Chapter 16

Variational Inference *

Consider the inference problem where our objective is to compute the probability distribution of unknown parameters (Bayesian inference) or latent variables $\boldsymbol{Z}$, conditioned on observations $\boldsymbol{X} = \boldsymbol{x}$. Mathematically, this is represented as

$$p(\boldsymbol{z}|\boldsymbol{x}) = \frac{p(\boldsymbol{x}, \boldsymbol{z})}{\int p(\boldsymbol{x}, \boldsymbol{z}) d\boldsymbol{z}} \tag{16.1}$$

Here, the joint distribution $p(\boldsymbol{x}, \boldsymbol{z})$ is typically known, either explicitly defined by the model or through combining $p(\boldsymbol{x}|\boldsymbol{z})$ and $p(\boldsymbol{z})$. However, the integral $\int p(\boldsymbol{x}, \boldsymbol{z}) d\boldsymbol{z}$, is challenging to compute, especially in high-dimensional spaces, as it sums over all possible configurations of the latent variables $\boldsymbol{z}$.

While Monte-Carlo methods enable sampling from the target distribution $p(\boldsymbol{z}|\boldsymbol{x})$, they are often computationally intensive. Variational inference, on the other hand, offers a computationally efficient alternative by approximating $p(\boldsymbol{z}|\boldsymbol{x})$ through optimization.

Variational inference simplifies the process by approximating the complex posterior distribution $p(\boldsymbol{z}|\boldsymbol{x})$ with a more tractable distribution $q(\boldsymbol{z})$ from a predefined family Q. The objective is to identify $q^* \in Q$ that is closest to $p(\boldsymbol{z}|\boldsymbol{x})$ as measured by the KL-divergence, $D_{KL}(q(\boldsymbol{z})||p(\boldsymbol{z}|\boldsymbol{x}))$. The optimization problem thus formulated is

$$q^* = \arg\min_{q \in Q} D_{KL}(q(\boldsymbol{z})||p(\boldsymbol{z}|\boldsymbol{x})), \tag{16.2}$$

where

$$D_{KL}(q(\boldsymbol{z})||p(\boldsymbol{z}|\boldsymbol{x})) = \int q(\boldsymbol{z}) \log \frac{q(\boldsymbol{z})}{p(\boldsymbol{z}|\boldsymbol{x})} d\boldsymbol{z} = \mathbb{E}_q \left[\log \frac{q(\boldsymbol{Z})}{p(\boldsymbol{Z}|\boldsymbol{x})} \right], \tag{16.3}$$

with $\mathbb{E}_q$ denoting that the expectation assumes distribution q for $\boldsymbol{Z}$. In variational inference, minimizing $D_{KL}(q(\boldsymbol{z})||p(\boldsymbol{z}|\boldsymbol{x}))$ is preferred over minimizing $D_{KL}(p(\boldsymbol{z}|\boldsymbol{x})||q(\boldsymbol{z}))$ because the latter is typically intractable.

Evidence Lower Bound (ELBO): Reformulating the KL-divergence yields:

$$\begin{aligned} D_{KL}(q(\boldsymbol{z})||p(\boldsymbol{z}|\boldsymbol{x})) &= \int q(\boldsymbol{z}) \log \frac{q(\boldsymbol{z})}{p(\boldsymbol{z}|\boldsymbol{x})} d\boldsymbol{z} && (16.4) \\ &= \int q(\boldsymbol{z}) \log \frac{q(\boldsymbol{z})p(\boldsymbol{x})}{p(\boldsymbol{z}, \boldsymbol{x})} d\boldsymbol{z} && (16.5) \\ &= \int q(\boldsymbol{z}) \log \frac{q(\boldsymbol{z})}{p(\boldsymbol{z}, \boldsymbol{x})} d\boldsymbol{z} + \log p(\boldsymbol{x}). && (16.6) \end{aligned}$$

Minimizing the KL-divergence, in this context, is equivalent to maximizing the Evidence Lower Bound (ELBO), defined as:

$$\mathcal{L}(q) = \log p(\boldsymbol{x}) - D_{KL}(q(\boldsymbol{z})||p(\boldsymbol{z}|\boldsymbol{x})) \tag{16.7}$$

$$= \int q(\boldsymbol{z}) \log \frac{p(\boldsymbol{x}, \boldsymbol{z})}{q(\boldsymbol{z})} d\boldsymbol{z} \tag{16.8}$$

$$= \int q(\boldsymbol{z}) \log p(\boldsymbol{x}, \boldsymbol{z}) d\boldsymbol{z} + H(q). \tag{16.9}$$

Note that since $D_{KL} \geq 0$, ELBO is no greater than $\log p(\boldsymbol{x})$, i.e., $\log p(\boldsymbol{x}) \geq \mathcal{L}(q)$. As $\boldsymbol{x}$ is sometimes referred to as *evidence*, this observation motivates the name “Evidence Lower Bound” or ELBO.

