STATS215: Assignment 2

Libby Zhang [eyz]

Due: February 14, 2020

Problem 1: *Bernoulli GLMs as a latent variable models.*

Consider a Bernoulli regression model,

$$\begin{aligned} w &\sim \mathcal{N}(\mu, \Sigma) \\ y_n \mid x_n, w &\sim \text{Bern}(f(w^\mathsf{T} x_n)) \quad \text{for } n = 1, \ldots, N, \end{aligned}$$

where w and x_n are vectors in $\mathbb{R}^D$, $y_n \in \{0, 1\}$, and $f : \mathbb{R} \to [0, 1]$ is the mean function. In class we studied Newton's method for finding the maximum a posteriori (MAP) estimate $w^\star = \arg\max p(w \mid \{x_n, y_n\}_{n=1}^N)$. Now we will consider methods for approximating the full posterior distribution.

(a) Rather than using the logistic function, let the mean function be the normal cumulative distribution function (CDF), or "probit" function,

$$\begin{aligned} f(u) &= \Pr(z \le u) \text{ where } z \sim \mathcal{N}(0, 1) \\ &= \textstyle\int_{-\infty}^{u} \mathcal{N}(z; 0, 1)\, \mathrm{d}z. \end{aligned}$$

This is called the probit regression model. Show that the likelihood $p(y_n \mid x_n, w)$ is a marginal of a joint distribution,

$$p(y_n, z_n \mid x_n, w) = \mathbb{I}[z_n \ge 0]^{\mathbb{I}[y_n = 1]}\, \mathbb{I}[z_n < 0]^{\mathbb{I}[y_n = 0]} \mathcal{N}(z_n \mid x_n^\mathsf{T} w, 1).$$

The product of expressions raised to an indicator function can be equivalently written as the sum of the expressions times the indicator function. For the augmented joint distribution, we have

$$\begin{aligned} p(y_n, z_n \mid x_n, w) &= \mathbb{I}[z_n \ge 0]^{\mathbb{I}[y_n = 1]}\, \mathbb{I}[z_n < 0]^{\mathbb{I}[y_n = 0]} \mathcal{N}(z_n \mid x_n^\mathsf{T} w, 1) \\ &= \mathbb{I}[y_n = 1]\, \mathbb{I}[z_n \ge 0]\, \mathcal{N}(z_n \mid x_n^\mathsf{T} w, 1) + \mathbb{I}[y_n = 0]\, \mathbb{I}[z_n < 0]\, \mathcal{N}(z_n \mid x_n^\mathsf{T} w, 1) \end{aligned}$$

Now, when we integrate over z_n to recover the marginal disribution of y_n, we can split the integral into a sum of two disjoint integration limits:

$$\begin{aligned} p(y_n \mid x_n, w) &= \textstyle\int_{-\infty}^{\infty} p(y_n, z_n \mid x_n, w) \\ &= \textstyle\int_{-\infty}^{0} \mathbb{I}[y_n = 1]\, \mathcal{N}(z_n \mid x_n^\mathsf{T} w, 1) \mathrm{d}z_n + \int_{0}^{\infty} \mathbb{I}[y_n = 0] \mathcal{N}(z_n \mid x_n^\mathsf{T} w, 1) \mathrm{d}z_n \\ &= \mathbb{I}[y_n = 1] \textstyle\int_{-\infty}^{x_n^\mathsf{T} w} \mathcal{N}(z_n \mid 0, 1) \mathrm{d}z_n + \mathbb{I}[y_n = 0] \int_{x_n^\mathsf{T} w}^{\infty} \mathcal{N}(z_n \mid 0, 1) \mathrm{d}z_n \\ &= \mathbb{I}[y_n = 1] f(x_n^\mathsf{T} w) + \mathbb{I}[y_n = 0](1 - f(x_n^\mathsf{T} w)) \\ &= \text{Bern}(y_n \mid x_n^\mathsf{T} w) \end{aligned}$$

(b) Derive the conditional distributions $p(w \mid \{x_n, y_n, z_n\}_{n=1}^N)$ and $p(z_n \mid x_n, y_n, w)$.[1]

Posterior on w:

$$\begin{aligned} p(w \mid \{x_n, y_n, z_n\}_{n=1}^N) &\propto \prod_n p(y_n, z_n, w \mid x_n) \\ &\propto p(w) \prod_n p(y_n, z_n \mid x_n, w) \\ &\propto \mathcal{N}(\mu, \Sigma) \prod_n \mathbb{I}[z_n \geq 0]^{\mathbb{I}[y_n=1]} \mathbb{I}[z_n < 0]^{\mathbb{I}[y_n=0]} \mathcal{N}(z_n \mid x_n^\mathsf{T} w, 1) \end{aligned}$$

$$\begin{aligned} p(z_n \mid x_n, y_n, w) &\propto p(y_n, z_n \mid x_n, w) \\ &\propto \mathbb{I}[z_n \geq 0]^{\mathbb{I}[y_n=1]} \mathbb{I}[z_n < 0]^{\mathbb{I}[y_n=0]} \mathcal{N}(z_n \mid x_n^\mathsf{T} w, 1) \\ &\propto \mathbb{I}[y_n = 1] \mathbb{I}[z_n \geq 0] \mathcal{N}(z_n \mid x_n^\mathsf{T} w, 1) + \mathbb{I}[y_n = 0] \mathbb{I}[z_n < 0] \mathcal{N}(z_n \mid x_n^\mathsf{T} w, 1) \end{aligned}$$

