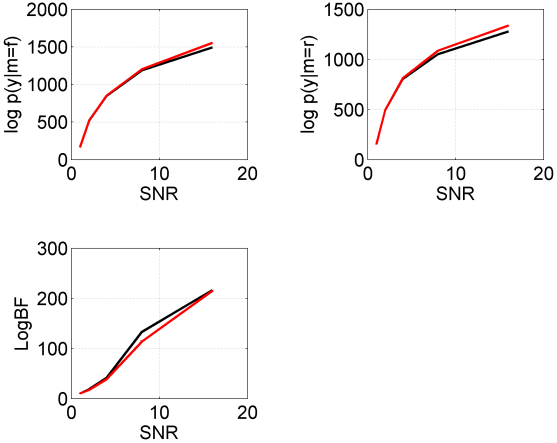
Evidence, Bayes Factors, Compute Time

Model	Estimate		Time(s)	
	VL	AIS	VL	AIS
Linear, LogEv, Full	-11.02	-11.00	0.005	15.4
Linear, LogEv, Red	-23.97	-23.94	0.002	3.1
Linear, LogBF	12.95	12.94	-	-
Approach, LogEv, Full	-73.88	-73.77	0.58	19.4
Approach, LogEv, Red	-783.62	-783.61	0.02	2.9
Approach, LogBF	709.74	709.84	-	-
Neural Mass, LogEv, Full	1524.1	1563.6	22	5290
Neural Mass, LogEv, Red	1288.4	1293.4	24	4610
Neural Mass, LogBF	235.74	270.2	-	-

AIS estimates from $I = 32$ samples and $J = 512$ trajectories.
The linear model VL results are for analytic solution.

Effect of SNR

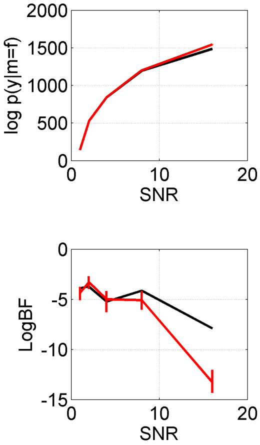
True model has full connectivity. AIS (red), VL (black).



AIS and VL are always in agreement in favouring the true model.

Effect of SNR

True model has reduced connectivity. AIS (red), VL (black).



AIS and VL are always in agreement in favouring the true model.

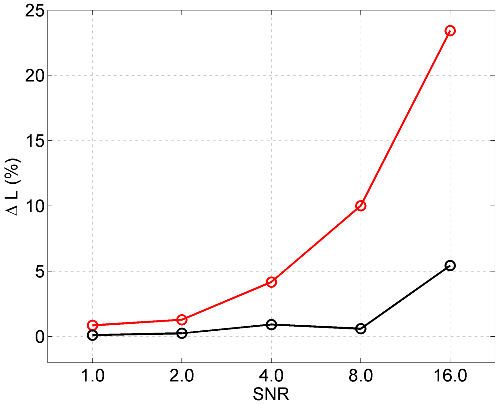
Gaussianity

Table: *p-values from Royston's Gaussianity test applied to AIS samples from 'Full' NMM.*

SNR	32 Trajectories		64 Trajectories	
	Full	Reduced	Full	Reduced
1	0.02	0.18	0.02	0.03
2	0.40	0.10	0.74	0.42
4	0.80	0.54	0.07	0.29
8	0.33	0.51	0.004	0.02
16	0.40	0.17	0.02	$5 \text{ x } 10^{-4}$

AIS versus Multistart VL

Baseline log joint, L, is from single default VL (start from prior mean). We are then plotting percentage improvement in this.



AIS (red) finds better parameters than best Multistart VL (black). They are matched for computer time.

Summary

- LMC explores local parameter space using gradients and curvatures
- Embed this in AIS for global Bayesian optimisation
- Better parameter estimates than Multistart VL
- AIS provides an estimate of the model evidence
- PAM and PHM are special cases of AIS
- Test parametric assumptions of VL

But its slow (about 80 mins per Neural Mass Model)

- Anneal from posterior of full model to posterior of reduced (or other) model to compute Bayes Factor
- Automatically tune annealing schedules whilst preserving parallelisation

UCL wellcometrust

Work with Biswa Sengupta @ UCL.

