Coordinate Ascent Variational Inference

STATS 305C: Applied Statistics

Scott Linderman

April 26, 2023

Where are we?

Model	Algorithm	Application
Multivariate Normal Models	Conjugate Inference	Bayesian Linear Regression
Hierarchical Models	MCMC (MH & Gibbs)	Modeling Polling Data
Probabilistic PCA & Factor Analysis	MCMC (HMC)	Images Reconstruction
Mixture Models	EM & Variational Inference	Image Segmentation
Mixed Membership Models	Coordinate Ascent VI	Topic Modeling
Variational Autoencoders	Black Box, Amortized VI	Image Generation
State Space Models	Message Passing	Segmenting Video Data
Bayesian Nonparametrics	Fancy MCMC	Modeling Neural Spike Trains

Review: Bayesian Mixture Model

1. Sample the proportions from a Dirichlet prior:

$$\boldsymbol{\pi} \sim \mathrm{Dir}(\boldsymbol{\alpha}) \tag{1}$$

2. Sample the parameters (e.g., cluster means) for each component:

$$\boldsymbol{\mu}_k \overset{\text{iid}}{\sim} p(\boldsymbol{\mu} \mid \boldsymbol{\phi}, \nu) \qquad \text{for } k = 1, \ldots, K \tag{2}$$

3. Sample the assignment of each data point:

$$z_n \overset{\text{iid}}{\sim} \boldsymbol{\pi} \qquad\qquad \text{for } n = 1, \ldots, N \tag{3}$$

4. Sample data points given their assignments:

$$\boldsymbol{x}_n \sim p(\boldsymbol{x} \mid \boldsymbol{\mu}_{z_n}) \qquad \text{for } n = 1, \ldots, N \tag{4}$$

Review: Exponential family mixture models

What about $p(\boldsymbol{x} \mid \boldsymbol{\mu}_k)$ and $p(\boldsymbol{\mu}_k \mid \boldsymbol{\phi}, \nu)$?

Let's assume an **exponential family** likelihood,

$$p(\boldsymbol{x} \mid \boldsymbol{\mu}_k) = h(\boldsymbol{x}_n) \exp\left\{\langle t(\boldsymbol{x}_n), \boldsymbol{\mu}_k\rangle - A(\boldsymbol{\mu}_k)\right\}. \tag{5}$$

Then assume a **conjugate prior**,

$$p(\boldsymbol{\mu}_k \mid \boldsymbol{\phi}, \nu) \propto \exp\left\{\langle \boldsymbol{\phi}, \boldsymbol{\mu}_k\rangle - \nu A(\boldsymbol{\mu}_k)\right\}. \tag{6}$$

The hyperparmeters $\boldsymbol{\phi}$ are **pseudo-observations** of the sufficient statistics (like statistics from fake data points) and ν is a **pseudo-count** (like the number of fake data points).

Note that the product of prior and likelihood remains in the same family as the prior. That's why we call it conjugate.

Review: Gaussian mixture model

Assume the conditional distribution of $\boldsymbol{x}_n$ is a Gaussian with mean $\boldsymbol{\mu}_k \in \mathbb{R}^D$ and identity covariance,

$$\begin{aligned} p(\boldsymbol{x}_n \mid \boldsymbol{\mu}_k) &= \mathcal{N}(\boldsymbol{x}_n \mid \boldsymbol{\mu}_k, \boldsymbol{I}) && (7) \\ &= (2\pi)^{-D/2} \exp\left\{-\tfrac{1}{2}(\boldsymbol{x}_n - \boldsymbol{\mu}_k)^\top (\boldsymbol{x}_n - \boldsymbol{\mu}_k)\right\} && (8) \\ &= (2\pi)^{-D/2} \exp\left\{-\tfrac{1}{2}\boldsymbol{x}_n^\top \boldsymbol{x}_n + \boldsymbol{x}_n^\top \boldsymbol{\mu}_k - \tfrac{1}{2}\boldsymbol{\mu}_k^\top \boldsymbol{\mu}_k\right\}, && (9) \end{aligned}$$