(c) *Gibbs sampling* is a Markov chain Monte Carlo (MCMC) method for approximate posterior inference. It works by repeatedly sampling from the conditional distribution of one variable, holding all others fixed. For the probit regression model, this means iteratively performing these two steps:

1. Sample $z_n \sim p(z_n \mid x_n, y_n, w)$ for $n = 1, \ldots, N$ holding w fixed;
2. Sample $w \sim p(w \mid \{x_n, y_n, z_n\}_{n=1}^N)$ holding $\{z_n\}_{n=1}^N$ fixed.

Note the similarity to EM: rather than computing a posterior distribution over z_n, we draw a sample from it; rather than setting w to maximize the ELBO, we draw a sample from its conditional distribution. It can be shown that this algorithm defines a Markov chain on the space of $(w, \{z_n\}_{n=1}^N)$ whose stationary distribution is the posterior $p(w, \{z_n\}_{n=1}^N \mid \{x_n, y_n\}_{n=1}^N)$. In other words, repeating these steps infinitely many times would yield samples of w and $\{z_n\}_{n=1}^N$ drawn from their posterior distribution.

Implement this Gibbs sampling algorithm and test it on a synthetic dataset with $D = 2$ dimensional covariates and $N = 100$ data points. Scatter plot your samples of w and, for comparison, plot the true value of w that generated the data. Do your samples look approximately Gaussian distributed? How does the posterior distribution change when you vary N?

«Your figures and captions here.»

(d) **Bonus.** There are also auxiliary variable methods for logistic regression, where $f(u) = e^u/(1+e^u)$. Specifically, we have that,

$$\frac{e^{y_n \cdot w^\mathsf{T} x_n}}{1 + e^{w^\mathsf{T} x_n}} = \int_0^\infty \tfrac{1}{2} \exp\left\{\left(y_n - \tfrac{1}{2}\right) x_n^\mathsf{T} w - \tfrac{1}{2} z_n (w^\mathsf{T} x_n)^2\right\} \mathrm{PG}(z_n; 1, 0)\, \mathrm{d}z_n,$$

where $\mathrm{PG}(z; b, c)$ is the density function of the *Pólya-gamma* (PG) distribution over $z \in \mathbb{R}_+$ with parameters b and c. The PG distribution has a number of nice properties: it is closed under exponential tilting so that,

$$e^{-\frac{1}{2} z c^2} \mathrm{PG}(z; b, 0) \propto \mathrm{PG}(z; b, c),$$

[1] Observe that z_n is conditionally independent of $\{x_{n'}, y_{n'}, z_{n'}\}_{n' \neq n}$ given w.

and its expectation is available in closed form,

$$\mathbb{E}_{z\sim\mathrm{PG}(b,c)}[z] = \frac{b}{2c}\tanh\left(\frac{c}{2}\right).$$

Use these properties to derive an EM algorithm for finding $w^\star = \arg\max p(\{y_n\} \mid \{x_n\}, w)$. How do the EM updates compare to Newton's method?

Problem 2: *Spike sorting with mixture models*

As discussed in class, "spike sorting" is ultimately a mixture modeling problem. Here we will study the problem in more detail. Let $\{y_n\}_{n=1}^N$ represent a collection of spikes. Each $y_n \in \mathbb{R}^D$ is a vector containing features of the n-th spike waveform. For example, the features may be projections of the spike waveform onto the top D principal components. We have the following, general model,

$$\begin{aligned} z_n \mid \pi &\sim \pi \\ y_n \mid z_n, \theta &\sim p(y_n \mid \theta_{z_n}). \end{aligned}$$

The label $z_n \in \{1, \ldots, K\}$ indicates which of the K neurons generated the n-th spike waveform. The probability vector $\pi \in \Delta_K$ specifies a prior distribution on spike labels, and the parameters $\theta = \{\theta_k\}_{k=1}^K$ determine the likelihood of the spike waveforms y_n for each of the K neurons. The goal is to infer a posterior distribution $p(z_n \mid y_n, \pi, \theta)$ over labels for each observed spike, and to learn the parameters $\pi^\star$ and $\theta^\star$ that maximize the likelihood of the data.

(a) Start with a Gaussian observation model,

$$y_n \mid z_n, \theta \sim \mathcal{N}(y_n \mid \mu_{z_n}, \Sigma_{z_n}),$$

where $\theta_k = (\mu_k, \Sigma_k)$ includes the mean and covariance for the k-th neuron.