Why does maximizing $\mathcal{L}(q)$ make sense? Let us inspect each term in $\mathcal{L}(q)$. The distribution q that maximizes $\int q(\boldsymbol{z}) \log p(\boldsymbol{x}, \boldsymbol{z}) d\boldsymbol{z}$ is the one that puts all the probability mass on $\hat{\boldsymbol{z}} = \arg\max_{\boldsymbol{z}} \log p(\boldsymbol{x}, \boldsymbol{z}) = \arg\max_{\boldsymbol{z}} \log p(\boldsymbol{z}|\boldsymbol{x})$, i.e., the Bayesian mode point estimator. This is a degenerate distribution that tells us that $\boldsymbol{Z}$ is equal to $\hat{\boldsymbol{z}}$ with probability 1. To balance this overconfidence, q that maximizes the second term, $H(q)$, must be high-entropy.

Alternatively, we can rewrite ELBO in the following way to gain more intuition about why maximizing the ELBO gives a reasonable approximation [1]. By (16.8),

$$\mathcal{L}(q) = \int q(\boldsymbol{z}) \log \frac{p(\boldsymbol{z})}{q(\boldsymbol{z})} d\boldsymbol{z} + \int q(\boldsymbol{z}) \log p(\boldsymbol{x}|\boldsymbol{z}) d\boldsymbol{z} \tag{16.10}$$

$$= -D_{KL}(q(\boldsymbol{z})||p(\boldsymbol{z})) + \int q(\boldsymbol{z}) \log p(\boldsymbol{x}|\boldsymbol{z}) d\boldsymbol{z}. \tag{16.11}$$

The two terms are now the negative KL divergence between $q(\boldsymbol{z})$ and the prior $p(\boldsymbol{z})$, and the expected likelihood assuming $\boldsymbol{Z} \sim q(\boldsymbol{z})$. For the divergence to be small, $q(\boldsymbol{z})$ is encouraged to be close to the prior. On the other hand, for the expected likelihood to be large, $q(\boldsymbol{z})$ should assign more mass to $\boldsymbol{Z}$ that can better explain our observed data $\boldsymbol{x}$, i.e., $\arg\max_{\boldsymbol{z}} p(\boldsymbol{x}|\boldsymbol{z})$. So, the solution balances closeness to the prior with the maximum-likelihood solution, similar to the true posterior.

16.1 Background on Calculus of Variations

Before proceeding further, we review the calculus of variations, a mathematical area that focuses on determining functions that optimize a *functional*. A functional $F : \mathcal{F} \to \mathbb{R}$ is a function that maps the elements of a specified family $\mathcal{F}$ of functions to $\mathbb{R}$. In variational inference, the functional is $D_{KL}(q(\boldsymbol{z})||p(\boldsymbol{z}|\boldsymbol{x}))$, which assigns a real number to each choice of $q \in \mathcal{Q}$.

A simple class of functionals are those of the form $\int_S J(x, f(x)) dz$ for some function J and set S. For entropy,

$$H[p] = \int_{\mathcal{X}} p(x) \log \frac{1}{p(x)} d(x),$$

we have $J(x, p) = J(p) = p \log \frac{1}{p}$. For KL divergence

$$D_{KL}(q||p) = \int q(x) \log \frac{q(x)}{p(x)},$$

viewed as a functional of q and for fixed p, we have $J(x, q) = q \log \frac{q}{p(x)}$.

The **functional differential** of F at f in the direction of ϕ is defined as

$$\lim_{\epsilon \to 0} \frac{F[f + \epsilon\phi] - F[f]}{\epsilon} = \left. \frac{dF[f + \epsilon\phi]}{d\epsilon} \right|_{\epsilon=0}$$

This tells us how the functional changes at f if it is perturbed by moving infinitesimally in the “direction” of ϕ. This quantity is useful for optimizing a functional, that is, for finding a function that maximizes or minimizes the functional. We will explore this concept through an analogy to multivariate calculus.

Analogy from multivariate calculus: Consider $g : \mathbb{R}^n \to \mathbb{R}$, a function that assigns to each vector $\boldsymbol{x} \in \mathbb{R}^n$ a real number $g(\boldsymbol{x})$. If we are at $\boldsymbol{x}$, how does g change if we move in the direction of some vector $\boldsymbol{v}$? The change in g can be quantified by

$$\lim_{\epsilon \to 0} \frac{g(\boldsymbol{x}+\epsilon\boldsymbol{v}) - g(\boldsymbol{x})}{\epsilon}$$

This is useful for optimizing g. For instance, if it is 0 for all $\boldsymbol{v}$, then we are at a local extremum. But for each vector $\boldsymbol{v}$, we would need to compute it from scratch. To address this, we define the gradient

$$\nabla g = \left(\lim_{\epsilon \to 0} \frac{g(\boldsymbol{x}+\epsilon\mathbf{i}_1) - g(\boldsymbol{x})}{\epsilon}, \dots, \lim_{\epsilon \to 0} \frac{g(x+\epsilon\mathbf{i}_n) - g(\boldsymbol{x})}{\epsilon}\right),$$

where $\mathbf{i}_1, \dots, \mathbf{i}_n$ are unit vectors in the standard basis. Then we can find the rate of change for any vector $\boldsymbol{v}$ as the inner product of the gradient and the $\boldsymbol{v}$

$$\lim_{\epsilon \to 0} \frac{g(\boldsymbol{x}+\epsilon\boldsymbol{v}) - g(\boldsymbol{x})}{\epsilon} = \langle \nabla g, \boldsymbol{v} \rangle.$$

