Parameter estimation for the gestalt model

1 Generative model

A gestalt, a perceptual object, is characterised by a covariance component for the joint distribution of visual neural activity.

$$p(v \mid g) = \mathcal{N}(v; 0, C_v) \tag{1}$$

$$C_v = \sum_{k=1}^{K} g_k C_k \tag{2}$$

where K is the fixed number of possible gestalts in the visual scene and g_k is the strength of the gestalt number k, coming from a K-dimensional Gamma prior distribution with shape and scale parameters α_g and θ_g controlling the sparsity of the prior.

$$p(g) = \mathrm{Gam}(g; \alpha_g, \theta_g) \tag{3}$$

The global contrast of the image patch is encoded by a scalar variable z, also coming from a Gamma prior

$$p(z) = \mathrm{Gam}(z; \alpha_z, \theta_z) \tag{4}$$

The pixel intensities are generated from the neural activity through a set of linear projective field models, possibly Gabor filters, A, scaled by the contrast and adding some independent observational noise.

$$p(x \mid v, z) = \mathcal{N}(x; zAv, C_x) \tag{5}$$

$$C_x = \sigma_x I; \tag{6}$$

We might assume that a single composition of gestalts, characterised by the g vector, generates a batch of B independent images, described by cellular activities $V = \{v_1 \dots v_B\}$ and observations $X = \{x_1 \dots x_B\}$. Then the following likelihood distributions hold

$$p(V \mid g) = \prod_{b=1}^{B} \mathcal{N}(v_b; 0, C_v) \tag{7}$$

$$p(X \mid V) = \prod_{b=1}^{B} \mathcal{N}(x_b; zAv_b, C_x) \tag{8}$$

2 Gibbs sampling as the E-step

An efficient way to collect samples from the joint posterior over all hidden variables is to employ a Gibbs sampling scheme, where we sample from the conditional posteriors. The first is over v, and can be defined as follows

$$p(v \mid x, g, z) = \frac{p(x \mid v, z, g)p(v \mid z, g)}{p(x \mid z, g)} = \frac{\mathcal{N}(x; zAv, \sigma_x I)\mathcal{N}(v; 0, C_v)}{\int_{-\infty}^{\infty} \mathcal{N}(x; zAv, \sigma_x I)\mathcal{N}(v; 0, C_v)\mathrm{d}v} \tag{9}$$

The Gaussian over x can be rewritten to a Gaussian over v times a constant c_1 in the following way

$$\begin{aligned} \mathcal{N}(x; zAv, \sigma_x I) = c_1 \mathcal{N}(v; \frac{1}{z}(A^T A)^{-1} A^T x, \frac{\sigma_x}{z^2}(A^T A)^{-1}) = \\ = c_1 \mathcal{N}(v; \frac{1}{z} A^+ x, \frac{\sigma_x}{z^2}(A^T A)^{-1}) \end{aligned} \tag{10}$$

where A^+ is the Moore-Penrose pseudoinverse of A, with $D_x = D_v \rightarrow A^+ = A^{-1}$. Consequently, the product of two Gaussians in the numerator of Eq. 9 can also be written as a Gaussian over v introducing a new constant

$$\mathcal{N}(x; zAv, \sigma_x I)\mathcal{N}(v; 0, C_v) = c_1 c_2 \mathcal{N}(v; \mu_{post}, C_{post}) \tag{11}$$

The denominator of Eq. 9 is the integral of this formula, which evaluates to $c_1 c_2$, as the Gaussian integrates to one. This cancels the constants in the numerator, making the conditional posterior equal to the combined Gaussian over v, which, after expanding μ_{post} and C_{post}, is

$$p(v \mid x, g, z) = \mathcal{N}\left(v; \frac{z}{\sigma_x}\left(\frac{z^2}{\sigma_x} A^T A + C_v^{-1}\right)^{-1} A^T x, \left(\frac{z^2}{\sigma_x} A^T A + C_v^{-1}\right)^{-1}\right) \tag{12}$$

and for a batch of size B

$$p(V \mid X, g, z) = \prod_{b=1}^{B} \mathcal{N}\left(v_b; \mu_{post}(x_b), C_{post}\right) \tag{13}$$

which can be sampled directly from a Gaussian of dimension $D_v \times B$.