Derive an EM algorithm to compute $\pi^\star, \theta^\star = \arg\max p(\{y_n\}_{n=1}^N \mid \pi, \theta)$. Start by deriving the "responsibilities" $w_{nk} = p(z_n = k \mid y_n, \pi', \theta')$ for fixed parameters π' and θ'. Then use the responsibilities to compute the expected log joint probability,

$$\mathcal{L}(\pi, \theta) = \sum_{n=1}^N \mathbb{E}_{p(z_n \mid y_n, \pi', \theta')}[\log p(y_n, z_n \mid \pi, \theta)].$$

Finally, find closed-form expressions for $\pi^\star$ and $\theta^\star$ that optimize $\mathcal{L}(\pi, \theta)$.

First, we derive the weights. This is equivalent to the E-step of EM:

$$\begin{aligned} w_{nk} &= p(z_n = k \mid y_n, \pi', \theta') \\ &= p(y_n \mid z_n = k, \pi', \theta')\, p(z_n = k \mid \pi') \\ &= \mathcal{N}(y_n \mid \mu_k', \pi_k')\, \pi_k' \end{aligned}$$

Next, we calculate the expected log joint probability. This is the objective that we will be maximizing during our M-step.

$$\begin{aligned} \mathcal{L}(\pi, \theta) &= \sum_{n=1}^N \mathbb{E}_{p(z_n \mid y_n, \pi', \theta')}\big[\log[p(y_n, z_n \mid \pi', \theta')]\big] \\ &= \sum_{n=1}^N \mathbb{E}_{p(z_n \mid y_n, \pi', \theta')}\Big[\log\Big[\textstyle\prod_k (p(y_n \mid z_n, \pi', \theta') p(z_n \mid \pi', \theta'))^{\mathbb{I}[z_n = k]}\Big]\Big] + \text{const.} \\ &= \sum_{n=1}^N \mathbb{E}_{p(z_n \mid y_n, \pi', \theta')}\Big[\sum_k \mathbb{I}[z_n = k]\Big(-\frac{1}{2}(y_n - \mu_k')^\top J_k'^{-1}(y_n - \mu_k') + \frac{1}{2}\log|J_k| + \log \pi_k'\Big)\Big] + \text{const.} \\ &= \sum_{n=1}^N \sum_k \mathbb{E}_{p(z_n = k \mid y_{nk}, \pi_k', \theta_k')}\Big[\mathbb{I}[z_n = k]\Big(-\frac{1}{2}(y_n - \mu_k')^\top J_k'^{-1}(y_n - \mu_k') + \frac{1}{2}\log|J_k| + \log \pi_k'\Big)\Big] + \text{const.} \end{aligned}$$

where we substitute the inverse of the covariance matrix with the precision matrix, $J'_k = \Sigma'^{-1}_k$.

Now, to find the parameters π^* and $\theta^* = (\mu^*, \Sigma^*)$ which maximize this expected log joint, we take the gradient of the objective with respect to each parameter and set to 0.

$$\begin{aligned}
\nabla_{\mu_k} \mathscr{L}(\pi, \theta) &= \sum_n w_{nk} J'_k (y_n - \mu^*_k) \\
&= J'_k \sum_n w_{nk} (y_n - \mu^*_k) = 0 \\
\Rightarrow \mu^*_k &= \frac{\sum_n w_{nk}\, y_n}{\sum_n w_{nk}}
\end{aligned}$$

$$\begin{aligned}
\nabla_{J_k} \mathscr{L}(\pi, \theta) &= \sum_n w_{nk} (-\frac{1}{2})(y_n - \mu'_k)(y_n - \mu'_k)^\mathsf{T} + \frac{1}{2} J^{*-1}_k \\
\Rightarrow J^{*-1}_k &= \frac{\sum_n w_{nk} (y_n - \mu'_k)(y_n - \mu'_k)^\mathsf{T}}{\sum_n w_{nk}}
\end{aligned}$$

where we Jacobi's formula to calculate the derivative of a determinant:

$$\frac{d}{dx}|X| = \text{adj}(X) = |X|\, X^{-1}; \qquad \frac{d}{dx} \log |X| = \frac{1}{|X|} |X|\, X^{-1} = X^{-1}$$

We can interpret the optimal θ parameters as the weighted average of the previous θ', weighted by the responsibilities/posterior values of assignments z_n.