Furthermore, for small ϵ,

$$g(\boldsymbol{x}+\epsilon\boldsymbol{v}) \simeq g(\boldsymbol{x}) + \epsilon\langle \nabla g, \boldsymbol{v} \rangle.$$

Back to functionals: For functionals of the form $F[f] = \int_S J(x, f(x))dx$, we can find something similar to a gradient. Specifically, there is a function $\frac{\partial F}{\partial f}(x)$, called a **functional derivative** such that

$$\left.\frac{dF[f+\epsilon\phi]}{d\epsilon}\right|_{\epsilon=0} = \int_{\mathcal{X}} \frac{\partial F}{\partial f}(x)\phi(x)dx$$

This derivative measures how the functional $F[f]$ changes when the function f is perturbed infinitesimally at the point x.

The Defining the inner product in the space of functions as $\langle f(x), g(x) \rangle = \int_{\mathcal{X}} f(x)g(x)dx$ for some predetermined set $\mathcal{X}$, we can write

$$\left.\frac{dF[f+\epsilon\phi]}{d\epsilon}\right|_{\epsilon=0} \langle \frac{\partial F}{\partial f}(x), \phi(x) \rangle$$
$$F[f+\epsilon\phi] \simeq F[f] + \epsilon\langle \frac{\partial F}{\partial f}(x), \phi(x) \rangle$$

Observe that

$$\begin{aligned}\left.\frac{dF[f+\epsilon\phi]}{d\epsilon}\right|_{\epsilon=0} &= \left.\frac{d}{d\epsilon}\int J(x, f(x)+\epsilon\phi(x))dx\right|_{\epsilon=0}\\ &= \left.\int \frac{d}{d\epsilon} J(x, f(x)+\epsilon\phi(x))dx\right|_{\epsilon=0}\\ &= \int J_2(x, f(x))\phi(x)dx,\end{aligned}$$

where J_2 is the prtial dervative of J with respect to its second argument.

Hence,

$$\frac{\partial F}{\partial f}(x) = J_2(x, f(x))$$

Example 16.1. Let us find $\frac{\partial H[p]}{\partial p}(x)$ where H is the entropy function. Here, we have $J(x,p) = p \log \frac{1}{p}$. Hence,

$$\frac{\partial H[p]}{\partial p}(x) = \frac{\partial(p\log\frac{1}{p})}{\partial p}(x) = \log\frac{1}{p(x)} - 1.$$

△

Example 16.2. For fixed p, let us find $\frac{\partial D_{KL}(q||p)}{\partial q}(x)$. Here, we have $J(x,q) = q \log \frac{q}{p(x)}$. Hence,

$$\frac{\partial D_{KL}}{\partial q}(x) = \frac{\partial (q \log \frac{q}{p(x)})}{\partial q}(x) = 1 + \log \frac{q(x)}{p(x)}.$$

△

Optimization of Functionals: Now that we have functional derivatives, we can optimized functionals by setting the derivative to 0. When we have constrained, we can use Lagrange multipliers.

Example 16.3. We find the distribution with the highest possible entropy with variance at most σ^2, i.e.,

$$\begin{aligned} &\text{maximize } H[p] \\ &\text{s.t. } S[p] = \int p(x)dx = 1 \\ &\qquad V[p] = \int p(x)x^2 dx = 1 \end{aligned}$$

Using Lagrange multipliers:

$$\frac{\partial H}{\partial p}(x) + \lambda_1 \frac{\partial S}{\partial p}(x) + \lambda_2 \frac{\partial V}{\partial p}(x) = 0$$

Hence,

$$\log \frac{1}{p(x)} + \lambda_1 + \lambda_2 x^2 = 0 \Rightarrow p(x) = e^{\lambda_1 + \lambda_2 x^2}.$$

This is a Gaussian distribution. Since we know which Gaussian distribution has variance $sigma^2$, we have

$$p(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{x^2/\sigma^2}.$$

We could also find the constants by solving the constraint equations. Note that the mean is arbitrary. △

Example 16.4. For fixed p, let us find the distribution that minimizes $D_{KL}(q||p)$, i.e.,

$$\begin{aligned} &\text{maximize } D_{KL}(q||p) \\ &\text{s.t. } S[q] = \int q(x)dx = 1 \end{aligned}$$

Again, using Lagrange multipliers, we have

$$1 + \log \frac{q(x)}{p(x)} + \lambda_1 = 0 \Rightarrow q(x) = q(x) \propto p(x),$$

which, along with the constraint, leads to

$$q(x) = p(x).$$

△

16.2 Mean-field variational inference

In this chapter, we restrict the "nice" family $\mathcal{Q}$ to be the family of distributions that factorize (being tractable is important!), i.e.,

$$q(\boldsymbol{z}) = \prod_{j=1}^{J} q_j(z_j), \tag{16.12}$$

where $z_1, z_2, \dots, z_J$ form a partition of all hidden variables in $\boldsymbol{z}$. This is called the **mean-field** approximation and leads to

$$\mathcal{L}(q) = \int \prod_{j=1}^{J} q_j(z_j) \log p(\boldsymbol{x}, \boldsymbol{z}) d\boldsymbol{z} + \sum_{j=1}^{J} H(q_j). \tag{16.13}$$