which is an exponential family distribution with base measure $h(\boldsymbol{x}_n) = (2\pi)^{-D/2} e^{-\frac{1}{2}\boldsymbol{x}_n^\top \boldsymbol{x}_n}$, sufficient statistics $t(\boldsymbol{x}_n) = \boldsymbol{x}_n$, and log normalizer $A(\boldsymbol{\mu}_k) = \frac{1}{2}\boldsymbol{\mu}_k^\top \boldsymbol{\mu}_k$.

The conjugate prior is a Gaussian prior on the mean,

$$p(\boldsymbol{\mu}_k \mid \boldsymbol{\phi}, \nu) = \mathcal{N}(\nu^{-1}\boldsymbol{\phi}, \nu^{-1}\boldsymbol{I}) \propto \exp\left\{\boldsymbol{\phi}^\top \boldsymbol{\mu}_k - \tfrac{\nu}{2}\boldsymbol{\mu}_k^\top \boldsymbol{\mu}_k\right\} = \exp\left\{\boldsymbol{\phi}^\top \boldsymbol{\mu}_k - \nu A(\boldsymbol{\mu}_k)\right\}. \qquad (10)$$

Note that $\boldsymbol{\phi}$ sets the location and ν sets the precision (i.e. inverse variance).

EM in the Gaussian mixture model

EM was like K-Means but with **soft assignments**.

1. For each data point *n* and component *k*, set the *responsibility* to,

$$\omega_{nk} = \frac{\pi_k \mathcal{N}(\boldsymbol{x}_n \mid \boldsymbol{\mu}_k, \boldsymbol{I})}{\sum_{j=1}^{K} \pi_j \mathcal{N}(\boldsymbol{x}_n \mid \boldsymbol{\mu}_j, \boldsymbol{I})}. \tag{11}$$

2. For each component *k*, set the new mean to

$$\boldsymbol{\mu}_k^\star = \frac{1}{N_k} \sum_{n=1}^{N} \omega_{nk} \boldsymbol{x}_n, \tag{12}$$

where $N_k = \sum_{n=1}^{N} \omega_{nk}$.

This gives us a **point estimate** of $\boldsymbol{\mu}_k$. What if we need a full posterior?

Taking stock of posterior inference algorithms thus far

We've covered a number of posterior inference algorithms thus far:

- **Exact inference:** for simple models (e.g. conjugate exponential family models) where the posterior is available in closed form.
- **Gibbs sampling:** an MCMC algorithm that iteratively samples conditional distributions for one variable at a time. This works well for conditionally conjugate models with weak correlations.
- **Metropolis-Hastings:** a very general MCMC algorithm to sample the posterior, and the building block for many other MCMC techniques.
- **Hamiltonian Monte Carlo:** an MCMC algorithm to draw samples from the posterior by leveraging gradients of the log joint probability. This works well for more general posteriors over continuous variables.

Question: which of these algorithms could we use for the Gaussian mixture model?

Variational inference

MCMC methods are asymptotically unbiased (though for finite samples there is a transient bias that shrinks as $O(S^{-1})$. The real issue is variance: it only shrinks as $O(S^{-1/2})$.

Motivation: With finite computation, can we get better posterior estimates by trading asymptotic bias for smaller variance?

Idea: approximate the posterior by with a simple, parametric form (though not strictly a Gaussian on the mode!). Optimize to find the approximation that is as "close" as possible to the posterior.

