Notes On Variational Inference

Ludwig Winkler

December 31, 2017

Variational Bayesian Inference is a family of techniques for approximating intractable integrals arising in Bayesian inference and machine learning. They are typically used in complex statistical models consisting of observed variables as well as unknown parameters and latent variables, with various sorts of relationships among the three types of random variables.

1 PROBLEM SETTING

Let's assume that we have a set of observations x which we know where somehow produced by a process. We can call our set of observations $\mathbf{x}$ our data set.

But the process that generated the observations $\mathbf{x}$ also has a set of latent, random variables $\mathbf{z}$ which we cannot observe. An intuitive example would be a classification task where we have an observation $\mathbf{x}$ and a label $\mathbf{z}$. A standard classifier would construct a decision boundary from which we could obtain the probablity $p(\mathbf{z}|\mathbf{x})$. An analogy would be a neural network classifier with a final softmax layer which outputs a probability of a class, dependent on the input.

Variational Inference (VI) trains a generative model on the data which constructs a joint probability $p(\mathbf{x},\mathbf{z})$ from which we can conveniently infer much more information than just the class-conditional probability. By building the entire generative process $p(\mathbf{x},\mathbf{z})$ and not just the class-dependent probability $p(\mathbf{z}|\mathbf{x})$ we have the entire generative process at our disposal. Take for example studying for an exam: It is usually way better to actually

understand the entire topic intrinsically than just learning by heart a mapping from some question to some answer.

Furthermore we can assume that the true model $p(\mathbf{x}, \mathbf{z})$ has a set of parameters θ which parameterize the joint probability density function (PDF) and which we don't know. It is important to note that is helpful to assume that joint PDF $p(\mathbf{x}, \mathbf{z})$ consists of an arbitrary combination of parameterized PDFs. This combination of parameterized PDFs could be as simple as two normal distributions with independent parameters or a complicated combination of complex PDFs.

Variational Inference aims to find suitable parameters θ which explain the generation of the observations $\mathbf{x}$ with the latent $\mathbf{z}$. Using the analogy of images and labels, VI tries to find parameters which create great pictures for any label. Since we only receive the images, more generally $\mathbf{x}$, we want to model the generative process that produced them from labels, more generally $\mathbf{z}$, which we can't observe. Once we modeled the 'forward' process from $\mathbf{z}$ to $\mathbf{x}$ we can also go 'backward' and infer $\mathbf{z}$, the label, from $\mathbf{x}$, the image. This is possible because we are working with a generative model.

VI therefore aims to maximize the evidence of the observations $\mathbf{x}$ by finding the right parameter θ_{max}. If we were to find the right parameters θ_{max} we could generate valid observations $\mathbf{x}$ for any $\mathbf{z}$.

$$\theta_{\text{max}} = \text{argmax}_\theta \log p_\theta(\mathbf{x}) \tag{1.1}$$

Another nice thing to know, given some observations $\mathbf{x}$, for example images, would be to know the latent factors $\mathbf{z}$, for example the labels, that generated them. So we would like to know

$$p_{\theta_{\text{max}}}(\mathbf{z}|\mathbf{x}) = \frac{p_{\theta_{\text{max}}}(\mathbf{x}, \mathbf{z})}{\int p_{\theta_{\text{max}}}(\mathbf{x}, \mathbf{z}) d\mathbf{z}} = \frac{p_{\theta_{\text{max}}}(\mathbf{x}, \mathbf{z})}{p_{\theta_{\text{max}}}(\mathbf{x})} \tag{1.2}$$

The problem is that the denominator $\int p_{\theta_{\text{max}}}(\mathbf{x}, \mathbf{z}) d\mathbf{z}$ is usually an intractable integral. It might work for simpler models but for models which try to infer something on high-dimensional data, calculating the integral becomes quickly very difficult. Due to the usually intractable denominator it is difficult to correctly estimate $p_{\theta_{\text{max}}}(\mathbf{z}|\mathbf{x})$.

Because we condition the distribution $p_{\theta_{\text{max}}}(\mathbf{z}|\mathbf{x})$ on $\mathbf{x}$ we have to calculate the evidence $\int p_{\theta_{\text{max}}}(\mathbf{x}, \mathbf{z}) d\mathbf{z}$. In a nutshell, the trick of VI is to try to estimate an 'unconditional' distribution $q_\Psi(\mathbf{z})$ which doesn't require an intractable integral like $\int p_{\theta_{\text{max}}}(\mathbf{x}, \mathbf{z}) d\mathbf{z}$ does.

