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 symmetric Dirichlet prior distribution with concentration parameter α controlling the sparsity of the prior.

$$p(g) = \mathrm{Dir}(g; \alpha) \tag{3}$$

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

$$p(x \mid v) = \mathcal{N}(x; Av, C_x) \tag{4}$$

$$C_x = \sigma_x I; \tag{5}$$

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

$$p(V \mid g) = \prod_{b=1}^{B} \mathcal{N}(v_b; 0, C_v) \tag{6}$$

$$p(X \mid V) = \prod_{b=1}^{B} \mathcal{N}(x_b; Av_b, C_x) \tag{7}$$

The likelihood of the covariance components with respect to a dataset of size N, $\mathbf{X} = \{X_1 \dots X_N\}$ can be expressed as follows, using $\mathbf{C} = \{C_1 \dots C_k\}$

$$
\begin{aligned}
p(\mathbf{X} \mid \mathbf{C}) &= \prod_{n=1}^{N} p(X_n \mid \mathbf{C}) = \\
&= \prod_{n=1}^{N} \iint_{-\infty}^{\infty} p(X_n \mid V_n) p(V_n \mid g_n) p(g_n) \mathrm{d}V_n \mathrm{d}g_n = \\
&= \prod_{n=1}^{N} \int_{-\infty}^{\infty} p(g_n) \int_{-\infty}^{\infty} \prod_{b=1}^{B} p(x_{n,b} \mid v_{n,b}) p(v_{n,b} \mid g_n) \mathrm{d}V_n \mathrm{d}g_n
\end{aligned}
\tag{8}
$$

this can be transformed in the following way similarly to 13 and 14

$$
\begin{aligned}
p(x_{n,b} \mid v_{n,b}) p(v_{n,b} \mid g_n) &= \mathcal{N}(x_{n,b} : A v_{n,b}, \sigma_x I) \mathcal{N}(v_{n,b} : 0, \sum_{j=1}^{K} g_j C_j) = \\
&= \mathcal{N}(A^{+} x_{n,b} : 0, \sigma_x (A^T A)^{-1} + \sum_{j=1}^{K} g_j C_j) \cdot \mathcal{N}(v_{n,b} : \mu_v, C_v) \cdot const.
\end{aligned}
\tag{9}
$$

so the integral over v gives the first Gaussian over $A^{+}x$, as this is constant in v, and the second Gaussian integrates to one. We can approximate the integral over g in 8 with a finite sum over L samples from $p(g)$, which can be sampled directly. Discarding the term in 9 that is constant in all variables we get

$$
\begin{aligned}
p(\mathbf{X} \mid \mathbf{C}) &\approx \prod_{n=1}^{N} \frac{1}{L} \sum_{l=1}^{L} \prod_{b=1}^{B} \mathcal{N}(A^{+} x_{n,b} : 0, \sigma_x (A^T A)^{-1} + \sum_{j=1}^{K} g_j^{l,n} C_j) \sim \\
&\sim \prod_{n=1}^{N} \sum_{l=1}^{L} \prod_{b=1}^{B} \mathcal{N}\left(x_{n,b} : 0, \sigma_x I + A \left(\sum_{j=1}^{K} g_j^{l,n} C_j \right) A^T \right)
\end{aligned}
\tag{10}
$$

and we may take the logarithm

$$
\log p(\mathbf{X} \mid \mathbf{C}) \sim \sum_{n=1}^{N} \log \left[\sum_{l=1}^{L} \prod_{b=1}^{B} \mathcal{N}\left(x_{n,b} : 0, \sigma_x I + A \left(\sum_{j=1}^{K} g_j^{l,n} C_j \right) A^T \right) \right]
\tag{11}
$$

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

$$p(v \mid x, g) = \frac{p(x \mid v, g)p(v \mid g)}{p(x \mid g)} = \frac{\mathcal{N}(x; Av, \sigma_x I)\mathcal{N}(v; 0, C_v)}{\int_{-\infty}^{\infty} \mathcal{N}(x; Av, \sigma_x I)\mathcal{N}(v; 0, C_v)\mathrm{d}v} \tag{12}$$

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

$$\mathcal{N}(x; Av, \sigma_x I) = c_1 \mathcal{N}(v; (A^T A)^{-1} A^T x, \sigma_x (A^T A)^{-1}) = c_1 \mathcal{N}(v; A^+ x, \sigma_x (A^T A)^{-1}) \tag{13}$$

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. 12 can also be written as a Gaussian over v introducing a new constant

$$\mathcal{N}(x; Av, \sigma_x I)\mathcal{N}(v; 0, C_v) = c_1 c_2 \mathcal{N}(v; \mu_{post}, C_{post}) \tag{14}$$

The denominator of Eq. 12 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

$$p(v \mid x, g) = \mathcal{N}\left(v; \frac{1}{\sigma_x}\left(\frac{1}{\sigma_x}A^T A + C_v^{-1}\right)^{-1} A^T x, \left(\frac{1}{\sigma_x}A^T A + C_v^{-1}\right)^{-1}\right) \tag{15}$$