Coordinate ascent variational inference (CAVI)

$\mathcal{L}(q)$ is a functional of J functions. The most common way for optimizing (16.13) is coordinate ascent. In other words, we will take turns to optimize $\mathcal{L}(q)$ with respect to one component q_i while fixing the others $q_j, j \neq i$. Now, let us assume that we fix q_j for all $j \neq i$. We can write the ELBO as

$$\begin{aligned}
\mathcal{L}(q) &= \int \prod_{j=1}^{J} q_j(z_j) \log p(\boldsymbol{x}, \boldsymbol{z}) d\boldsymbol{z} + \sum_{j=1}^{J} H(q_j) && (16.14)\\
&= \sum_{j \neq i} H(q_j) + H(q_i) + \int q_i(z_i) \left(\int \prod_{j \neq i} q_j(z_j) \log p(\boldsymbol{x}, \boldsymbol{z}) d\boldsymbol{z}_{-i} \right) dz_i && (16.15)\\
&= \sum_{j \neq i} H(q_j) + H(q_i) + \int q_i(z_i) \tilde{f}_i(z_i) dz_i, && (16.16)
\end{aligned}$$

where $\boldsymbol{z}_{-i} = \{z_j\}_{j \neq i}$ and

$$\begin{aligned}
\tilde{f}_i(z_i) &= \int \prod_{j \neq i} q_j(z_j) \log p(\boldsymbol{x}, \boldsymbol{z}) d\boldsymbol{z}_{-i} && (16.17)\\
&= \int \boldsymbol{q}_{-i}(\boldsymbol{z}_{-i}) \log p(\boldsymbol{x}, \boldsymbol{z}) d\boldsymbol{z}_{-i} && (16.18)\\
&= \mathbb{E}_{\boldsymbol{z}_{-i} \sim \boldsymbol{q}_{-i}} [\log p(\boldsymbol{x}, \boldsymbol{z})], && (16.19)
\end{aligned}$$

where $\boldsymbol{q}_{-i} = \{q_j\}_{j \neq i}$.

Taking the derivative, we have

$$\frac{\partial \mathcal{L}(q)}{\partial q_i}(z_i) = \log \frac{1}{q_i(z_i)} - 1 + \tilde{f}_i(z_i) = 0 \Rightarrow q_i(z_i) \propto \exp(\tilde{f}_i(z_i)).$$

So we update q_i to

$$q_i^*(z_i) = \frac{\exp\Big(\tilde{f}_i(z_i)\Big)}{\int \exp\Big(\tilde{f}_i(z_i)\Big) dz_i}. \tag{16.20}$$

So $q_i^*(z_i)$ is also a distribution over z_i and since $\tilde{f}_i(z_i)$ is a function of z_i and does not depend on q_i, neither does $\int \exp\Big(\tilde{f}_i(z_i)\Big) dz_i$.

We summarize the above process in the following algorithm.

Algorithm 1 Coordinate ascent variational inference (CAVI)

```
**Input:** visible variables $\boldsymbol{x}$; latent variables $\boldsymbol{z} = (z_1, \dots, z_J)$; joint distribution $p(\boldsymbol{x}, \boldsymbol{z})$;
**Output:** an approximation for $p(\boldsymbol{z}|\boldsymbol{x})$;
Initialize distributions $q_1, \dots, q_J$ over $z_1, \dots, z_J$, respectively;
**while** not converged **do**
    **for** $i = 1$ to $J$ **do**
        $\tilde{f}_i(z_i) = \mathbb{E}_{\boldsymbol{z}_{-i} \sim \boldsymbol{q}_{-i}}[\log p(\boldsymbol{x}, \boldsymbol{z})]$;
        $q_i(z_i) = \frac{\exp\big(\tilde{f}_i(z_i)\big)}{\int \exp\big(\tilde{f}_i(z_i)\big) dz_i}$;
    **end for**
**end while**
```

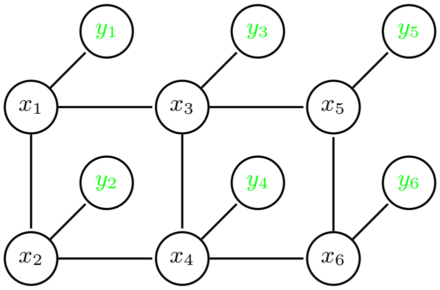
Note that the update rule (16.20) is given in the form of a function involving an integration. In actual implementation, we often derive a parametric form based on $q_i^*(z_i) \propto \exp\Big(\tilde{f}_i(z_i)\Big)$ and perform update over the "variational" parameters. Especially when variables z_i are discrete, we can always represent q_i by $k-1$ parameters, where k is the number of possible values that z_i can take.

16.3 Examples

Next, let us take a look at two examples, one in discrete case and the other in continuous case. The examples are adopted from [3] and [1].