2 Jensen's Inequality & Kullback-Leibler Divergence

Jensen's Inequality is a useful equation for various proofs in probability and information theory. The inequality holds for convex or concave functions and a good intuition can be obtained from its geometric interpretation. As seen in Figure 2.1, Jensen's inequality relates the function values $f(x)$ between x_1 and x_2 to the linear combination of $f(x_1)$ and $f(x_2)$ through the inequality (2.1) for a concave function $f(x)$.

$$f(tx_1 + (1-t)x_2) \leq tf(x_1) + (1-t)f(x_2) \tag{2.1}$$

Jensen's inequality is stated in its general form in (2.2) for a concave function $\varphi(x)$.

$$\varphi(\mathbb{E}[X]) \leq \mathbb{E}\big[\varphi(X)\big] \tag{2.2}$$

For a convex function $\varphi(x)$ the inequality would be reversed as stated in (2.3).

$$\varphi(\mathbb{E}[X]) \geq \mathbb{E}\big[\varphi(X)\big] \tag{2.3}$$

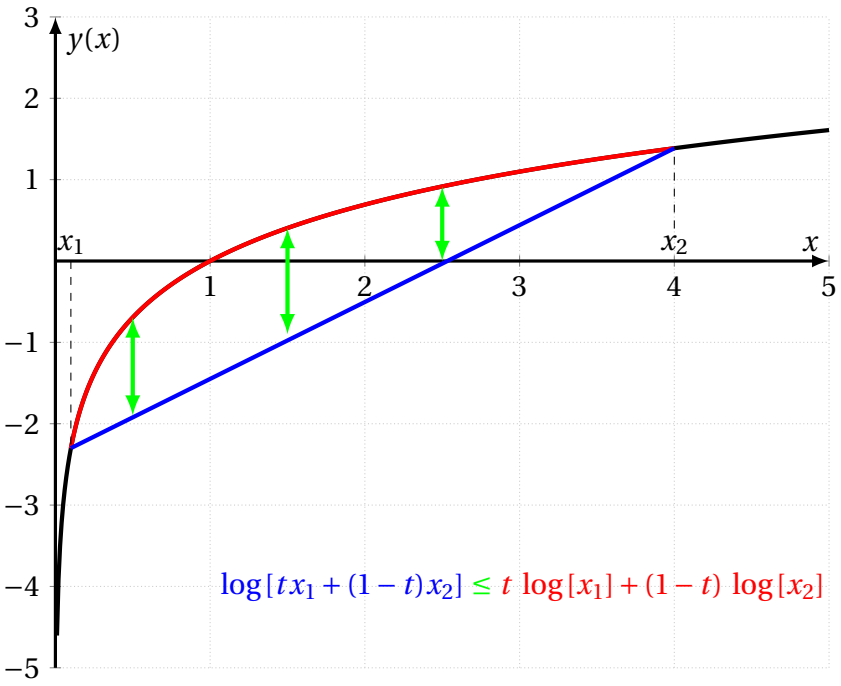

Figure 2.1: Visualization of Jensen's inequality for the concave function $f(x) = \log[x]$

The Kullback-Leibler divergence (KL divergence) is based on the information theoretic entropy. In order to obtain a good understanding of the KL divergence one should first consider what the intuition of the information theoretic entropy is.

In machine learning we usually work with probability distributions which assign probabilities $p(x) \in [0,1]$ to each possible event $x \in \mathcal{X}$ where $\mathcal{X}$ is the set of all possible events.

Furthermore the probabilities have to sum up to one, i.e. $\int_{\mathcal{X}} p(x)dx = 1$. The term 'usually' was used because most distributions are somewhat evenly distributed, i.e. they do not aggregate the probabilities in an infinitesimal small range like a Dirac impulse.

If the probability of an event x would be very large, e.g. $p(x) = 0.9$ then the occurences of such that event would be very frequent. The occurence of an event which has a high probability is not unexpected whereas the occurence of an event with a small probability would be unexpected.

Imagine a number generator which produces the number '0' with a probability $p(0) = 0.9$ and the number '1' with probability $p(1) = 0.1$. Sitting in front of the number generator the occurence of '0' would not be unexpected since we now it has a probability $p(0) = 0.9$ but the occurence of '1' would be something unexpected.

In Figure 2.1 the logarithm is plotted in black and red where the different colors were only used to aid at discerning the different components of Jensen's inequality. The negative logarithmic function $f(x) = -\log[x]$ conveniently transforms the probability of an event x into its 'unexpectedness'. The negative logarithm will result in a lot of 'unexpectedness' for small probabilities whereas large probabilities will result in just a little 'unexpectedness'. It is also noteworthy that the negative logarithm is positive for all values $[0,1]$ which is precisely the support for probabilities we 'usually' work with in probability density functions.

