EM 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

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

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

2 E-step

The joint posterior over hidden variables g and V is the following

$$p(V, g \mid X) = p(X \mid V, g)\frac{p(V, g)}{p(x)} = p(X \mid V)p(V \mid g)p(g)\frac{1}{p(X)} \tag{8}$$

For the purpose of sampling, we can discard the normalisation factor $p(X)$ and the normalisation constants of the Gaussians. The logarithm of the Dirichlet prior over g looks like the following

$$\log p(g) = \log(\Gamma(\alpha K)) - \log(\Gamma(\alpha)^K) + (\alpha - 1)\sum_{k=1}^{K}\log(g_k) \tag{9}$$

We can discard the terms not depending on g. So taking the logarithm of the unnormalised posterior, the sampling target will look like this

$$\begin{aligned}&\log p(V, g \mid X) \sim \\ &-\frac{1}{2}\left[\sum_{b=1}^{B}\left[(x_b - Av_b)^T C_x^{-1}(x_b - Av_b) + v_b^T C_v^{-1} v_b\right] + B\log(\det(C_v))\right] + \\ &+ (\alpha - 1)\sum_{k=1}^{K}\log(g_k)\end{aligned} \tag{10}$$

where $g \in (0, 1]$ and $\sum_{k=1}^{K} g_k = 1$, and $-\infty$ everywhere else.

2.1 Hamiltonian Monte Carlo Sampling

The negative log-posterior can be regarded as an energy function for a dynamical system updating invariantly to the posterior distribution. To use Hamiltonian MC sampling, the gradient of this energy has to be constructed as follows (for $B = 1$)

$$\begin{aligned}&E(g, v) = \\ &\frac{1}{2}\left[(x - Av)^T C_x^{-1}(x - Av) + \log(\det(C_v)) + v^T C_v^{-1} v\right] + (\alpha - 1)\sum_{k=1}^{K}\log(g_k)\end{aligned} \tag{11}$$

$$\begin{aligned}&\frac{\partial E(g, v)}{\partial g_j} = \\ &\frac{1}{2}\sum_{a=1}^{D_v}\sum_{b=1}^{D_v}\left[\left(\sum_{k=1}^{K} g_k C_k\right)^{-1}\left[I - vv^T\left(\sum_{k=1}^{K} g_k C_k\right)^{-1}\right]\right]_{a,b}[C_j]_{a,b} + \frac{(\alpha - 1)}{g_j}\end{aligned} \tag{12}$$

$$\frac{\partial E(g, v)}{\partial v} = A^T C_x^{-1}(Av - x) + \left(\sum_{k=1}^{K} g_k C_k\right)^{-1} v \tag{13}$$

denoting the dimension of v as D_v and using differentiating rules 14, 15 and 16

$$\frac{\partial}{\partial M}\log(\det(M)) = M^{-T} \tag{14}$$

$$\frac{\partial}{\partial M}a^T M^{-1} a = -M^{-T}aa^T M^{-T} \tag{15}$$

$$f : \mathbb{R}^{D\times D} \to \mathbb{R} \quad h : \mathbb{R} \to \mathbb{R}^{D\times D}$$

$$\frac{\partial}{\partial x}f(h(x)) = \sum_{a=1}^{D}\sum_{b=1}^{D}\left[\frac{\partial}{\partial h(x)}f(h(x))\right]_{a,b}\left[\frac{\partial}{\partial x}h(x)\right]_{a,b} \tag{16}$$

However, given that the prior and thus the posterior over g does not have relevant gradient information outside of the unit interval, it is a better idea to use a combined sampler that proposes from Hamiltonian dynamics only for the dimensions of v.

2.2 Gibbs sampling

A more 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

$$p(v \mid x, g) = \frac{p(x \mid v, g)p(v \mid g)}{p(x \mid g)} = \frac{\mathcal{N}(x; Av, \sigma_x I)\mathcal{N}(v; 0, C_v)}{\int_{-\infty}^{\infty}\mathcal{N}(x; Av, \sigma_x I)\mathcal{N}(v; 0, C_v)\mathrm{d}v} \tag{17}$$

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

$$\mathcal{N}(x; Av, \sigma_x I) = c_1\mathcal{N}(v; -2(A^T A)^{-1}A^T x, \sigma_x(A^T A)^{-1}) \tag{18}$$

Consequently, the product of two Gaussians in the numerator of Eq. 17 can also be written as a Gaussian over v introducing a new constant

$$\mathcal{N}(x; Av, \sigma_x I)\mathcal{N}(v; 0, C_v) = c_1 c_2\mathcal{N}(v; \mu_{post}, C_{post}) \tag{19}$$

The denominator of Eq. 17 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

$$p(v \mid x, g) = \mathcal{N}\left(v; -\frac{2}{\sigma_x}\left(\frac{1}{\sigma_x}A^T A + C_v^{-1}\right)^{-1}A^T x, \left(\frac{1}{\sigma_x}A^T A + C_v^{-1}\right)^{-1}\right) \tag{20}$$

and for a batch of size B

$$p(V \mid X, g) = \prod_{b=1}^{B} \mathcal{N}\left(v_b; \mu_{post}(x_b), C_{post}\right) \tag{21}$$

