Variational Bayesian inference

1 Variational inference

Consider the inference problem where we want to approximate some true posterior distribution $p^*(X) = p(X|\mathcal{D})$, $\mathcal{D}$ being the dataset. The basic idea of variational inference is to pick an approximation $q(X)$ to the true distribution from a suited family of distributions, and to make this approximation as close as possible from the true posterior. This reduces **inference** to an **optimization** problem. Variational inference often gives us the speed benefits of MAP estimation but the statistical benefits of the Bayesian approach.

2 Mathematical framework and notations

2.1 Hypotheses

A sample X lives in the input space $\mathcal{X}$. The objective of a generative model is to sample from $P(X)$. Variational bayesian models have latent variables z living in a latent space $\mathcal{Z}$, with $P(z)$ a distribution we can easily sample from. We denote by Θ the parameter space. We use the law of total probability to have an expression of $p(X)$:

$$p(X) = \int_{\mathcal{Z}} \underbrace{p(X|z)}_{\text{likelihood}} \underbrace{p(z)}_{\text{prior}} \,\mathrm{d}z \ . \tag{1}$$

2.2 Mean-field variational Bayes

Consider a parametrized likelihood function q_ϕ. Mean-field variational Bayes uses the Reverse Kullback-Leibler divergence as a metric between two distributions:

$$\mathbb{KL}(q_\phi(z|X) \parallel p(z|X)) = \sum_{z \in \mathcal{Z}} q_\phi(z|X) \log \frac{q_\phi(z|X)}{p(z|X)} \ . \tag{2}$$

Although it does not define a distance in some distribution space, it is still a quantity we would like to minimize. It is the amount of information required to “distort” $p(z|X)$ into $q_\phi(z|X)$, which ideally we would like to be small. The following form makes it more tractable for an optimization algorithm:

$$\begin{aligned}
\mathbb{KL}(q_\phi(z|X) \parallel p(z|X)) &= \sum_{z\in\mathcal{Z}} q_\phi(z|X) \log \frac{q_\phi(z|X)p(X)}{p(z,X)} && (3)\\
&= \sum_{z\in\mathcal{Z}} q_\phi(z|X) \log \frac{q_\phi(z|X)}{p(z,X)} + \sum_{z\in\mathcal{Z}} q_\phi(z|X) \log p(X) && (4)\\
&= \log p(X) + \sum_{z\in\mathcal{Z}} q_\phi(z|X) \log \frac{q_\phi(z|X)}{p(z,X)} \quad . && (5)
\end{aligned}$$

Therefore,

$$\arg\min_\phi \mathbb{KL}(q_\phi(z|X) \parallel p(z|X)) = \arg\min_\phi \sum_{z\in\mathcal{Z}} q_\phi(z|X) \log \frac{q_\phi(z|X)}{p(z,X)} \quad . \tag{6}$$

The divergence is still not in a tractable form since we need to evaluate $p(z,X)$.

$$\begin{aligned}
\sum_{z\in\mathcal{Z}} q_\phi(z|X) \log \frac{q_\phi(z|X)}{p(z,X)} &= \mathbb{E}_{z\sim q_\phi(z|X)}\left[\log \frac{q_\phi(z|X)}{p(z,X)}\right] && (7)\\
&= \mathbb{E}_{q_\phi}\left[\log q_\phi(z|X) - \log p(X,z)\right] && (8)\\
&= \mathbb{E}_{q_\phi}\left[\log q_\phi(z|X) - \log p(X|z) - \log p(z)\right] && (9)
\end{aligned}$$

While computer software typically **minimizes** an objective function, we will write the equivalent maximization problem:

$$\begin{aligned}
\max_\phi \mathcal{L}(\phi) &= -\mathbb{E}_{q_\phi}\left[\log q_\phi(z|X) - \log p(X|z) - \log p(z)\right] && (10)\\
&= \mathbb{E}_{q_\phi}\left[\log p(X|z) + \log \frac{p(z)}{q_\phi(z|X)}\right] && (11)\\
&= \mathbb{E}_{q_\phi}\left[\log p(X|z)\right] + \sum_{z\in\mathcal{Z}} q_\phi(z|X) \log \frac{p(z)}{q_\phi(z|X)} && (12)\\
&= \mathbb{E}_{q_\phi}\left[\log p(X|z)\right] - \mathbb{KL}(q_\phi(z|X) \parallel p(z)) \quad . && (13)
\end{aligned}$$

Plugging back $\mathcal{L}$ into Eq. (2), we get

$$\begin{aligned}
\mathbb{KL}(q_\phi(z|X) \parallel p(z|X)) &= \log p(X) - \mathcal{L}\\
\log p(X) &= \mathcal{L} + \mathbb{KL}(q_\phi(z|X) \parallel p(z|X)) \quad .
\end{aligned} \tag{14}$$

Since $\mathbb{KL}(q_\phi(z|X) \parallel p(z|X)) \geq 0$, $\log p(X) \geq \mathcal{L}$, making $\mathcal{L}$ a lower bound for $\log p(X)$. This bound is called the **variational lower bound** (or **evidence lower bound**, ELBO).