The information theoretic entropy computes the expectation over the 'unexpectedness' of all events $x \in \mathcal{X}$.

$$H(p(x)) = \mathbb{E}\big[-\log\big[p(x)\big]\big] = \sum_{x\in\mathcal{X}} p(x)\big(-\log\big[p(x)\big]\big) \tag{2.4}$$

The Kullback-Leibler divergence uses the entropy of one distribution $p(x)$ and the relative entropy of another distribution $q(x)$ to compute the divergence between them. The relative entropy measures the expected 'unexpectedness' of one distribution over another distribution. The KL divergence is only zero if the two distributions $p(x)$ and $q(x)$ are identical. Using the relative entropy reveals one drawback of the KL divergence, though: it is not a true distance metric since if we swapped $p(x)$ and $q(x)$ we would not obtain the same value, i.e. $\mathrm{KL}\big[q(x)||p(x)\big] \neq \mathrm{KL}\big[p(x)||q(x)\big]$.

The KL divergence is usually used to measure the difference of an approximate distribution $q(x)$ to a true distribution $p(x)$. It tends to concentrate its mass in areas where $p(x)$ has larger probabilities since areas where $p(x)$ has small values are weighted less due to the expectation over $p(x)$.

$$\mathrm{KL}\big[p(x)||q(x)\big] = \mathbb{E}_{p(x)}\big[-\log\big[q(x)\big]\big] - \underbrace{\mathbb{E}_{p(x)}\big[-\log\big[p(x)\big]\big]}_{\text{relative entropy}} \tag{2.5}$$

$$= \mathbb{E}_{p(x)}\big[-\log\big[q(x)\big] + \log\big[p(x)\big]\big] \tag{2.6}$$

$$= \mathbb{E}_{p(x)}\left[\log\left[\frac{p(x)}{q(x)}\right]\right] \tag{2.7}$$

Jensen's inequality can be used to show that the KL divergence is always greater equal zero.

$$\begin{aligned}
\mathrm{KL}\left[q(x)||p(x)\right] &= \mathbb{E}_{p(x)}\left[\log\left[\frac{p(x)}{q(x)}\right]\right] && (2.8)\\
&= \int_{\mathcal{X}} p(x)\,\log\left[\frac{p(x)}{q(x)}\right] dx && (2.9)\\
&= -\int_{\mathcal{X}} p(x)\,\log\left[\frac{q(x)}{p(x)}\right] dx && (2.10)\\
&\geq -\log\left[\int_{\mathcal{X}} p(x)\frac{q(x)}{p(x)}\right] dx && (2.11)\\
&= -\log\left[\int_{\mathcal{X}} p(x)dx\right] && (2.12)\\
&= -\log[1] && (2.13)\\
&= 0 && (2.14)
\end{aligned}$$

3 ELBO

The Kullback-Leibler-Divergence $\mathrm{KL}\left[q(\mathbf{x})||p(\mathbf{x})\right]$ measures the distance between two probability distributions $q(\mathbf{x})$ and $p(\mathbf{x})$. In variational inference it is used as a criterion with which we minimize the difference between $q_\Psi(\mathbf{z})$ and $p_\theta(\mathbf{z}|\mathbf{x})$ while fitting $q_\Psi(\mathbf{z})$ to the possible intractable posterior $p_\theta(\mathbf{z}|\mathbf{x})$.