Notation

This notation could be a bit confusing. Let,

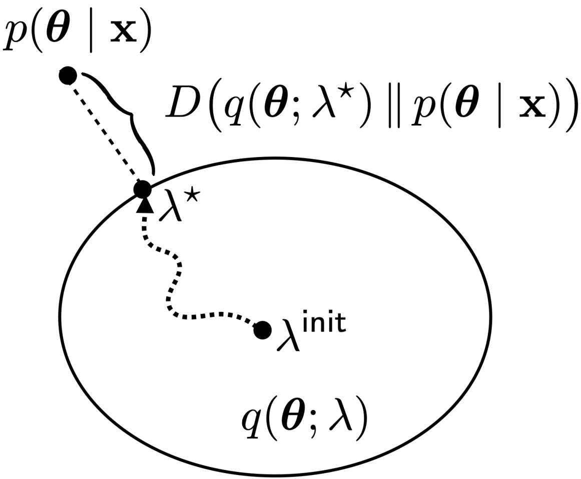
- $\boldsymbol{\theta} \in \mathbb{R}^J$ denote **all of latent variables and parameters** we wish to infer.
- In the GMM, $\boldsymbol{\theta} = (\boldsymbol{\pi}, \{\boldsymbol{\mu}_k\}_{k=1}^K, \{z_n\}_{n=1}^N)$.

In contrast to last week, here we will obtain a **full posterior over parameters and latent variables**.

Likewise, let

- $p(\boldsymbol{\theta} \mid \boldsymbol{x})$ denote the true posterior distribution we want to approximate.
- $q(\boldsymbol{\theta}; \boldsymbol{\lambda})$ denote a parametric *variational approximation* to the posterior where...
- $\boldsymbol{\lambda}$ denotes the *variational parameters* that we will optimize.
- $D(q \,\|\, p)$ denote a *divergence measure* that takes in two distributions q and p and returns a measure of how similar they are.

A view of variational inference


$p(\boldsymbol{\theta} \mid \mathbf{x})$
$D\big(q(\boldsymbol{\theta}; \lambda^{\star}) \,\|\, p(\boldsymbol{\theta} \mid \mathbf{x})\big)$
$\lambda^{\star}$
λ^{init}
$q(\boldsymbol{\theta}; \lambda)$

Key questions

- *What parametric family should we use?*
  - This lecture: the **mean-field family**.
- *How should we measure closeness?*
  - This lecture: the **Kullback-Leibler (KL)** divergence.
- *How do we find the closest distribution in that family?*
  - This lecture: **coordinate ascent.**

These choices are what Blei et al. [2017] call **coordinate ascent variational inference** CAVI).

The mean-field family

The *mean-field family* gets its name from statistical mechanics. It treats each latent variable and parameter as independent with its own variational parameter,

$$q(\boldsymbol{\theta}\,;\boldsymbol{\lambda}) = \prod_{j=1}^{J} q(\theta_j;\lambda_j). \tag{13}$$

For example, in the GMM, the mean field approximation treats the cluster proportions, means, and assignments as independent,

$$q(\boldsymbol{\theta}\,;\boldsymbol{\lambda}) = q(\boldsymbol{\pi};\widetilde{\boldsymbol{\alpha}})\prod_{k=1}^{K} q(\boldsymbol{\mu}_k;\widetilde{\nu}_k,\widetilde{\boldsymbol{\phi}}_k)\prod_{n=1}^{N} q(z_n;\widetilde{\boldsymbol{\omega}}_n) \tag{14}$$

for some functional forms and parameterizations that we will specify shortly.

Question: Does the true posterior factor in this way? If not, why not?

The Kullback-Leibler (KL) divergence

The KL divergence is a measure of closeness between two distributions. It is defined as,

$$\begin{aligned} D_{\mathrm{KL}}\left(q(\boldsymbol{\theta}\,;\boldsymbol{\lambda}) \,\|\, p(\boldsymbol{\theta} \mid \boldsymbol{x}\right) &= \mathbb{E}_{q(\boldsymbol{\theta}\,;\boldsymbol{\lambda})}\left[\log \frac{q(\boldsymbol{\theta}\,;\boldsymbol{\lambda})}{p(\boldsymbol{\theta} \mid \boldsymbol{x})}\right] \qquad (15) \\ &= \mathbb{E}_{q(\boldsymbol{\theta}\,;\boldsymbol{\lambda})}\left[\log q(\boldsymbol{\theta}\,;\boldsymbol{\lambda})\right] - \mathbb{E}_{q(\boldsymbol{\theta}\,;\boldsymbol{\lambda})}\left[\log p(\boldsymbol{\theta} \mid \boldsymbol{x})\right] \qquad (16) \end{aligned}$$