Finally, to find the maximizing argument π^*_k, we want to first add a Lagrangian multiplier to encoe the constraint that $\sum_k \pi_k = 1$.

$$\nabla_{\pi_k} \mathscr{L}(\pi, \theta) + \sum_n \lambda (1 - \sum_k \pi_{nk}) = \sum_n w_{nk} \frac{1}{\pi^*_{nk}} - \lambda = 0 \Rightarrow \pi^*_{nk} = \frac{1}{\lambda} \sum_n w_{nk}$$

Finally, to find the value of this Lagrange multiplier, note that we share the same λ for each of the K π_k's. So, with the simplex constraint on π^*, we find

$$\begin{aligned}
\sum_k \pi^*_k &= \sum_k \frac{1}{\lambda} \left(\sum_n \pi^*_{nk} \right) \\
&= \frac{1}{\lambda} \sum_n \left(\sum_k \pi^*_{nk} \right) \\
&= \frac{1}{\lambda} \sum_n 1 = \frac{N}{\lambda} = 1 \Rightarrow \lambda = N
\end{aligned}$$

Therefore, the maximizing responsibilities are just the expected value of expeted value of the responsibilities/weights.

$$\pi^*_{nk} = \frac{\sum_n w_{nk}}{N}$$

(b) The Gaussian model can be sensitive to outliers and lead spikes from one neuron to be split into two clusters. One way to side-step this issue is to replace the Gaussian with a heavier-tailed distribution like the multivariate Student's t, which has probability density,

$$p(y_n \mid \theta_{z_n}) = \frac{\Gamma[(\alpha_0 + D)/2]}{\Gamma(\alpha_0/2)\alpha_0^{D/2}\pi^{D/2}\left|\Sigma_{z_n}\right|^{1/2}}\left[1 + \frac{1}{\alpha_0}(y_n - \mu_{z_n})^\top \Sigma_{z_n}^{-1}(y_n - \mu_{z_n})\right]^{-(\alpha_0+D)/2}.$$

We will treat α_0 as a fixed hyperparameter.

Like the negative binomial distribution studied in HW1, the multivariate Student's t can also be represented as an infinite mixture,

$$p(y_n \mid \theta_{z_n}) = \int p(y_n, \tau_n \mid \theta_{z_n})\, \mathrm{d}\tau_n = \int \mathcal{N}(y_n; \mu_{z_n}, \tau_n^{-1}\Sigma_{z_n})\,\mathrm{Gamma}(\tau_n; \tfrac{\alpha_0}{2}, \tfrac{1}{2})\, \mathrm{d}\tau_n.$$

We will derive an EM algorithm to find $\pi^\star, \theta^\star$ in this model.

First, show that the posterior takes the form

$$\begin{aligned} p(\tau_n, z_n \mid y_n, \pi, \theta) &= p(z_n \mid y_n, \pi, \theta)\, p(\tau_n \mid z_n, y_n, \theta) \\ &= \prod_{k=1}^{K}\Big[w_{nk}\,\mathrm{Gamma}(\tau_n \mid a_{nk}, b_{nk})\Big]^{\mathbb{I}[z_n=k]}, \end{aligned}$$

and solve for the parameters w_{nk}, a_{nk}, b_{nk} in terms of y_n, π, and θ.

$$\begin{aligned} p(\tau_n, z_n \mid y_n, \pi, \theta) &= p(\tau_n \mid y_n, \theta, z_n)\, p(z_n \mid y_n, \pi, \theta) \\ &\propto p(y_n \mid \tau_n, \theta, z_n)\, p(\tau_n \mid \alpha_0)\; p(y_n \mid z_n, \pi) p(z_n \mid \pi) \\ &\propto \mathcal{N}(y_n \mid \mu_{z_n}, \tau_n^{-1}\Sigma_{z_n})\,\mathrm{Gamma}\Big(z_n \mid \frac{\alpha_0}{2}, \frac{1}{2}\Big)\,\mathrm{tdist}\big(y_n \mid (\theta, z_n, \alpha_0)\big)\,\mathrm{Cat}(z_n \mid \pi) \\ &\propto \prod_k \Big[|\tau^{-1}\Sigma_{z_n}|^{-\frac{1}{2}} \exp\Big\{-\frac{1}{2}||y_n - \mu_{z_n}||^2_{\Sigma_{z_n}}\tau_n\Big\}\,\tau_n^{\frac{\alpha_0}{2}-1}\exp\Big\{-\frac{1}{2}\tau_n\Big\} \\ &\qquad\qquad \Big(\frac{1}{\alpha_0}||y_n - \mu_{z_n}||^2_{\Sigma_{z_n}} + 1\Big)\,\pi_k\Big]^{\mathbb{I}[z_n=k]} \\ &\propto \prod_k \Big[\tau_n^{\frac{\alpha_0+D}{2}-1}\exp\Big\{-\tau_n\big(1 + ||y_n - \mu_{z_n}||^2_{\Sigma_{z_n}}\big)/2\Big\}\,\tilde{\pi}_{nk}\Big]^{\mathbb{I}[z_n=k]} \\ &= \prod_k \Big[\mathrm{Gamma}(\tau_n \mid a_{nk}, b_{nk})\, w_{nk}\Big]^{\mathbb{I}[z_n=k]} \end{aligned}$$

where $\tilde{\pi}_{nk} = \mathrm{tdist}\big(y_n \mid (\theta, z_n = k, \alpha_0)\big)\,\pi_k$ and the parameters are

$$\begin{aligned} a_{nk} &= (\alpha_0 + D)/2 \\ b_{nk} &= (1 + ||y_n - \mu'_k||^2_{\Sigma'_k})/2 \\ w_{nk} &= \frac{\tilde{\pi}_{nk}}{\sum_{k'}\tilde{\pi}_{nk'}} \end{aligned}$$