$$\begin{aligned}
\min_\Psi \mathrm{KL}\left[q_\Psi(\mathbf{z})||p_\theta(\mathbf{z}|\mathbf{x})\right] &= \mathbb{E}_{q_\Psi(\mathbf{z})}\left[\log\frac{q_\Psi(\mathbf{z})}{p_\theta(\mathbf{z}|\mathbf{x})}\right] && (3.1)\\
&= \mathbb{E}_{q_\Psi(\mathbf{z})}\left[\log q_\Psi(\mathbf{z})\right] - \mathbb{E}_{q_\Psi(\mathbf{z})}\left[\log p_\theta(\mathbf{z}|\mathbf{x})\right] && (3.2)\\
&= \mathbb{E}_{q_\Psi(\mathbf{z})}\left[\log q_\Psi(\mathbf{z})\right] - \mathbb{E}_{q_\Psi(\mathbf{z})}\left[\log\frac{p_\theta(\mathbf{x},\mathbf{z})}{p_\theta(\mathbf{x})}\right] && (3.3)\\
&= \mathbb{E}_{q_\Psi(\mathbf{z})}\left[\log q_\Psi(\mathbf{z})\right] - \mathbb{E}_{q_\Psi(\mathbf{z})}\left[\log p_\theta(\mathbf{x},\mathbf{z})\right] + \mathbb{E}_{q_\Psi(\mathbf{z})}\left[\log p_\theta(\mathbf{x})\right] && (3.4)\\
&= \underbrace{\mathbb{E}_{q_\Psi(\mathbf{z})}\left[\log q_\Psi(\mathbf{z})\right] - \mathbb{E}_{q_\Psi(\mathbf{z})}\left[\log p_\theta(\mathbf{x},\mathbf{z})\right]}_{-\mathrm{ELBO}\left[q_\Psi(\mathbf{z})\right]} + \log p_\theta(\mathbf{x}) && (3.5)\\
&= \mathbb{E}_{q_\Psi(\mathbf{z})}\left[\log\frac{q_\Psi(\mathbf{z})}{p_\theta(\mathbf{x},\mathbf{z})}\right] + \log p_\theta(\mathbf{x}) && (3.6)\\
&= \mathrm{KL}\left[q_\Psi(\mathbf{z})||p_\theta(\mathbf{x},\mathbf{z})\right] + \log p_\theta(\mathbf{x}) && (3.7)
\end{aligned}$$

We can intuitively see from 3.5 and 3.7 that we can reformulate the minimization problem as a minimization of the KL-Divergence between the variational distribution $q_\Psi(\mathbf{z})$ and $p_\theta(\mathbf{x},\mathbf{z})$. The term $\log p_\theta(\mathbf{x})$ is a constant with respect to our variational distribution $q_\Psi(\mathbf{z})$.

The name 'Evidence Lower Bound' (ELBO) is derived from a property of the Kullback-Leibler-Divergence, namely that $\mathrm{KL}\left[(||q\right](\mathbf{x}) \,||\, p(\mathbf{x})) \geq 0$ for any $q(\mathbf{x})$ and $p(\mathbf{x})$. For variational inference this implicates the following identity:

$$\begin{aligned}
\mathrm{KL}\left[q_\Psi(\mathbf{z})||p_\theta(\mathbf{z}|\mathbf{x})\right] &\geq 0 && (3.8)\\
\mathrm{KL}\left[q_\Psi(\mathbf{z})||p_\theta(\mathbf{x},\mathbf{z})\right] + \log p_\theta(\mathbf{x}) &\geq 0 && (3.9)\\
-\mathrm{ELBO} + \log p_\theta(\mathbf{x}) &\geq 0 && (3.10)\\
\log p_\theta(\mathbf{x}) &\geq \mathrm{ELBO} && (3.11)
\end{aligned}$$

In order to minimze our original objective in 3.1 we have to maximize the ELBO in 3.5. With the property of the KL-Divergence, namely $\mathrm{KL}\left[||() \, q(\mathbf{x}) \,||\, p(\mathbf{x})\right) \geq 0$, we can see that the ELBO is bounded from above by the log-probability of $p_\theta(\mathbf{x})$. So we can only maximize the ELBO up to the log-probability of $p_\theta(\mathbf{x})$.

In order to minimize our original objective in 3.1 we will try to approximate the joint distribution $p_\theta(\mathbf{x},\mathbf{z})$ with a simpler distribution $q_\Psi(\mathbf{z})$. The KL-Divergence is always ≥ 0 so we will be always left with the constant $\log p_\theta(\mathbf{x})$ which we will not be able to optimize. In a nutshell by approximating the complicated conditional distribution $p_\theta(\mathbf{z}|\mathbf{x})$ with a simpler $q_\Psi(\mathbf{z})$ we don't have to deal with the intractable integral but we also obtain a term in our optimization problem which we cannot reduce.

To gain more intuition about the ELBO we can go two steps back and look at the following:

$$\begin{aligned}
\text{ELBO}\left(q_\Psi(\mathbf{z})\right) &= \mathbb{E}_{q_\Psi(\mathbf{z})}\left[\log p_\theta(\mathbf{x},\mathbf{z})\right] - \mathbb{E}_{q_\Psi(\mathbf{z})}\left[\log q_\Psi(\mathbf{z})\right] && (3.12)\\
&= \mathbb{E}_{q_\Psi(\mathbf{z})}\left[\log \frac{p_\theta(\mathbf{x},\mathbf{z})\, p_\theta(\mathbf{z})}{p_\theta(\mathbf{z})}\right] - \mathbb{E}_{q_\Psi(\mathbf{z})}\left[\log q_\Psi(\mathbf{z})\right] && (3.13)\\
&= \mathbb{E}_{q_\Psi(\mathbf{z})}\left[\log p_\theta(\mathbf{x}|\mathbf{z})\right] + \mathbb{E}_{q_\Psi(\mathbf{z})}\left[p_\theta(\mathbf{z})\right] - \mathbb{E}_{q_\Psi(\mathbf{z})}\left[\log q_\Psi(\mathbf{z})\right] && (3.14)\\
&= \mathbb{E}_{q_\Psi(\mathbf{z})}\left[\log p_\theta(\mathbf{x}|\mathbf{z})\right] - \text{KL}\left[q_\Psi(\mathbf{z})||p_\theta(\mathbf{z})\right] && (3.15)
\end{aligned}$$

Keeping in mind that we have to maximize the ELBO term in 3.5 can furthermore see in 3.15 that $q_\Psi(\mathbf{z})$ will be balanced between putting weight to the likelihood as well as the prior. The expected likelihood emphasizes $q_\Psi(\mathbf{z})$ putting its probability on configurations of $\mathbf{z}$ that explain the observed data $\mathbf{x}$. If too much emphasis is paid to the expected likelihood the KL-Divergence between the prior $p_\theta(\mathbf{z})$ and $q_\Psi(\mathbf{z})$ will grow. The ELBO term therefore is encouraged to find a balanced approximation of the prior $p_\theta(\mathbf{z})$ and the likelihood $p_\theta(\mathbf{x}|\mathbf{z})$. This is corroborated by the fact that we try to approximate the entire joint probability $p_\theta(\mathbf{x},\mathbf{z})$ with $q_\Psi(\mathbf{z})$.

4 MEAN-FIELD VARIATIONAL INFERENCE

Previously we derived the optimization procedure for a variational distribution $q_\Psi(\mathbf{z})$ where $q_\Psi(\mathbf{z})$ was a joint distribution over the latent variables $\mathbf{z}$. Modelling the variational distribution as an interdependent distribution allows the model to capture rich settings but increases the computational load during optimization.

A simple yet powerful way of improving the optimizatin is to assume statistical independence between the latent variables z_i. So instead of modelling the distribution $q_\Psi(\mathbf{z})$ over all latent variables $\mathbf{z}$ jointly we factorize the distribution into statistically independent marginal distributions $q_\Psi(z_i)$ over each of the latent variables z_i [1].

$$q_\Psi(\mathbf{z}) = q_\Psi(z_1, \ldots, z_N) = \prod_{i=0}^{N} q_\Psi(z_i) \tag{4.1}$$

The use of factorized variational distributions is called mean-field variational inference. The use of the term 'mean field' originates from physics and probability theory where complex systems are composed of a large number of simpler components. To understand how mean field methods are used in variational inference we will first simplify the Kullback-Leibler divergence as before.

$$\min_\Psi \mathrm{KL}\left[q_\Psi(\mathbf{z}) || p_\theta(\mathbf{z}|\mathbf{x})\right] = \mathbb{E}_{q_\Psi(\mathbf{z})}\left[\log\left[\frac{q_\Psi(\mathbf{z})}{p_\theta(\mathbf{x},\mathbf{z})}\right]\right] \tag{4.2}$$

$$= \int q_\Psi(\mathbf{z}) \log\left[\frac{q_\Psi(\mathbf{z})}{p_\theta(\mathbf{x},\mathbf{z})}\right] d\mathbf{z} \tag{4.3}$$

$$= \int q_\Psi(\mathbf{z}) \log\left[q_\Psi(\mathbf{z})\right] d\mathbf{z} - \int q_\Psi(\mathbf{z}) \log\left[p_\theta(\mathbf{x},\mathbf{z})\right] d\mathbf{z} \tag{4.4}$$

Now we can use the fact that we set up our variational distribution as a mean field distribution. It should be paid special attention to the simplifications which the statistical independence of the marginal distributions allow.