It has some nice properties:

- It is non-negative.
- It is zero iff $q(\boldsymbol{\theta}\,;\boldsymbol{\lambda}) \equiv p(\boldsymbol{\theta} \mid \boldsymbol{x})$.
- It is defined in terms of expectations wrt q.

But it's also a bit weird...

- It's asymmetric ($D_{\mathrm{KL}}\left(q \,\|\, p\right) \neq D_{\mathrm{KL}}\left(p \,\|\, q\right)$).

The evidence lower bound (ELBO) from another angle

More concerning, the KL divergence involves the posterior $p(\boldsymbol{\theta} \mid \boldsymbol{x})$, which we cannot compute!

But notice that...

$$\begin{aligned} D_{\mathrm{KL}}\left(q(\boldsymbol{\theta};\boldsymbol{\lambda}) \,\|\, p(\boldsymbol{\theta} \mid \boldsymbol{x})\right) &= \mathbb{E}_{q(\boldsymbol{\theta};\boldsymbol{\lambda})}\left[\log q(\boldsymbol{\theta};\boldsymbol{\lambda})\right] - \mathbb{E}_{q(\boldsymbol{\theta};\boldsymbol{\lambda})}\left[\log p(\boldsymbol{\theta} \mid \boldsymbol{x})\right] && (17) \\ &= \mathbb{E}_{q(\boldsymbol{\theta};\boldsymbol{\lambda})}\left[\log q(\boldsymbol{\theta};\boldsymbol{\lambda})\right] - \mathbb{E}_{q(\boldsymbol{\theta};\boldsymbol{\lambda})}\left[\log p(\boldsymbol{\theta},\boldsymbol{x})\right] + \mathbb{E}_{q(\boldsymbol{\theta};\boldsymbol{\lambda})}\left[\log p(\boldsymbol{x})\right] && (18) \\ &= \underbrace{\mathbb{E}_{q(\boldsymbol{\theta};\boldsymbol{\lambda})}\left[\log q(\boldsymbol{\theta};\boldsymbol{\lambda})\right] - \mathbb{E}_{q(\boldsymbol{\theta};\boldsymbol{\lambda})}\left[\log p(\boldsymbol{\theta},\boldsymbol{x})\right]}_{\text{negative ELBO,}-\mathscr{L}(\boldsymbol{\lambda})} + \underbrace{\log p(\boldsymbol{x})}_{\text{evidence}} && (19) \end{aligned}$$

The first term involves the log joint, which we can compute, and the last term is independent of the variational parameters!

Rearranging, we see that $\mathscr{L}(\boldsymbol{\lambda})$ is a lower bound on the marginal likelihood, aka the evidence,

$$\mathscr{L}(\boldsymbol{\lambda}) = \log p(\boldsymbol{x}) - D_{\mathrm{KL}}\left(q(\boldsymbol{\theta};\boldsymbol{\lambda}) \,\|\, p(\boldsymbol{\theta} \mid \boldsymbol{x})\right) \leq \log p(\boldsymbol{x}). \tag{20}$$

That's why we call it the **evidence lower bound (ELBO)**.

Viewer discretion advised...

https://www.youtube.com/watch?v=jugUBL4rEIM

Optimizing the ELBO with coordinate ascent

We want to find the variational parameters $\boldsymbol{\lambda}$ that minimize the KL divergence or, equivalently, maximize the ELBO.

For the mean-field family, we can typically do this via **coordinate ascent**.