16.3.1 CAVI on a MRF for image denoising

Consider the task of denoising an image using the following MRF



with energy function

$$E(\boldsymbol{x}, \boldsymbol{y}) = -\sum_{i=1}^{m} \alpha_i x_i - \sum_{(i,j) \in \mathcal{E}(G)} \beta_{i,j} x_i x_j - \sum_{i=1}^{m} \zeta_i x_i y_i,$$

where $\mathcal{E}(G)$ is the set of edges between neighboring pixels and $\beta_{i,j}, \zeta_i > 0$. In this task, the visible variables are the noisy pixels y_i and hidden variables are pixels x_i. All variables are discrete and take values in $\{+1, -1\}$.

To recover the original image based on its noisy version, let us apply CAVI to obtain the distribution of $\boldsymbol{x}$ given $\boldsymbol{y}$. The joint distribution $p(\boldsymbol{x}, \boldsymbol{y})$ is

$$p(\boldsymbol{x}, \boldsymbol{y}) = \frac{1}{Z} e^{-E(\boldsymbol{x}, \boldsymbol{y})}, \quad Z = \sum_{\boldsymbol{x}} \sum_{\boldsymbol{y}} e^{-E(\boldsymbol{x}, \boldsymbol{y})}. \tag{16.21}$$

We now assume a distribution $q(\boldsymbol{x})$ that factorizes:

$$q(\boldsymbol{x}) = \prod_{i=1}^{m} q_i(x_i). \tag{16.22}$$

Let $\mathbb{E}_{q_i}[x_i] = \mu_i$. Since every x_i takes two values, it suffices to optimize the ELBO over μ_i. We have

$$\begin{aligned}
\log q_i^*(x_i) &= \mathbb{E}_{\boldsymbol{q}_{-i}}[\log p(\boldsymbol{x}, \boldsymbol{y})] + \text{const} && (16.23)\\
&= \mathbb{E}_{\boldsymbol{q}_{-i}}[-E(\boldsymbol{x}, \boldsymbol{y}) - \log Z] + \text{const} && (16.24)\\
&= \mathbb{E}_{\boldsymbol{q}_{-i}}\left[\sum_i^m \alpha_i x_i + \sum_{(i,j)\in\mathcal{E}(G)} \beta_{i,j} x_i x_j + \sum_i^m \zeta_i x_i y_i - \log Z\right] + \text{const} && (16.25)\\
&= \mathbb{E}_{\boldsymbol{q}_{-i}}\left[\alpha_i x_i + \sum_{j\in\mathcal{E}(x_i)} \beta_{i,j} x_i x_j + \zeta_i x_i y_i\right] + \text{const} && (16.26)\\
&= \alpha_i x_i + \sum_{j\in\mathcal{E}(x_i)} \beta_{i,j} x_i \mu_j + \zeta_i x_i y_i + \text{const}, && (16.27)
\end{aligned}$$

where $\mathcal{E}(x_i)$ is the set of neighbors of x_i.

It follows that

$$q_i^*(x_i = 1) = \frac{e^{f_i}}{e^{f_i} + e^{-f_i}} = \frac{1}{1 + e^{-2f_i}}, \tag{16.28}$$

where $f_i = \alpha_i + \sum_{j\in\mathcal{E}(x_i)} \beta_{i,j}\mu_j + \zeta_i y_i$. Hence, the updating rules are given by

$$\mu_i^* = +1 \cdot q_i^*(x_i = 1) + (-1) \cdot q_i^*(x_i = -1) = \frac{1}{1 + e^{-2f_i}} - \frac{1}{1 + e^{2f_i}}. \tag{16.29}$$

16.3.2 Bayesian estimation of a univariate Gaussian [3]

Another application where we need to do inference about hidden variables given the visible ones is in Bayesian estimation. For a prior $p(\boldsymbol{\theta})$ and evidence $p(\mathcal{D}|\boldsymbol{\theta})$, we find an approximation for the posterior $p(\boldsymbol{\theta}|\mathcal{D}) \propto p(\boldsymbol{\theta})p(\mathcal{D}|\boldsymbol{\theta})$ by maximizing the ELBO

$$\mathcal{L}(q) = \log p(\mathcal{D}) - KL(q(\boldsymbol{\theta})||p(\boldsymbol{\theta}|\mathcal{D})) = \int q(\boldsymbol{\theta}) \log p(\mathcal{D}, \boldsymbol{\theta}) d\boldsymbol{\theta} + H(q). \tag{16.30}$$

univariate Guassian Consider Bayesian modeling of a univariate Gaussian. Let our data x follow a Gaussian distribution $\mathcal{N}(\mu, \lambda^{-1})$, where λ is the precision. Here we use precision λ as the parameter instead of the variance to simplify our computation.