$$\int \prod_i^N q_\Psi(z_i) \log\left[\prod_j^N q_\Psi(z_j)\right] dz_i \ldots dz_N - \int \prod_i^N q_\Psi(z_i) \log\left[p_\theta(\mathbf{x},\mathbf{z})\right] dz_1 \ldots dz_N \tag{4.5}$$

$$= \int \prod_i^N q_\Psi(z_i) \sum_j \log\left[q_\Psi(z_j)\right] dz_i \ldots dz_N - \int \prod_i^N q_\Psi(z_i) \log\left[p_\theta(\mathbf{x},\mathbf{z})\right] dz_1 \ldots dz_N \tag{4.6}$$

$$= \sum_j^N \int \prod_i^N q_\Psi(z_i) \log\left[q_\Psi(z_j)\right] dz_i \ldots dz_N - \int \prod_i^N q_\Psi(z_i) \log\left[p_\theta(\mathbf{x},\mathbf{z})\right] dz_1 \ldots dz_N \tag{4.7}$$

$$= \sum_j^N \int q_\Psi(z_j) \log\left[q_\Psi(z_j)\right] dz_j - \int \prod_i^N q_\Psi(z_i) \log\left[p_\theta(\mathbf{x},\mathbf{z})\right] dz_1 \ldots dz_N \tag{4.8}$$

It took me personally some time to understand and verify the algebra in the first term of (4.7). Yet it can be easily understood when one leverages the statistical independence in the

variational distribution. To illustrate the algebra lets have a look at a simple example of a mean field variational distribution over three latent variables.

$$\sum_j^3 \int\int\int \prod_i^3 q_\Psi(z_i)\ \log\left[q_\Psi(z_j)\right] dz_i \dots dz_N \tag{4.9}$$

$$= \sum_j^3 \int\int\int q_\Psi(z_1) q_\Psi(z_2) q_\Psi(z_3)\ \log\left[q_\Psi(z_j)\right] dz_1 dz_2 dz_3 \tag{4.10}$$

$$\begin{aligned} &= \int\int\int q_\Psi(z_1) q_\Psi(z_2) q_\Psi(z_3)\ \log\left[q_\Psi(z_1)\right] dz_1 dz_2 dz_3 \\ &\quad + \int\int\int q_\Psi(z_1) q_\Psi(z_2) q_\Psi(z_3)\ \log\left[q_\Psi(z_2)\right] dz_1 dz_2 dz_3 \\ &\quad\quad + \int\int\int q_\Psi(z_1) q_\Psi(z_2) q_\Psi(z_3)\ \log\left[q_\Psi(z_3)\right] dz_1 dz_2 dz_3 \end{aligned} \tag{4.11}$$

$$\begin{aligned} &= \int q_\Psi(z_1) \underbrace{\int q_\Psi(z_2) dz_2}_{=1} \underbrace{\int q_\Psi(z_3) dz_3}_{=1} \log\left[q_\Psi(z_1)\right] dz_1 \\ &\quad + \int \underbrace{\int q_\Psi(z_1) dz_1}_{=1} q_\Psi(z_2) \underbrace{\int q_\Psi(z_3) dz_3}_{=1} \log\left[q_\Psi(z_2)\right] dz_2 \\ &\quad\quad + \int \underbrace{\int q_\Psi(z_1) dz_1}_{=1} \underbrace{\int q_\Psi(z_2) dz_2}_{=1} q_\Psi(z_3)\ \log\left[q_\Psi(z_3)\right] dz_3 \end{aligned} \tag{4.12}$$

$$\begin{aligned} &= \int q_\Psi(z_1)\ \log\left[q_\Psi(z_1)\right] dz_1 + \int q_\Psi(z_2)\ \log\left[q_\Psi(z_2)\right] dz_2 \\ &\quad + \int q_\Psi(z_3)\ \log\left[q_\Psi(z_3)\right] dz_3 \end{aligned} \tag{4.13}$$

$$= \sum_j^3 \int q_\Psi(z_j)\ \log\left[q_\Psi(z_j)\right] dz_j \tag{4.14}$$

We now have two terms in our objective function: a sum of entropies over the marginals of the variational distribution in the first term and a relative entropy of the joint model distribution over the latent and observable variables with respect to the full variational distribution in the second term. Again we can leverage the statistical independence of the variational distribution to arrive at 'Coordinate Ascent Variational Inference' (CAVI) [2].