(c) Now compute the expected log joint probability,

$$\mathscr{L}(\pi,\theta) = \sum_{n=1}^{N} \mathbb{E}_{p(\tau_n, z_n \mid y_n, \pi', \theta')}[\log p(y_n, z_n, \tau_n \mid \pi, \theta)],$$

using the fact that $\mathbb{E}[X] = a/b$ for $X \sim \text{Gamma}(a,b)$. You may omit terms that are constant with respect to π and θ.

$$\begin{aligned}
\mathscr{L}(\pi,\theta) &= \sum_n \mathbb{E}_{p(\tau_n, z_n \mid y_n, \pi', \theta')}\Big[\log p(y_n, \tau_n, z_n \mid \pi, \theta)\Big] \\
&= \sum_n \mathbb{E}_{p(\tau_n, z_n \mid y_n, \pi', \theta')}\Big[\log \prod_k \Big[p(y_n, \mid \tau_n, z_n = k, \pi_k, \theta_k) p(\tau_n \mid \alpha_0) p(z_n = k \mid \pi_k)\Big]^{\mathbb{I}[z_n = k]}\Big] \\
&= \sum_n \sum_k \mathbb{E}_{p(\tau_n, z_n \mid y_n, \pi', \theta')}\Big[\mathbb{I}[z_n = k] \log \mathcal{N}(y_n \mid \mu_k', \tau_n^{-1}\Sigma_k)\,\text{Gamma}(\tau_n \mid \frac{\alpha_0}{2}, \frac{1}{2})\,\pi_k\Big] \\
&= \sum_n \sum_k \mathbb{E}_{p(z_n \mid y_n, \pi', \theta')\, p(\tau_n \mid y_n, \pi', \theta')}\Big[\mathbb{I}[z_n = k]\big(-\frac{\tau_n}{2}||y_n - \mu_k||^2_{J_k^{-1}} + \frac{D}{2}\log(\tau_n) + \frac{1}{2}\log(|J_k|) \\
&\qquad + (\frac{\alpha_0}{2} - 1)\log(\tau_n) - \frac{1}{2}\tau_n + \log(\pi_k')\big)\Big] + \text{const.} \\
&= \sum_n \sum_k \mathbb{E}_{p(z_n \mid y_n, \pi', \theta')\, p(\tau_n \mid y_n, \pi', \theta')}\Big[\mathbb{I}[z_n = k]\big(\tau_n(-\frac{1}{2}||y_n - \mu_k||^2_{J_k^{-1}} - \frac{1}{2}) + \frac{1}{2}\log(|J_k|) \\
&\qquad + (\frac{\alpha_0 + D}{2} - 1)\log(\tau_n) + \log(\pi_k)\big)\Big] + \text{const.} \\
&= \sum_n \sum_k w_{nk}\Big(\frac{a_{nk}}{b_{nk}}(-\frac{1}{2}||y_n - \mu_k||^2_{J_k^{-1}} - \frac{1}{2}) + \frac{1}{2}\log(|J_k|) \\
&\qquad + (\frac{\alpha_0 + D}{2} - 1)[\log(b_{nk}) + \psi(a_{nk})] + \log(\pi_k)\Big) + \text{const.}
\end{aligned}$$

where we use the fact that $\mathbb{E}[X] = a/b$ and $\mathbb{E}[\log X] = \log b + \psi(a)$ for $X \sim \text{Gamma}(a,b)$ and where $\psi(\cdot)$ is the digamma function.

(d) Finally, solve for $\pi^\star$ and $\theta^\star$ that maximize the expected log joint probability. How does your answer compare to the solution you found in part (a)?

Finally, we take the derivative of the objective with respect to each parameter, as we did in the latter part of part (a) to find π^* and $\theta^* = (\mu^*, \Sigma^*)$

$$\begin{aligned}
\nabla_{\mu_k} \mathcal{L}(\pi, \theta) &= \sum_n w_{nk} \frac{a_{nk}}{b_{nk}} J_k (y_n - \mu_k^*) \\
&= J_k \sum_n w_{nk} \frac{a_{nk}}{b_{nk}} (y_n - \mu_k^*) = 0 \\
\Rightarrow \mu_k^* &= \frac{\sum_n \tilde{w}_{nk} y_n}{\sum_n \tilde{w}_{nk}}
\end{aligned}$$

where we let $\tilde{w}_{nk} = w_{nk} \frac{a_{nk}}{b_{nk}}$. Compared to the optimal mean parameter found in part (a), which we will denote $\hat{\mu}_k^*$, we see that both optimal parameters take the form of a "weighted" sum of observed data points y_n. However, we have

$$\hat{w}_{nk} = \mathcal{N}(y_n \mid \theta', z_n = k)\pi_k' \qquad \text{vs.} \qquad \tilde{w}_{nk} = \frac{a_{nk}}{b_{nk}} \text{tdist}(y_n \mid (\theta', z_n = k, \alpha_0))\pi_k'$$

We see that our w_{nk} (which is the updated weights, so not exactly the same as in part (b)), capture the spread of y_n according to the t-distribution and the posterior mode of τ_n.