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

$$p(g \mid X, V) = \frac{p(X \mid g, V)p(g \mid V)}{p(X \mid V)} = \frac{p(V \mid g)p(g)}{p(V)} \tag{22}$$

which can be sampled by a Metropolis-Hastings or slice sampling scheme with the following target

$$\log p(g \mid X, V) \sim -\frac{1}{2}\left[B \log(\det(C_v)) + \sum_{b=1}^{B} v_b^T C_v^{-1} v_b\right] + (\alpha - 1)\sum_{k=1}^{K} \log(g_k) \tag{23}$$

3 M-step

The complete-data likelihood with respect to a set of batch observations of size B, $\mathbf{X} = \{X_1 \dots X_N\}$ is the following

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

Its logarithm ($\mathcal{L} = \log p(\mathbf{V}, G, \mathbf{X} \mid C_{1..K})$) is similar to 10. 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 covariance 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

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

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. So taking the derivative with respect to one of the covariance components $j \in [1, k]$, we get

$$\frac{\partial \mathcal{L}}{\partial C_j} = -\frac{1}{2L} \sum_{m=1}^{LN} g_j^m \left[B \left(C_v^m\right)^{-1} - \sum_{b=1}^{B} \left(C_v^m\right)^{-1} v^{m,b} v^{(m,b)T} \left(C_v^m\right)^{-1} \right] \tag{26}$$

using $\frac{\partial}{\partial C_j}(C_v^m)^{-1} = \frac{\partial}{\partial C_j} \sum_{k=1}^{K} g_k^m C_k = g_j^m$ and the differentiating rules 14 and 15 (all C_j matrices are symmetric and regular, so transposes are identical and determinants are nonzero).

Setting 26 to zero, substituting for C_v and multiplying it from the left by $\sum_{k=1}^{K} g_k^m C_k$ gives

$$0 = -\frac{1}{2L} \sum_{m=1}^{LN} g_j^m \left[BI - \sum_{b=1}^{B} v^{m,b} v^{(m,b)T} \left(\sum_{k=1}^{K} g_k^m C_k \right)^{-1} \right] \tag{27}$$

multiplying by $-2L$ and by $\sum_{k=1}^{K} g_k^m C_k$, but this time from the right, gives

$$0 = \sum_{m=1}^{LN} g_j^m \left[B \left(\sum_{k=1}^{K} g_k^m C_k \right) - \sum_{b=1}^{B} v^{m,b} v^{(m,b)T} \right] \tag{28}$$

rearranging the sums yields

$$0 = \sum_{k=1}^{K} C_k B \left(\sum_{m=1}^{LN} g_j^m g_k^m \right) - \sum_{m=1}^{LN} g_j^m \sum_{b=1}^{B} v^{m,b} v^{(m,b)T} \tag{29}$$

From this we can express C_j to get

$$C_j^{new} = \frac{1}{B \sum_{m=1}^{LN} g_j^{m2}} \left[\sum_{m=1}^{LN} g_j^m \sum_{b=1}^{B} v^{m,b} v^{(m,b)T} - B \sum_{k \neq j}^{K} C_k \sum_{m=1}^{LN} g_j^m g_k^m \right] \tag{30}$$

that may be rearranged to

$$C_j^{new} = \frac{1}{\sum_{m=1}^{LN} g_j^{m2}} \sum_{m=1}^{LN} g_j^m \left[\frac{1}{B} \sum_{b=1}^{B} v^{m,b} v^{(m,b)T} - \sum_{k \neq j}^{K} g_k^m C_k \right] \tag{31}$$

or alternatively

$$C_j^{new} = \frac{1}{\sum_{m=1}^{LN} g_j^{m2}} \sum_{m=1}^{LN} g_j^m \left[\mathrm{Cov}(V^m) - (C_v^m - g_j^m C_j^{old}) \right] \tag{32}$$

3.1 Cholesky decomposition

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

$$C_j = U_j^T U_j \tag{33}$$

$$C_v = \sum_{k=1}^{K} g_k U_k^T U_k \tag{34}$$

$$\frac{\partial C_v}{\partial U_j} = 2 g_j U_j \tag{35}$$

so by the chain rule, the derivative of $\mathcal{L}$ according to U_j looks like this

$$\begin{aligned}
&\frac{\partial \mathcal{L}}{\partial U_j} = \frac{\partial \mathcal{L}}{\partial C_v} \frac{\partial C_v}{\partial U_j} = \\
&-\frac{1}{L} \sum_{m=1}^{LN} g_j^m \left[B \left(C_v^m\right)^{-1} - \sum_{b=1}^{B} \left(C_v^m\right)^{-1} v^{m,b} v^{(m,b)T} \left(C_v^m\right)^{-1} \right] U_j = \\
&-\frac{1}{L} \sum_{m=1}^{LN} g_j^m \left[B \left(C_v^m\right)^{-1} - \left(C_v^m\right)^{-1} \left(\sum_{b=1}^{B} v^{m,b} v^{(m,b)T} \right) \left(C_v^m\right)^{-1} \right] U_j
\end{aligned} \tag{36}$$