The likelihood is thus given by

$$p(\mathcal{D}|\mu, \lambda) = \left(\frac{\lambda}{2\pi}\right)^{N/2} \prod_{n=1}^N \exp\left(-\frac{\lambda}{2}(x_n - \mu)^2\right). \tag{16.31}$$

We pick the conjugate *Gaussian-Gamma prior* of the form

$$\begin{aligned}
p(\lambda; a_0, b_0) &= \text{Gamma}(a_0, b_0) = \frac{\lambda^{a_0 - 1}\exp(-b_0\lambda) b_0^{a_0}}{\Gamma(a_0)}, && (16.32)\\
p(\mu|\lambda; \mu_0, \kappa_0) &= \mathcal{N}\left(\mu_0, (\kappa_0\lambda)^{-1}\right) = \left(\frac{\kappa_0\lambda}{2}\right)^{1/2} \exp\left(-\frac{\kappa_0\lambda}{2}(\mu - \mu_0)^2\right), && (16.33)\\
p(\mu, \lambda; \mu_0, \kappa_0, a_0, b_0) &= \text{GaussGamma}(\mu_0, \kappa_0, a_0, b_0) && (16.34)\\
&\propto \lambda^{a_0 - \frac{1}{2}} \exp(-b_0\lambda) \exp\left(-\frac{\kappa_0}{2}(\mu - \mu_0)^2\lambda\right). && (16.35)
\end{aligned}$$

Then, $\mathbb{E}\,\lambda = a_0/b_0, \mathbb{E}\,\mu = \mu_0, \text{Var}[\lambda] = a_0/b_0^2, \text{Var}[\mu] = b_0/(\kappa_0(a_0 - 1))$.

We are interested in the posterior

$$p(\mu, \lambda|\mathcal{D}) \propto p(\mu, \lambda)p(\mathcal{D}|\mu, \lambda). \tag{16.36}$$

Exact posterior ** The exact posterior can be shown to be

$$p(\mu,\lambda|\mathcal{D}) = \text{GaussGamma}\left(\frac{\kappa_0\mu_0 + N\bar{x}}{\kappa_0+N}, \kappa_0+N, a_0+\frac{N}{2}, b_0+\frac{1}{2}\left(Ns+\frac{\kappa_0 N(\bar{x}-\mu_0)^2}{\kappa_0+N}\right)\right), \tag{16.37}$$

where $\bar{x} = \frac{1}{N}\sum_{n=1}^N x_n$, $s = \frac{1}{N}\sum_{n=1}^N (x_n-\bar{x})^2$.

Approximate posterior Next, we approximate $p(\mu,\lambda|\mathcal{D})$ by

$$q(\mu,\lambda) = q_\mu(\mu)q_\lambda(\lambda). \tag{16.38}$$

Let us derive the updating rules needed by CAVI. Suppose we begin with two guesses $q_\mu(\mu)$ and $q_\lambda(\lambda)$. By (16.20),

$$\begin{aligned}
\log q_\mu^*(\mu) &= \mathbb{E}_{q_\lambda}[\log p(\mathcal{D},\mu,\lambda)] + \text{const} && (16.39)\\
&= \mathbb{E}_{q_\lambda}[\log p(\mathcal{D}|\mu,\lambda) + \log p(\mu|\lambda)] + \text{const} && (16.40)\\
&= \mathbb{E}_{q_\lambda}\left[-\frac{\lambda}{2}\left(\sum_{n=1}^N (x_n-\mu)^2 + \kappa_0(\mu-\mu_0)^2\right)\right] + \text{const} && (16.41)\\
&= -\frac{\mathbb{E}_{q_\lambda}[\lambda]}{2}\left(\sum_{n=1}^N (x_n-\mu)^2 + \kappa_0(\mu-\mu_0)^2\right) + \text{const} && (16.42)
\end{aligned}$$

$$\Rightarrow \quad q_\mu^*(\mu) \sim \mathcal{N}(\nu,\tau^{-1}), \quad \nu = \frac{\kappa_0\mu_0 + \sum_{n=1}^N x_n}{N+\kappa_0}, \quad \tau = (N+\kappa_0)\mathbb{E}_{q_\lambda}[\lambda]. \tag{16.43}$$

Further,

$$\begin{aligned}
\log q_\lambda^*(\lambda) &= \mathbb{E}_{q_\mu}[\log p(\mathcal{D},\mu,\lambda)] + \text{const} && (16.44)\\
&= \mathbb{E}_{q_\mu}[\log p(\mathcal{D}|\mu,\lambda) + \log p(\mu|\lambda) + \log p(\lambda)] + \text{const} && (16.45)\\
&= \mathbb{E}_{q_\mu}\Bigg[\frac{N}{2}\log\left(\frac{\lambda}{2\pi}\right) + \sum_{n=1}^N\left(-\frac{\lambda}{2}(x_n-\mu)^2\right) + \frac{1}{2}\log\left(\frac{\kappa_0\lambda}{2}\right) + \left(-\frac{\kappa_0\lambda}{2}(\mu-\mu_0)^2\right) && (16.46)\\
&\quad + (a_0-1)\log\lambda + (-b_0\lambda)\Bigg] + \text{const} && (16.47)\\
&= \mathbb{E}_{q_\mu}\left[\left(\frac{N+1}{2}+a_0-1\right)\log\lambda + \left(-\frac{1}{2}\sum_{n=1}^N (x_n-\mu)^2 - \frac{\kappa_0}{2}(\mu-\mu_0)^2 - b_0\right)\lambda\right] + \text{const} && (16.48)\\
&= \left(\frac{N+1}{2}+a_0-1\right)\log\lambda - \left(b_0 + \mathbb{E}_{q_\mu}\left[\frac{1}{2}\sum_{n=1}^N (x_n-\mu)^2 + \frac{\kappa_0}{2}(\mu-\mu_0)^2\right]\lambda\right) + \text{const} && (16.49)\\
& && (16.50)\\
\Rightarrow \quad & q_\lambda^*(\lambda) \sim \text{Gamma}(a,b), && (16.51)
\end{aligned}$$

where

$$a = \frac{N+1}{2} + a_0, \quad b = b_0 + \frac{1}{2}\mathbb{E}_{q_\mu}\left[\sum_{n=1}^N (x_n-\mu)^2 + \kappa_0(\mu-\mu_0)^2\right]. \tag{16.52}$$