The conditional posterior over g is defined as follows

$$p(g \mid X, V, z) = \frac{p(X \mid g, V, z)p(g \mid V, z)}{p(X \mid V, z)} = \frac{p(V \mid g)p(g)}{p(V)} \tag{14}$$

which can be sampled by an MCMC sampling scheme with the following target

$$\log p(g \mid X, V) \sim -\frac{1}{2}\left[B \log(\det(C_v)) + \sum_{b=1}^{B} v_b^T C_v^{-1} v_b\right] + (\alpha - 1)\sum_{k=1}^{K} \log(g_k) \tag{15}$$

We can cycle over elements of g with a Gibbs sampling scheme too. The unnormalised conditionals, assuming an independent (e.g. Gamma) prior look as follows

$$\begin{aligned} \log p(g_j \mid g_{\neg j}, X, V) = \frac{p(V \mid g_j, g_{\neg j}, X)p(g_j \mid g_{\neg j}, X)}{p(V \mid g_{\neg_j}, X)} = \\ = \frac{p(V \mid g)p(g_j)}{p(V \mid g_{\neg_j})} \sim p(V \mid g)p(g_j) \end{aligned} \tag{16}$$

These one-dimensional targets have to be sampled by MCMC too.

The conditional posterior over z will look the following

$$p(z \mid X, V, g) = \frac{p(X \mid g, z, V)p(z \mid V, g)}{p(X \mid V, g)} \sim p(X \mid z, V)p(z) \tag{17}$$

the log-posterior being

$$\log p(z \mid X, V) \sim -\frac{1}{2}\left[BD_x \log(\sigma_x) + \frac{1}{\sigma_x}\sum_{b=1}^{B}(x_b - zAv_b)^T(x_b - zAv_b)\right] + \log p(z) \tag{18}$$

also to be sampled by MCMC.

3 Hamiltonian sampling as the M-step

The complete posterior:

$$p(v, g, z \mid x) \sim p(x \mid v, g, z)p(v \mid g)p(g)p(z) \tag{19}$$

$$\begin{aligned} \log p(v, g, z \mid x) \sim -\frac{1}{2}\left[\frac{1}{\sigma_x}(x - zAv)^T(x - zAv) + \log\left(\det\left(C_v\right)\right) + v\left(C_v\right)^{-1} v^T\right] + \\ + \sum_{j=1}^{K}\left[(sh_g - 1)\log(g_j) - \frac{g_j}{sc_g}\right] + (sh_z - 1)\log(z) - \frac{z}{sc_z} \end{aligned} \tag{20}$$

the gradient with respect to different variable types assuming a Gamma prior over g and z:

$$\frac{\partial \log p(v, g, z \mid x)}{\partial v} = -\frac{z}{\sigma_x}\left[zA^TA + (C_v)^{-1}\right]v + \frac{z}{\sigma_x}A^Tx \quad (21)$$

$$\frac{\partial \log p(v, g, z \mid x)}{\partial g_i} = -\frac{1}{2}\mathrm{Tr}\left[\left(C_v^{-1} - C_v^{-1}vv^TC_v^{-1}\right)C_i\right] + \frac{sh_g - 1}{g_i} - \frac{g_i}{sc_g} \quad (22)$$

$$\frac{\partial \log p(v, g, z \mid x)}{\partial z} = 2v^TA^TAvz - x^TAv + \frac{sh_z - 1}{z} - \frac{z}{sc_z} \quad (23)$$

4 Gradient descent as the M-step

The complete-data likelihood with respect to a set of batch observations of size B is the following

$$\begin{gathered} p(\mathbf{V}, G, \mathbf{X} \mid C_{1..K}) = \prod_{n=1}^{N} p(X_n \mid V_n)p(V_n \mid g_n)p(g_n) = \\ \prod_{n=1}^{N} p(g_n) \prod_{b=1}^{B} p(x_{nb} \mid v_{nb})p(v_{nb} \mid g_n) \end{gathered} \quad (24)$$