Annealed Importance Sampling

We define the normalising constant at each temperature as

$$\begin{aligned} Z_j &= \int f_j(w) dw \\ &= \int p(y|w,m)^{\beta_j} p(w|m) dw \end{aligned}$$

We then have

$$\begin{aligned} Z_1 &= \int p(w|m) dw = 1 \\ Z_J &= \int p(y|w,m) p(w|m) dw \\ &= p(y|m) \end{aligned}$$

Annealed Importance Sampling

Therefore

$$\begin{aligned} p(y) &= \frac{Z_J}{Z_1} \\ &= \frac{Z_2}{Z_1}\frac{Z_3}{Z_2}\cdots\frac{Z_J}{Z_{J-1}} \\ &= \prod_{j=1}^{J-1} r_j \end{aligned}$$

where $r_j = Z_{j+1}/Z_j$. We can then write

$$\begin{aligned} r_j &= \frac{1}{Z_j}\int f_{j+1}(w)dw \\ &= \int \frac{f_{j+1}(w)}{f_j(w)}\frac{f_j(w)}{Z_j}dw \\ &\approx \frac{1}{N}\sum_{n=1}^{N}\frac{f_{j+1}(w_n)}{f_j(w_n)} \end{aligned}$$

where the last line indicates a Monte-Carlo approximation of the integral with samples w_n drawn from the distribution at temperature β_j. This can in turn be written as

$$r_j = \frac{1}{N}\sum_{n=1}^{N} p(y|w_n, m)^{\beta_{j+1}-\beta_j}$$

For $N = 1$ this equals the importance weight.

Reverse Annealing

By inverting the equation for the model evidence we have

$$
\begin{aligned}
\frac{1}{p(y|m)} &= \frac{Z_1}{Z_J} \\
&= \frac{Z_{J-1}}{Z_J} \cdots \frac{Z_2}{Z_3} \cdots \frac{Z_1}{Z_2} \\
&= \prod_{j=1}^{J-1} \frac{1}{r_j}
\end{aligned}
$$

Importance weights for reverse annealing are given by

$$
v^{(i)} = \frac{f_{J-1}(w_{J-1})}{f_J(w_{J-1})} \cdots \frac{f_2(w_2)}{f_3(w_2)} \frac{f_1(w_1)}{f_2(w_1)}
$$

and a series of samples $w_J, w_{J-1}, \ldots w_2, w_1$ are created by starting with w_J from forward annealing, and generating the others sequentially using LMC.

Reverse Annealing

For $J = 2$ temperatures $\beta_2 = 1$, $\beta_1 = 0$ we get

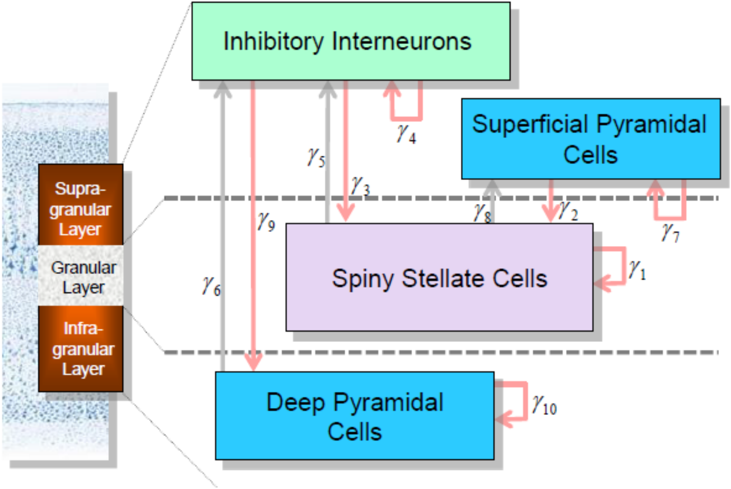
$$\frac{1}{p(y|m)} = \frac{1}{p(y|w, m)}$$

Averaging over multiple trajectories gives

$$\frac{1}{p(y|m)} = \frac{1}{I} \sum_{i=1}^{I} \frac{1}{p(y|w_i, m)}$$

which shows that the PHM approximation to the model evidence is a special case of AIS with a reverse annealing schedule and only $J = 2$ temperatures.

Canonical Microcircuit Model



Moran et al, J Neuroscience, 2013