As we can see from (16.43) and (16.51), q_μ is a Gaussian and q_λ is a Gamma. Therefore, in practice, we can initialize with these parametric forms and do updating on their parameters. Note that we did not specify the Gaussian and Gamma parametric forms beforehand.

The updating rules for parameters ν, τ, a, b are thus

$$\nu = \frac{\kappa_0 \mu_0 + \sum_{n=1}^N x_n}{N + \kappa_0}, \tag{16.53}$$

$$\tau = (N + \kappa_0)\mathbb{E}_{q_\lambda}[\lambda] = (N + \kappa_0)\frac{a}{b}, \tag{16.54}$$

$$a = \frac{N+1}{2} + a_0, \tag{16.55}$$

$$b = b_0 + \frac{1}{2}\mathbb{E}_{q_\mu}\left[\sum_{n=1}^N (x_n - \mu)^2 + \kappa_0(\mu - \mu_0)^2\right] \tag{16.56}$$

$$= b_0 + \frac{1}{2}\left(\left(\sum_{n=1}^N x_n^2\right) + \kappa_0\mu_0^2 - 2\left(\sum_{n=1}^N x_n + \kappa_0\mu_0\right)\mathbb{E}_{q_\mu}[\mu] + (N + \kappa_0)\mathbb{E}_{q_\mu}\left[\mu^2\right]\right) \tag{16.57}$$

$$= b_0 + \frac{1}{2}\left(\left(\sum_{n=1}^N x_n^2\right) + \kappa_0\mu_0^2 - 2\left(\sum_{n=1}^N x_n + \kappa_0\mu_0\right)\nu + (N + \kappa_0)\left(\nu^2 + \tau^{-1}\right)\right). \tag{16.58}$$

Figure 16.1 shows the updates when we apply CAVI to approximate the posterior of Gaussian parameters.

16.4 Factorized variational approximations are compact

The variational approximations $q(\boldsymbol{z})$ tend to be more compact than the actual posterior $p(\boldsymbol{z}|\boldsymbol{x})$. This is partly due to the natural asymmetry of KL-divergence. Consider that we approximate $p(x)$ using $q(x)$ by minimizing

$$D_{KL}(q(x)||p(x)) = \sum_x q(x) \log \frac{q(x)}{p(x)}. \tag{16.59}$$

It can be seen that when $p(x)$ is close to 0, $q(x)$ being large will contribute a large positive value to the KL. Therefore, to minimize $D_{KL}(q(x)||p(x))$, wherever $p(x)$ is small, $q(x)$ must also be small. $q(x)$ thus has a tendency of "shrinking" to only regions where $p(x)$ is not close to 0, shown in Figure 16.2a.