Let's denote the logarithm of this by $\mathcal{L} = \log p(\mathbf{V}, G, \mathbf{X} \mid C_{1..K})$. We can approximate the integral of this logarithm over the joint posterior by averaging over L samples from it, separately for each observation x_n. As we will seek the values of the precision components $C_{1...K}$ that maximise this integral, we can discard each term not depending on these parameters. This way we arrive to the following expression

$$\begin{aligned} \mathcal{L} &\sim \sum_{n=1}^{N} \frac{1}{L} \sum_{l=1}^{L} -\frac{1}{2}\left[B \log\left(\det\left(C_v^{(l,n)}\right)\right) + \sum_{b=1}^{B} v^{(l,n,b)T}\left(C_v^{l,n}\right)^{-1} v^{l,n,b}\right] = \\ &= -\frac{1}{2L}\sum_{m=1}^{NL}\left[B \log\left(\det\left(C_v^m\right)\right) + \sum_{b=1}^{B} v^{(m,b)T}\left(C_v^m\right)^{-1} v^{m,b}\right] \end{aligned} \quad (25)$$

noting that the double summation over L samples over all N observations always happens on the same terms, so we can substitute it with a single sum that iterates over the full sample set.

To ensure that the optimisation procedure does not produce precision matrices that are not positive definite, we can optimise for the Cholesky upper triangle matrix istead of the precision matrix, as this also specifies the Gaussian completely.

$$C_k = U_k^T U_k \tag{26}$$

$$C_v = \sum_{k=1}^{K} g_k U_k^T U_k \tag{27}$$

$$\frac{\partial C_v^m}{\partial \left[U_k\right]_{i,j}} = g_k^m \frac{\partial \left(U_k^T U_k\right)}{\partial \left[U_k\right]_{i,j}} = g_k^m \left(U_k^T J^{ij} + J^{ji} U_k\right) \equiv g_k^m \hat{U}_k^{ij} \tag{28}$$

where J^{ij} is the single-entry matrix so that its element at index (i, j) is 1, and 0 everywhere else. Then by the chain rule, the derivative of $\mathcal{L}$ according to an element of U_k looks like this

$$\begin{aligned}
&\frac{\partial \mathcal{L}}{\partial \left[U_k\right]_{i,j}} = -\frac{1}{2L} \sum_{m=1}^{NL} \mathrm{Tr}\left[\frac{\partial \mathcal{L}^m}{\partial C_v^m} \frac{\partial C_v^m}{\partial \left[U_k\right]_{i,j}}\right] = \\
&-\frac{1}{2L} \sum_{m=1}^{LN} \mathrm{Tr}\left[\left[B\left(C_v^m\right)^{-1} - \sum_{b=1}^{B} \left(C_v^m\right)^{-1} v^{m,b} v^{(m,b)T} \left(C_v^m\right)^{-1}\right] g_k^m \hat{U}_k^{ij}\right] = \\
&-\frac{1}{2L} \mathrm{Tr}\left[\sum_{m=1}^{LN} g_k^m \left[B\left(C_v^m\right)^{-1} - \left(C_v^m\right)^{-1} \left(\sum_{b=1}^{B} v^{m,b} v^{(m,b)T}\right) \left(C_v^m\right)^{-1}\right] \hat{U}_k^{ij}\right]
\end{aligned} \tag{29}$$

The regularities of the $\hat{U}_k$ matrices allow us to replace the trace with a much more efficient computation:

$$M = -\frac{1}{2L} \left[\sum_{m=1}^{LN} g_k^m \left[B\left(C_v^m\right)^{-1} - \left(C_v^m\right)^{-1} \left(\sum_{b=1}^{B} v^{m,b} v^{(m,b)T}\right) \left(C_v^m\right)^{-1}\right]\right] \tag{30}$$

$$\frac{\partial \mathcal{L}}{\partial \left[U_k\right]_{i,j}} = \sum_{a=1}^{Dv} \left[M\right]_{j,a} \left[U_k\right]_{i,a} + \left[M\right]_{a,j} \left[U_k\right]_{i,a} \tag{31}$$

As a generalised M-step, we can move the parameters in the direction of the gradient scaled by a learning rate

$$\left[U_k\right]_{i,j}^{new} = \left[U_k\right]_{i,j}^{old} + \epsilon \frac{\partial \mathcal{L}}{\partial \left[U_k\right]_{i,j}} \tag{32}$$