If we can assume that $\frac{1}{\sigma_x}$ is small compared to elements of the matrices (i.e. observation variance is large), then we may use the following approximation

$$C_{post} \approx C_v - \frac{1}{\sigma_x} C_v A^T A C_v \tag{16}$$

and for a batch of size B

$$p(V \mid X, g) = \prod_{b=1}^{B} \mathcal{N}\left(v_b; \mu_{post}(x_b), C_{post}\right) \tag{17}$$

which can be sampled directly from a Gaussian of dimension $D_v \times B$. The conditional posterior over g is defined as follows

$$p(g \mid X, V) = \frac{p(X \mid g, V)p(g \mid V)}{p(X \mid V)} = \frac{p(V \mid g)p(g)}{p(V)} \tag{18}$$

which can be sampled by a slice sampling scheme with the following target

$$\log p(g \mid X, V) \sim -\frac{1}{2}\left[B \log(\det(C_v)) + \sum_{b=1}^{B} v_b^T C_v^{-1} v_b\right] + (\alpha - 1)\sum_{k=1}^{K} \log(g_k) \tag{19}$$

3 M-step

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

$$p(\mathbf{V}, G, \mathbf{X} \mid C_{1..K}) = \prod_{n=1}^{N} p(X_n \mid V_n) p(V_n \mid g_n) p(g_n) = \prod_{n=1}^{N} p(g_n) \prod_{b=1}^{B} p(x_{nb} \mid v_{nb}) p(v_{nb} \mid g_n) \tag{20}$$

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

$$\begin{aligned} \mathcal{L} &\sim \sum_{n=1}^{N} \frac{1}{L} \sum_{l=1}^{L} -\frac{1}{2} \left[B \log \left(\det \left(C_v^{(l,n)} \right) \right) + \sum_{b=1}^{B} v^{(l,n,b)T} \left(C_v^{l,n} \right)^{-1} v^{l,n,b} \right] = \\ &= -\frac{1}{2L} \sum_{m=1}^{NL} \left[B \log \left(\det \left(C_v^{m} \right) \right) + \sum_{b=1}^{B} v^{(m,b)T} \left(C_v^{m} \right)^{-1} v^{m,b} \right] \end{aligned} \tag{21}$$

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.

$$C_k = U_k^T U_k \tag{22}$$

$$C_v = \sum_{k=1}^{K} g_k U_k^T U_k \tag{23}$$

$$\frac{\partial C_v^m}{\partial \left[U_k \right]_{i,j}} = g_k^m \frac{\partial \left(U_k^T U_k \right)}{\partial \left[U_k \right]_{i,j}} = g_k^m \left(U_k^T J^{ij} + J^{ji} U_k \right) \equiv g_k^m \hat{U}_k^{ij} \tag{24}$$

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

$$\frac{\partial \mathcal{L}}{\partial \left[U_k\right]_{i,j}} = -\frac{1}{2L}\sum_{m=1}^{NL}\mathrm{Tr}\left[\frac{\partial \mathcal{L}^m}{\partial C_v^m}\frac{\partial C_v^m}{\partial \left[U_k\right]_{i,j}}\right] =$$
$$-\frac{1}{2L}\sum_{m=1}^{LN}\mathrm{Tr}\left[\left[B\left(C_v^m\right)^{-1} - \sum_{b=1}^{B}\left(C_v^m\right)^{-1} v^{m,b}v^{(m,b)T}\left(C_v^m\right)^{-1}\right] g_k^m \hat{U}_k^{ij}\right] =$$
$$-\frac{1}{2L}\sum_{m=1}^{LN} g_k^m \mathrm{Tr}\left[\left[B\left(C_v^m\right)^{-1} - \left(C_v^m\right)^{-1}\left(\sum_{b=1}^{B} v^{m,b}v^{(m,b)T}\right)\left(C_v^m\right)^{-1}\right]\hat{U}_k^{ij}\right] \tag{25}$$

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

$$\left[U_k\right]_{i,j}^{new} = \left[U_k\right]_{i,j}^{old} + \epsilon\frac{\partial \mathcal{L}}{\partial \left[U_k\right]_{i,j}} \tag{26}$$

4 Parametrisation with precision components

The model can be equally well parametrised by precision components

$$p(v \mid g) = \mathcal{N}(v; 0, \Lambda_v^{-1}) \tag{27}$$
$$\Lambda_v = \sum_{k=1}^{K} g_k \Lambda_k \tag{28}$$

in this case the conditional posterior over v takes the form

$$p(v \mid x, g) = \mathcal{N}\left(v; \frac{1}{\sigma_x}\left(\frac{1}{\sigma_x}A^TA + \Lambda_v\right)^{-1}A^Tx, \left(\frac{1}{\sigma_x}A^TA + \Lambda_v\right)^{-1}\right) \tag{29}$$

and the sampling target for the conditional posterior of g will look as follows

$$\log p(g \mid X, V) \sim -\frac{1}{2}\left[B\log(\det(\Lambda_v^{-1})) + \sum_{b=1}^{B} v_b^T\Lambda_v v_b\right] + (\alpha - 1)\sum_{k=1}^{K}\log(g_k) \tag{30}$$

The gradient of the expectation of the complete-data log-likelihod with respect to the joint posterior will look like this

$$\Lambda_k = U_k^T U_k \tag{31}$$
$$\frac{\partial \mathcal{L}}{\partial \left[U_k\right]_{i,j}} = \frac{1}{L}\sum_{m=1}^{LN} g_k^m \mathrm{Tr}\left[\left[B\left(\Lambda_v^m\right)^{-1} - \sum_{b=1}^{B} v^{m,b}v^{(m,b)T}\right]\hat{U}_k^{ij}\right] \tag{32}$$