Similarly, we find that

$$J_k^{*-1} = \frac{\sum_n \tilde{w}_{nk} (y_n - \mu_k')(y_n - \mu_k')^\mathsf{T}}{\sum_n \tilde{w}_{nk}}$$

We find that the form of the maximizing assignments $\pi_k^* = \sum_n w_{nk}/N$ are also similiar in form to $\hat{\pi}_k^* = \sum_n \hat{w_{nk}}/N$, with the altered weights.

Problem 3: *Poisson matrix factorization*

Many biological datasets come in the form of matrices of non-negative counts. RNA sequencing data, neural spike trains, and network data (where each entry indicate the number of connections between a pair of nodes) are all good examples. It is common to model these counts as a function of some latent features of the corresponding row and column. Here we consider one such model, which decomposes a count matrix into a superposition of non-negative row and column factors.

Let $Y \in \mathbb{N}^{M \times N}$ denote an observed $M \times N$ matrix of non-negative count data. We model this matrix as a function of non-negative row factors $U \in \mathbb{R}_+^{M \times K}$ and column factors $V \in \mathbb{R}_+^{N \times K}$. Let $u_m \in \mathbb{R}_+^K$ and $v_n \in \mathbb{R}_+^K$ denote the m-th and n-th rows of U and V, respectively. We assume that each observed count y_{mn} is conditionally independent of the others given its corresponding row and column factors. Moreover, we assume a linear Poisson model,

$$y_{mn} \mid u_m, v_n \sim \text{Poisson}(u_m^\top v_n).$$

(Since u_m and v_n are non-negative, the mean parameter is valid.) Finally, assume gamma priors,

$$u_{mk} \sim \text{Gamma}(\alpha_0, \beta_0), \qquad v_{nk} \sim \text{Gamma}(\alpha_0, \beta_0).$$

Note that even though the gamma distribution is conjugate to the Poisson, here we have an inner product of two gamma vectors producing one Poisson random variable. The posterior distribution is more complicated. The entries of u_m are not independent under the posterior due to the "explaining away" effect. Nevertheless, we will derive a mean-field variational inference algorithm to approximate the posterior distribution.

(a) First we will use an augmentation trick based on the additivity of Poisson random variables; i.e. the fact that

$$y \sim \text{Poisson}\left(\sum_k \lambda_k\right) \iff y = \sum_k y_k \text{ where } y_k \sim \text{Poisson}(\lambda_k) \text{ independently,}$$

for any collection of non-negative rates $\lambda_1, \ldots, \lambda_K \in \mathbb{R}_+$. Use this fact to write the likelihood $p(y_{mn} \mid u_m, v_n)$ as a marginal of a joint distribution $p(y_{mn}, \bar{y}_{mn} \mid u_m, v_n)$ where $\bar{y}_{mn} = (y_{mn1}, \ldots, y_{mnK})$ is a length-K vector of non-negative counts.

We observe that y_{mn} is deterministic given $\bar{y}_{mn}$. This relation can also be stated as $p(y_{mn} \mid \bar{y}_{mn}, u_m, v_n) = 1$

The conditional joint distribution between the two variables can thus be stated as

$$\begin{aligned} p(y_{mn}, \bar{y}_{mn} \mid u_m, v_n) &= p(y_{mn} \mid \bar{y}_{mn}, u_m, v_n)\, p(\bar{y}_{mn} \mid u_m, v_n) \\ &= \mathbb{I}\left[y_{mn} = \textstyle\sum_k \bar{y}_{mnk}\right] \prod_k \text{Poisson}(\bar{y}_{mnk} \mid \lambda_{mnk}; u_{mk}, v_{mk}) \end{aligned}$$

When we sum over all combinations $\bar{y}_m n \in \bar{Y}_{mn}$, then we recover

$$\begin{aligned} \sum_{\bar{y}_{mn} \in \bar{Y}_{mn}} p(y_{mn}, \bar{y}_{mn} \mid u_m, v_n) &= \sum_{\bar{y}_{mn} \in \bar{Y}_{mn}} \mathbb{I}\left[y_{mn} = \textstyle\sum_k \bar{y}_{mnk}\right] \prod_k \text{Poisson}(\bar{y}_{mnk} \mid \lambda_{mnk}; u_{mk}, v_{nk}) \\ &= \text{Poisson}(y_{mn} = \textstyle\sum_k \bar{y}_{mnk} \mid \lambda_{mn} = \sum_k \lambda_{mnk}; u_{mk}, v_{nk}) \\ &= \text{Poisson}(y_{mn} \mid \lambda_{mn}; u_m, v_n) = p(y_{mn} \mid u_m, v_n) \end{aligned}$$