Consider optimizing the parameters for one factor $q(\theta_j; \lambda_j)$. As a function of λ_j, the ELBO is,

$$\begin{aligned}
\mathscr{L}(\boldsymbol{\lambda}) &= \mathbb{E}_{q(\theta_j;\lambda_j)}\left[\mathbb{E}_{q(\boldsymbol{\theta}_{\neg j};\boldsymbol{\lambda}_{\neg j})}\left[\log p(\boldsymbol{\theta}, \boldsymbol{x})\right]\right] - \mathbb{E}_{q(\theta_j;\lambda_j)}[\log q(\theta_j; \lambda_j)] + c && (21) \\
&= \mathbb{E}_{q(\theta_j;\lambda_j)}\left[\mathbb{E}_{q(\boldsymbol{\theta}_{\neg j};\boldsymbol{\lambda}_{\neg j})}\left[\log p(\theta_j \mid \boldsymbol{\theta}_{\neg j}, \boldsymbol{x})\right]\right] - \mathbb{E}_{q(\theta_j;\lambda_j)}[\log q(\theta_j; \lambda_j)] + c' && (22) \\
&= -D_{\mathrm{KL}}\left(q(\theta_j; \lambda_j) \,\|\, \tilde{p}(\theta_j)\right) + c'' && (23)
\end{aligned}$$

where

$$\tilde{p}(\theta_j) \propto \exp\left\{\mathbb{E}_{q(\boldsymbol{\theta}_{\neg j};\boldsymbol{\lambda}_{\neg j})}\left[\log p(\theta_j \mid \boldsymbol{\theta}_{\neg j}, \boldsymbol{x})\right]\right\} \tag{24}$$

The ELBO is maximized wrt λ_j when this KL is minimized; i.e. when $q(\theta_j; \lambda_j) = \tilde{p}(\theta_j)$, the exponentiated expected log conditional probability, holding all other factors fixed.

Coordinate Ascent Variational Inference for GMMs

Let's derive the CAVI updates for a Gaussian mixture model (GMM).

Assume a mean field family, and assume each factor is of the same exponential family form as the corresponding prior:

$$q(z_n; \widetilde{\boldsymbol{\omega}}_n) = \mathrm{Cat}(z_n; \widetilde{\boldsymbol{\omega}}_n) \tag{25}$$
$$q(\boldsymbol{\pi}; \widetilde{\boldsymbol{\alpha}}) = \mathrm{Dir}(\boldsymbol{\pi}; \widetilde{\boldsymbol{\alpha}}) \tag{26}$$
$$q(\boldsymbol{\mu}_k; \widetilde{\nu}_k, \widetilde{\boldsymbol{\phi}}_k) = \mathcal{N}(\boldsymbol{\mu}_k; \widetilde{\nu}_k^{-1}\widetilde{\boldsymbol{\phi}}_k, \widetilde{\nu}_k^{-1}\boldsymbol{I}). \tag{27}$$

so $\widetilde{\boldsymbol{\omega}}_n \in \Delta_K$, $\widetilde{\boldsymbol{\alpha}} \in \mathbb{R}_+^K$, $\widetilde{\nu}_k \in \mathbb{R}_+$ and $\widetilde{\boldsymbol{\phi}}_k \in \mathbb{R}^D$ constitute the set of variational parameters, $\boldsymbol{\lambda}$.

(Here we've assumed the functional form of the variational posteriors, but for conjugate exponential family models, it turns out the optimal variational factors are of the same form as the prior anyway!)

CAVI updates for the cluster assignments

Recall that the optimal CAVI updates are of the form in Eq. 24.