This algorithm is based on the independent optimization of each variational distribution $q_\Psi(z_j)$ such that we optimize with respect to all other variational distributions. We therefore pick one variational distribution $q_\Psi(z_i)$ in the sum in the first term and seperate it from all the other variational distributions into an expectation in the second term. When we recombine the two terms back into a Kullback-Leibler divergence we can see that the optimum of the objective function with respect to a single variational distribution is reached when the nominator an denominator are equal. In that case the fraction will be one and with the logarithm the objective function will be zero. We can therefore optimize each variational

distribution independently from all other. It should be noted that a normalization constant is required to obtain a probability distribution. While we can simply normalize the distribution into a probability distribution once we acquired it, the normalization term is contained in the terms which we discarded to obtain a feasible optimization problem.

$$\int q_\Psi(z_j)\ \log\left[q_\Psi(z_j)\right] dz_j - \int q_\Psi(z_j) \prod_{i\neq j}^{N} q_\Psi(z_i)\ \log\left[p_\theta(\mathbf{x},\mathbf{z})\right] dz_1 \dots dz_N \tag{4.15}$$

$$= \int q_\Psi(z_j)\ \log\left[q_\Psi(z_j)\right] dz_j - \int q_\Psi(z_j) \mathbb{E}_{q_\Psi(z_{-j})}\left[\ \log\left[p_\theta(\mathbf{x},\mathbf{z})\right]\right] dz_j \tag{4.16}$$

$$= \int q_\Psi(z_j)\ \log\left[q_\Psi(z_j)\right] dz_j - \int q_\Psi(z_j)\ \log\left[\exp\left[\mathbb{E}_{q_\Psi(z_{-j})}\left[\ \log\left[p_\theta(\mathbf{x},\mathbf{z})\right]\right]\right]\right] dz_j \tag{4.17}$$

$$= \int q_\Psi(z_j)\ \log\left[\frac{q_\Psi(z_j)}{\exp\left[\mathbb{E}_{q_\Psi(z_{-j})}\left[\ \log\left[p_\theta(\mathbf{x},\mathbf{z})\right]\right]\right]}\right] \tag{4.18}$$

5 Stochastic Gradient Optimization

Similarly to how neural networks are nowadays trained with stochastic gradient descent we can train variational inference algorithms with stochastic gradient otpimization [3]. Recall from 3.5 that in order to minimize our original objective we need to maximize the Evidence Lower Bound.

$$\nabla_\Psi \left[-\text{ELBO}\left(q_\Psi(\mathbf{z})\right)\right] \tag{5.1}$$
$$= \nabla_\Psi \left[\text{KL}\left[q_\Psi(\mathbf{z}) || p_\theta(\mathbf{x},\mathbf{z})\right]\right] \tag{5.2}$$
$$= \nabla_\Psi \left[\int q_\Psi(\mathbf{z}) \log \frac{q_\Psi(\mathbf{z})}{p_\theta(\mathbf{x},\mathbf{z})} d\mathbf{z}\right] \tag{5.3}$$
$$= \int \nabla_\Psi \left[q_\Psi(\mathbf{z})\right] \log\left[\frac{q_\Psi(\mathbf{z})}{p_\theta}(\mathbf{x},\mathbf{z})\right] d\mathbf{z} + \int q_\Psi(\mathbf{z}) \nabla_\Psi \left[\log\left[\frac{q_\Psi(\mathbf{z})}{p_\theta(\mathbf{x},\mathbf{z})}\right]\right] d\mathbf{z} \tag{5.4}$$
$$= \int \nabla_\Psi \left[q_\Psi(\mathbf{z})\right] \log\left[\frac{q_\Psi(\mathbf{z})}{p_\theta(\mathbf{x},\mathbf{z})}\right] d\mathbf{z} + \int q_\Psi(\mathbf{z}) \left(\nabla_\Psi \left[\log\left[q_\Psi(\mathbf{z})\right]\right] - \underbrace{\nabla_\Psi \left[p_\theta(\mathbf{x},\mathbf{z})\right]}_{=0}\right) d\mathbf{z} \tag{5.5}$$
$$= \int \nabla_\Psi \left[q_\Psi(\mathbf{z})\right] \log\left[\frac{q_\Psi(\mathbf{z})}{p_\theta(\mathbf{x},\mathbf{z})}\right] d\mathbf{z} + \underbrace{\int q_\Psi(\mathbf{z})\, \nabla_\Psi \left[\log\left[q_\Psi(\mathbf{z})\right]\right] d\mathbf{z}}_{=0} \tag{5.6}$$
$$= \int q_\Psi(\mathbf{z})\, \nabla_\Psi \left[\log\left[q_\Psi(\mathbf{z})\right]\right] \log\left[\frac{q_\Psi(\mathbf{z})}{p_\theta(\mathbf{x},\mathbf{z})}\right] d\mathbf{z} \tag{5.7}$$
$$\approx \frac{1}{N} \sum_{\mathbf{z}_n}^{N} \nabla_\Psi \left[q_\Psi(\mathbf{z}_n)\right] \left(\log\left[\frac{q_\Psi(\mathbf{z}_n)}{p_\theta(\mathbf{x},\mathbf{z}_n)}\right] + K\right) \tag{5.8}$$

This gives us a training algorithm in which we don't have to use the entire set, but with which we can train batch by batch just like stochastic gradient descent in neural networks.