(b) Let $\bar{Y} \in \mathbb{N}^{M \times N \times K}$ denote the augmented data matrix with entries y_{mnk} as above. We will use mean field variational inference to approximate the posterior as,

$$p(\bar{Y}, U, V \mid Y) \approx q(\bar{Y})\, q(U)\, q(V) = \left[\prod_{m=1}^{M}\prod_{n=1}^{N} q(\bar{y}_{mn})\right]\left[\prod_{m=1}^{M}\prod_{k=1}^{K} q(u_{mk})\right]\left[\prod_{n=1}^{N}\prod_{k=1}^{K} q(v_{nk})\right].$$

We will solve for the optimal posterior approximation via coordinate descent on the KL divergence to the true posterior. Recall that holding all factors except for $q(\bar{y}_{mn})$ fixed, the KL is minimized when

$$q(\bar{y}_{mn}) \propto \exp\left\{\mathbb{E}_{q(\bar{Y}_{\neg mn})q(U)q(V)}\left[\log p(Y, \bar{Y}, U, V)\right]\right\},$$

where $q(\bar{Y}_{\neg mn}) = \prod_{(m',n')\neq(m,n)} q(\bar{y}_{m'n'})$ denotes all variational factors except for the (m,n)-th.

Show that the optimal $q(\bar{y}_{mn})$ is a multinomial of the form,

$$q(\bar{y}_{mn}) = \text{Mult}(\bar{y}_{mn}; y_{mn}, \pi_{mn}),$$

and solve for $\pi_{mn} \in \Delta_K$. You should write your answer in terms of expectations with respect to the other variational factors.

First, we consider what the log of the multinomial distribution that we expect to get is:

$$\begin{aligned}
\log q(\bar{y}_{mn}) &= \log\Big[\text{Mult}(\bar{y}_{mn}; y_{mn}, \pi_{mn}),\Big] \\
&= \log\Big[\binom{y_{mn}}{\{\bar{y}_{mnk}\}_k}\frac{y_{mn}!}{\prod_k \bar{y}_{mnk}!}\prod_k \pi_{mnk}^{\bar{y}_{mnk}}\Big] \\
&= \log\Big[\mathbb{I}[y_{mn} = \textstyle\sum_k \bar{y}_{mnk}]\frac{y_{mn}!}{\prod_k \bar{y}_{mnk}!}\prod_k \pi_{mnk}^{\bar{y}_{mnk}}\Big] \\
&= \log \mathbb{I}[y_{mn} = \textstyle\sum_k \bar{y}_{mnk}] + \log[y_{mn}!] - \sum_k \log[\bar{y}_{mnk}!] + \sum_k \bar{y}_{mnk}\log(\pi_{mnk})
\end{aligned}$$

Now,

$$\begin{aligned}
\log q(\bar{y}_{mn}) &= \mathbb{E}_{q(u_{mn})q(v_{mn})}\left[\log p(y_{mn}, \bar{y}_{m,n} \mid u_{mn}, v_{mn})\right] + \text{const.} \\
&= \mathbb{E}_{q(u_{mn})q(v_{mn})}\left[\log\left[\mathbb{I}[y_{mn} = \textstyle\sum_k \bar{y}_{mnk}]\prod_k \frac{\lambda_{mnk}^{\bar{y}_{mnk}}}{\bar{y}_{mnk}!}\exp\{-\lambda_{mnk}\}\right]\right] + \text{const.} \\
&= \log\left[\mathbb{I}[y_{mn} = \textstyle\sum_k \bar{y}_{mnk}]\right] - \sum_k \log\left[\bar{y}_{mnk}!\right] - \mathbb{E}_{q(u_{mn})q(v_{mn})}\left[\textstyle\sum_k \lambda_{mnk} + \sum_k \bar{y}_{mnk}\log[\lambda_{mnk}]\right] + \text{const.}
\end{aligned}$$

which has the multinomial form. We find the unnormalized parameter

$$\log \tilde{\pi}_{mnk} = \mathbb{E}_{q(u_{mk})q(v_{nk})}[\log \lambda_{mnk}] = \mathbb{E}_{q(u_{mk})}[\log u_{mk}] + \mathbb{E}_{q(v_{nk})}[\log v_{nk}]$$

Recall that we have the constraint that $\pi_{mn} \in \Delta_K$, so

$$\pi_{mn} = \exp\left\{\mathbb{E}_{q(u_{mk})q(v_{nk})}[\log \lambda_{mnk}] - \sum_j \mathbb{E}_{q(u_{mj})q(v_{nj})}[\log \lambda_{mnj}]\right\}$$