On the other hand, if we instead minimize

$$D_{KL}(p(x)||q(x)) = \sum_x p(x) \log \frac{p(x)}{q(x)}, \tag{16.60}$$

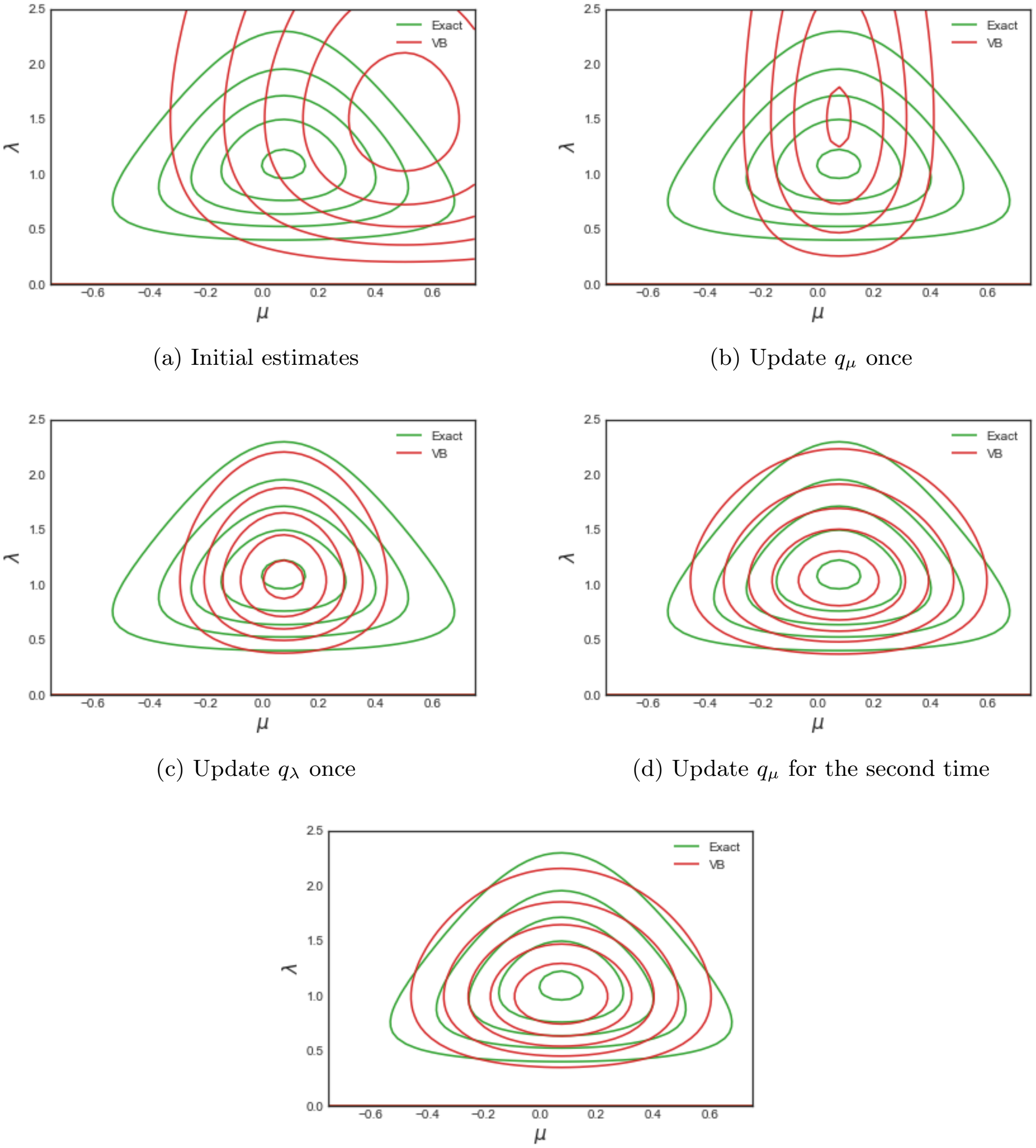
then wherever $p(x)$ is large, $q(x)$ must also be large. Therefore, $q(x)$ will have a tendency of "covering" regions where $p(x)$ is positive, shown in Figure 16.2b.

References

[1] David M Blei, Alp Kucukelbir, and Jon D McAuliffe. "Variational inference: A review for statisticians." In: *Journal of the American statistical Association* 112.518 (2017), pp. 859–877.

[2] Ian Goodfellow. "NIPS 2016 tutorial: Generative adversarial networks." In: *arXiv preprint arXiv:1701.00160* (2016).

[3] Kevin P Murphy. *Probabilistic machine learning: an introduction.* MIT press, 2022.



(a) Initial estimates

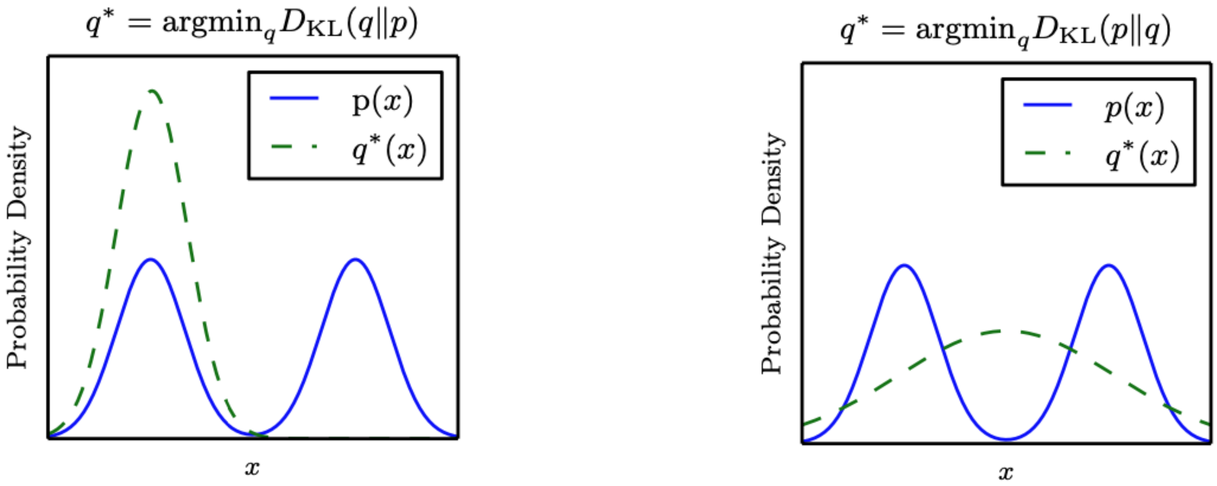
(b) Update q_μ once

(c) Update q_λ once

(d) Update q_μ for the second time

(e) Update q_λ for the second time

Figure 16.1: CAVI for the mean μ and precision λ of a univariate Gaussian distribution.



$q^* = \text{argmin}_q D_{\text{KL}}(q\|p)$

(a)

$q^* = \text{argmin}_q D_{\text{KL}}(p\|q)$

(b)

Figure 16.2: Approximating a bimodal distribution with a uni-modal distribution. Figures are from [2].