The identity $\nabla \log p(x) = \frac{\nabla p(x)}{p(x)}$ was used for

$$\int_{\mathbf{z}} q_\Psi(\mathbf{z}) \nabla_\Psi \left[\log q_\Psi(\mathbf{z})\right] = \int_{\mathbf{z}} \nabla_\Psi [q_\Psi(\mathbf{z})] \tag{5.9}$$
$$= \nabla_\Psi \left[\int_{\mathbf{z}} q_\Psi(\mathbf{z})\right] \tag{5.10}$$
$$= \nabla_\Psi [1] \tag{5.11}$$
$$= 0 \tag{5.12}$$

6 Example

This is an example which is taken from the tutorials on the probabilistic programming framework 'Pyro' by Uber AI Labs. In this example we want to infer the probability of a coin to land on its head (1) or tail (0). For that we have a data set of observations which consists of 60% heads and 40% tails.

It is straight forward to see from the distributin of observed heads and tails that the coin is biased towards heads but for more complex problems and observations the inference might not be as obvious.

In order to make variational inference work properly we need to ingredients: data and a model that structurally captures the generative process of the data. Once the data and the model are defined, variational inference will optimize the generative model to find the parameters of the model which best explain the data. Once the approximate solution for the latent parameters is found we can calculate the mean and variance of our latent factors.

The model for the coin problem is shown in Figure 6.1. The values of α and β are parameters for a Beta distribution which has the convenient property of having support of $x \in [0,1]$. This means that when we sample from a Beta distribution we will always obtain a value between zero and one, which we can use as a probability. The Beta distribution is often used for priors since it encodes the belief how likely certain values for the latent random variables are.

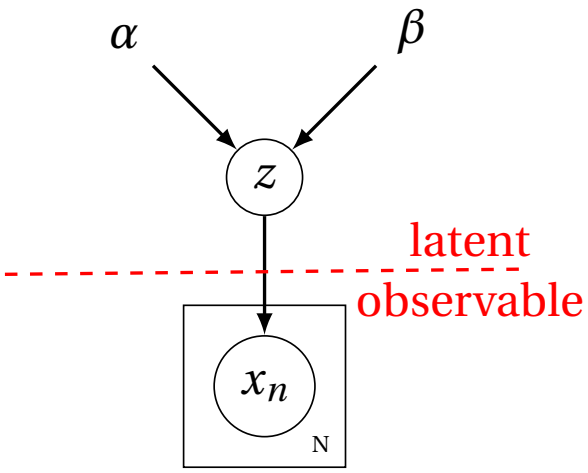

Figure 6.1: The probability z of the coin to give heads is given by the Beta distribution $z \sim p_{Beta}(\cdot|\alpha,\beta)$. Once the z is drawn, a number of N samples x_n are drawn according to a Bernoulli distribution, $x_n \sim p_{Bernoulli}(\cdot \mid z)$. Variational inference aims to find the parameters α and β.

The Beta distribution is in turn used as input for the Bernoulli distribution which tells us how probable each of the two observations ('head' or 'tail') is for the latent random variable x. If the prior distribution is very concentrated around the value 0.9 for example, the likelihood distribution will sample heads with a probability of 90%. Due to our very concentrated prior, the samples will have a high probability of being heads.

$$p(x, z; \alpha, \beta) = p_{\text{Bernoulli}}(x \mid z)\ p_{\text{Beta}}(z; \alpha, \beta) \tag{6.1}$$

Equation 6.1 formulates the generative model into mathematical form. The probability $p(x, z; \alpha, \beta)$ is the joint probability of the random variable x which we can observe and the latent random variable z. The parameters α and β define the prior probability $p(z; \alpha, \beta)$. The prior probability in turn defines the likelihood of the observations.

With the generative process defined, variational inference tries to find a minimum of the Kullback-Leibler divergence between the joint distribution $p(x, z; \alpha, \beta)$ and a 'variational' distribution $q_\Psi(\mathbf{z})$.

REFERENCES

[1] D. M. Blei, A. Kucukelbir, and J. D. McAuliffe, "Variational inference: A review for statisticians," *Journal of the American Statistical Association*, no. just-accepted, 2017.

[2] C. M. Bishop, *Pattern recognition and machine learning*. springer, 2006.

[3] D. Wingate and T. Weber, "Automated variational inference in probabilistic programming," *arXiv preprint arXiv:1301.1299*, 2013.