(c) Holding all factors but $q(u_{mk})$ fixed, show that optimal distribution is

$$q(u_{mk}) = \text{Gamma}(u_{mk}; \alpha_{mk}, \beta_{mk}).$$

Solve for α_{mk}, β_{mk}; write your answer in terms of expectations with respect to $q(\bar{y}_{mn})$ and $q(v_{nk})$.

$$\begin{aligned}
\log q(u_{mk}) &= \mathbb{E}_{q(\bar{y}_{mn})q(v_n)}\Big[\log p(y_{mn}, \bar{y}_{mn}, u_m, v_n)\Big] + \text{const.} \\
&= \mathbb{E}_{q(\bar{y}_{mn})q(v_n)}\Big[\log p(y_{mn}, \bar{y}_{mn}, \,|\, u_m, v_n) + \log p(u_m) + \log p(v_n)\Big] + \text{const.} \\
&= \mathbb{E}_{q(\bar{y}_{mnk})q(v_{nk})}\Big[\bar{y}_{mnk}\,\log[\lambda_{mnk}] - \lambda_{mnk} + (\alpha_0 - 1)\log[u_{mk}] - \beta_0\, u_{mk}\Big] + \text{const.} \\
&= \mathbb{E}_{q(\bar{y}_{mnk})q(v_{nk})}\Big[\bar{y}_{mnk}\,(\log[u_{mk}] + \log[v_{nk}]) - u_{mk}\, v_{nk} + (\alpha_0 - 1)\log[u_{mk}] - \beta_0\, u_{mk}\Big] + \text{const.} \\
&= \big(\mathbb{E}_{q(\bar{y}_{mnk})}[\bar{y}_{mnk}] + \alpha_0 - 1\big)\,\log[u_{mk}]] - \big(\mathbb{E}_{q(v_{nk})}[v_{nk}] + \beta_0\big)\,u_{mk} + \text{const.} \\
\Rightarrow \alpha_{mk} &= \mathbb{E}_{q(\bar{y}_{mnk})}[\bar{y}_{mnk}] + \alpha_0 \\
\Rightarrow \beta_{mk} &= \mathbb{E}_{q(v_{nk})}[v_{nk}] + \beta_0
\end{aligned}$$

(d) Use the symmetry of the model to determine the parameters of the optimal gamma distribution for $q(v_{nk})$, holding $q(\bar{y}_{mn})$ and $q(u_{mk})$ fixed,

$$q(v_{nk}) = \text{Gamma}(v_{nk}; \alpha_{nk}, \beta_{nk}).$$

Solve for α_{nk}, β_{nk}; write your answer in terms of expectations with respect to $q(\bar{y}_{mn})$ and $q(u_{mk})$.

By symmetry, we find

$$\begin{aligned}
\Rightarrow \alpha_{nk} &= \mathbb{E}_{q(\bar{y}_{mnk})}[\bar{y}_{mnk}] + \alpha_0 \\
\Rightarrow \beta_{nk} &= \mathbb{E}_{q(u_{mk})}[u_{mk}] + \beta_0
\end{aligned}$$

(e) Now that the form of all variational factors has been determined, compute the required expectations (in closed form) to write the coordinate descent updates in terms of the other variational parameters. Use the fact that $\mathbb{E}[\log X] = \psi(\alpha) - \log\beta$ for $X \sim \text{Gamma}(\alpha, \beta)$, where ψ is the digamma function.

$$\begin{aligned}
q(\bar{y}_{mn}): &\quad \log\tilde{\pi}_{mnk} = (\psi(a_m k) - \log[b_{mk}]) + (\psi(a_n k) - \log[b_{nk}]) \\
q(u_{mk}): &\quad a_m k = y_{mn}\,\pi_{mnk} + \alpha_0 \\
&\quad b_{mk} = \frac{a_{mk}}{b_{mk}} + \beta_0 \\
q(v_{nk}): &\quad a_n k = y_{mn}\,\pi_{mnk} + \alpha_0 \\
&\quad b_{nk} = \frac{a_{nk}}{b_{nk}} + \beta_0
\end{aligned}$$

(f) Suppose that Y is a sparse matrix with only $S \ll MN$ non-zero entries. What is the complexity of this mean-field coordinate descent algorithm?

«Your answer here.»

Problem 4: *Apply Poisson matrix factorization to C. elegans connectomics data*

Make a copy of this Colab notebook:

https://colab.research.google.com/drive/1ZMwcB6vzVaXz4WJiNT514b7zB5s3_SBk

Use your solutions from Problem 3 to finish the incomplete code cells. Once you're done, run all the code cells, save the notebook in .ipynb format, print a copy in .pdf format, and submit these files along with the rest of your written assignment.

I was not able to successfully download my .ipynb file to a PDF (problems with nbconvert, etc.). The viewable link to my CoLab for the assignment is here:

https://colab.research.google.com/drive/1MpaejcPT-47gLWmiCNf-e_SFSwnThPkQ