For the cluster assignments, the CAVI update is,

$$\begin{aligned}
\log q(z_n = k; \boldsymbol{\widetilde{\omega}}_n) &= \mathbb{E}_{q(\pi)q(\boldsymbol{\mu}_k)}\left[\log \pi_k + \log \mathcal{N}(\boldsymbol{x}_n \mid \boldsymbol{\mu}_k, \boldsymbol{I})\right] + c \qquad (28)\\
&= \mathbb{E}_{q(\pi_n)}\left[\log \pi_k\right] + \mathbb{E}_{q(\boldsymbol{\mu}_k)}\left[\log \mathcal{N}(\boldsymbol{x}_n \mid \boldsymbol{\mu}_k, \boldsymbol{I})\right] + c \qquad (29)\\
&= \log \mathrm{Cat}(z_n = k; \boldsymbol{\widetilde{\omega}}_n) \qquad (30)\\
\Rightarrow \log \widetilde{\omega}_{n,k} &= \mathbb{E}_{q(\pi)}\left[\log \pi_k\right] + \mathbb{E}_{q(\boldsymbol{\mu}_k)}\left[\log \mathcal{N}(\boldsymbol{x}_n \mid \boldsymbol{\mu}_k, \boldsymbol{I})\right] + c \qquad (31)
\end{aligned}$$

Since $\boldsymbol{\widetilde{\omega}}_n$ must sum to one,

$$\widetilde{\omega}_{n,k} = \frac{\exp\left\{\mathbb{E}_{q(\pi)}\left[\log \pi_k\right] + \mathbb{E}_{q(\boldsymbol{\mu}_k)}\left[\log \mathcal{N}(\boldsymbol{x}_n \mid \boldsymbol{\mu}_k, \boldsymbol{I})\right]\right\}}{\sum_{j=1}^{K} \exp\left\{\mathbb{E}_{q(\pi)}\left[\log \pi_j\right] + \mathbb{E}_{q(\boldsymbol{\mu}_j)}\left[\log \mathcal{N}(\boldsymbol{x}_n \mid \boldsymbol{\mu}_j, \boldsymbol{I})\right]\right\}} \tag{32}$$

Expectations under Dirichlet distributions

The variational factor for $\boldsymbol{\pi}$ is a Dirichlet distribution.

The necessary expectation has a closed form expression:

$$\mathbb{E}_{\mathrm{Dir}(\boldsymbol{\pi};\boldsymbol{\alpha})}[\log \pi_k] = \psi(\alpha_k) - \psi\Big(\sum_{j=1}^{K} \alpha_j\Big) \tag{33}$$

where $\psi(\cdot)$ is the *digamma function*, the logarithmic derivative of the gamma function.

Gaussian cross-entropy

The updates for $q(z_n)$ also require a Gaussian cross entropy,

$$\mathbb{E}_{\mathcal{N}(\boldsymbol{\mu}; \boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0)}[\log \mathcal{N}(\boldsymbol{x} \mid \boldsymbol{\mu}, \boldsymbol{\Sigma})] \tag{34}$$

Exercise: Derive an expression for the Gaussian cross entropy.

CAVI updates for the cluster proportions

The CAVI update is„

$$\begin{aligned}
\log q(\boldsymbol{\pi}; \widetilde{\boldsymbol{\alpha}}) &= \log \mathrm{Dir}(\boldsymbol{\pi}; \boldsymbol{\alpha}) + \sum_{n=1}^{N} \sum_{k=1}^{K} \mathbb{E}_{q(z_n)}[\mathbb{I}[z_n = k]] \log \pi_k + c && (35) \\
&= \sum_{k=1}^{K} \left(\alpha_k - 1 + \sum_{n=1}^{N} \widetilde{\omega}_{n,k} \right) \log \pi_k && (36) \\
&= \log \mathrm{Dir}(\boldsymbol{\pi}; \widetilde{\boldsymbol{\alpha}}) && (37)
\end{aligned}$$

where

$$\widetilde{\alpha}_k = \alpha_k + \sum_{n=1}^{N} \widetilde{\omega}_{n,k} \qquad (38)$$

CAVI updates for the cluster means

The cluster mean updates are similar

$$\log q(\boldsymbol{\mu}_k; \widetilde{\nu}_k, \widetilde{\boldsymbol{\phi}}_k) = \log \mathcal{N}(\boldsymbol{\mu}_k; \nu^{-1}\boldsymbol{\phi}, \nu^{-1}\boldsymbol{I}) + \sum_{n=1}^{N} \mathbb{E}_{q(z_n)}[\mathbb{I}[z_n = k]] \log \mathcal{N}(\boldsymbol{x}_n \mid \boldsymbol{\mu}_k, \boldsymbol{I}) + c \tag{39}$$

$$= \langle \boldsymbol{\phi}, \boldsymbol{\mu}_k \rangle + \nu A(\boldsymbol{\mu}_k) + \sum_{n=1}^{N} \mathbb{E}_{q(z_n)}[\mathbb{I}[z_n = k]] \left(\langle \boldsymbol{x}_n, \boldsymbol{\mu}_k \rangle - A(\boldsymbol{\mu}_k)\right) + c \tag{40}$$

$$= \log \mathcal{N}(\boldsymbol{\mu}_k \mid \widetilde{\nu}_k^{-1} \widetilde{\boldsymbol{\phi}}_k, \widetilde{\nu}_k^{-1}\boldsymbol{I}) \tag{41}$$

where

$$\widetilde{\boldsymbol{\phi}}_k = \boldsymbol{\phi} + \sum_{n=1}^{N} \widetilde{\omega}_{n,k} \boldsymbol{x}_n \tag{42}$$

$$\widetilde{\nu}_k = \nu + \sum_{n=1}^{N} \widetilde{\omega}_{n,k} \tag{43}$$

Calculating the ELBO

Dropping hyperparameters and variational parameters, the ELBO is,

$$\mathcal{L}(\boldsymbol{\lambda}) = \mathbb{E}_q[\log p(\{\boldsymbol{x}_n, z_n\}_{n=1}^N, \{\boldsymbol{\mu}_k\}_{k=1}^K, \pi)] - \mathbb{E}_q[\log q(\{z_n\}_{n=1}^N, \{\boldsymbol{\mu}_k\}_{k=1}^K, \pi)] \tag{44}$$

Thanks to the factorization of the joint distribution and the variational posterior, this simplifies,

$$\begin{aligned}\mathcal{L}(\boldsymbol{\lambda}) = &\sum_{n=1}^N \sum_{k=1}^K \mathbb{E}_{q(z_n)}[\mathbb{I}[z_n = k]]\mathbb{E}_{q(\boldsymbol{\mu}_k)}[\log \mathcal{N}(\boldsymbol{x}_n \mid \boldsymbol{\mu}_k, \boldsymbol{I})] \\ &- \sum_{n=1}^N \mathbb{E}_{q(z_n)}[\log q(z_n)] - \sum_{k=1}^K D_{\mathrm{KL}}(q(\boldsymbol{\mu}_k) \parallel p(\boldsymbol{\mu}_k)) - D_{\mathrm{KL}}(q(\pi) \parallel p(\pi))\end{aligned} \tag{45}$$

These terms are all easy to compute! They’re just cross-entropies and KL divergences for exponential family distributions.

Scaling up to very large datasets

There are a few tricks to make CAVI scale to massive datasets.

You can process data points in rolling fashion since we just need sums of expected sufficient statistics.

Likewise, you can use **stochastic variational inference** [Hoffman et al., 2013] to work with mini-batches of documents to get Monte Carlo estimates of the ELBO, since it includes a big sum over data points.

SVI can be seen as **stochastic gradient ascent** on the ELBO using **natural gradients** Amari [1998]; i.e., gradient descent preconditioned with the Fisher information matrix.

References I

David M Blei, Alp Kucukelbir, and Jon D McAuliffe. Variational inference: A review for statisticians. *Journal of the American Statistical Association*, 112(518):859–877, April 2017.

Matthew D Hoffman, David M Blei, Chong Wang, and John Paisley. Stochastic variational inference. *Journal of Machine Learning Research*, 14(5), 2013.

Shun-Ichi Amari. Natural gradient works efficiently in learning. *Neural computation*, 10(2):251–276, 1998.