2.3 Forward KL vs. Reverse KL

As stated previously, the Kullback-Leibler (KL) divergence is not symmetric, hence does not define a distance on metric spaces. Minimizing the forward or the reverse KL will lead to different behavior. Mean-field variational Bayes uses reverse KL instead of forward KL. Why? The forward KL divergence reads

$$\mathbb{KL}(p(z|X) \parallel q_\phi(z|X)) = \sum_{z \in \mathcal{Z}} p(z|X) \log \frac{p(z|X)}{q_\phi(z|X)} \ . \tag{15}$$

First, Eq. (15) cannot be evaluated since we need to compute $p(z|X)$, which we don't know yet. There is an even more fundamental reason to explain we prefer the reverse divergence to the forward.

As argued by [1], Eq. (2) is infinite if $p(z|X) = 0$ and $q_\phi(z|X) > 0$. Thus, to be minimized, Eq. (2) will ensure that when $p(z|X) = 0$, $q_\phi(z|X) = 0$. The reverse KL is **zero forcing** for q_ϕ. Conversely, Eq. (15) is infinite if $q_\phi(z|X) = 0$ and $p(z|X) > 0$. To be minimized, Eq. (15) will ensure that when $q_\phi(z|X) = 0$, $p(z|X) > 0$. The forward KL is **zero avoiding** for q_ϕ. Hence q_ϕ will under-estimate the support of p.

When the true distribution is multimodal (which is almost always the case in high-dimensional case), using the forward KL is a bad idea, since the resulting posterior mode/mean will be in a region of low density, right between the two peaks. In such contexts, the reverse KL is more sensible statistically.

3 Variational auto-encoders

Variational autoencoders (VAE) are generative latent variable models. In statistics, latent variables are hidden variables in the sense that they are not directly observed but are rather inferred through a mathematical model from other variables that are observed. The goal of a generative model is to construct a model that samples from a (high-dimensional) distribution P. Generative models usually face 3 main issues:

1. Strong assumptions have to be made on the structure of the data.
2. Models use severe approximations which lead to suboptimal results.
3. They have computationally expensive inference procedures (e.g. MCMC).

To circumvent these issues, powerful function approximators can be trained via backpropagation with a lot of samples. These function approximators are typically neural networks, a class of statistical models proved to be universal approximators under mild assumptions.

In VAEs, we assume that samples produced by the model are nearly like $z \in \mathcal{Z}$, that is

$$P(X|z,\theta) = \mathcal{N}(X|f(z;\theta), \sigma^2 I) \ \ , \tag{16}$$

with σ some hypothesized noise level.

3.1 A surrogate objective

VAE are trained by optimizing $\theta \in \Theta$ via gradient descent to increase $P(X)$ by making $f(z;\theta)$ approach X for some $z \in \mathcal{Z}$. Intuitively speaking, we are making the training data more likely under the generative model. This hints at some crucial properties for $f(z;\theta)$ to be optimized: it needs to be computable, continuous and differentiable in $\theta \in \Theta$. To solve Eq. (1), there remains 2 problems:

1. What is $z \in \mathcal{Z}$? How is it defined? What does it represent?
2. How to evaluate the integral over $\mathcal{Z}$?

3.1.1 Defining latent variables

A lot of information might and need to be encoded in $z \in \mathcal{Z}$, in order to generate a sample. VAE assume that $z \sim \mathcal{N}(0, I)$. First, since we do not have a general prior knowledge of how z is distributed we prefer to use a prior maximizing entropy. Second, by inverse transform sampling, any d-dimensional distribution can be generated by taking a set of d variables that are normally distributed and mapping them through a sufficiently complicated function. In the case of VAE, these functions are typically learnt by gradient descent and approximated by a neural network. For multi-layer neural networks, one can intuitively think that the first layers are used to map the normally distributed z's to latent values, which can then be mapped to the sampling space. In other words, the neural network learns the latent structure on its own. Therefore,

$$P(X) = \int_{\mathcal{Z}} P(X|z;\theta)\mathcal{N}(z|0,I)\,\mathrm{d}z \ \ . \tag{17}$$

3.2 Sampling from $\mathcal{Z}$

A naive way to estimate $P(X)$ is to use a Monte-Carlo sampling method, sampling from $P(z)$ and then applying $P(X) \approx \sum_z P(X|z;\theta)$. In practice, this is intractable as we need exponentially more samples as the dimension grows, an instance of the curse of dimensionality. Intuitively, one might realize that for most $z \in \mathcal{Z}, P(X|z) \approx 0$. We cannot use a stronger prior on z, since it is assumed to be Gaussian. We need to find a way to select z for which $P(X|z)$ is high and compute $P(X)$ just from those. The solution is to use a new function $Q(z|X)$ which can take a value $X \in \mathcal{X}$ and outputs a distribution over $\mathcal{Z}$ likely to produce $X \in \mathcal{X}$. Hopefully, the space of z values likely under Q is much smaller than the space of z values likely under the prior $P(z)$.

Now, that we have defined Q, we need to find some mathematical relationship between $P(X)$ and $\mathbb{E}_{z\sim Q}[P(X|z)]$. Remember that for $\theta \in \Theta$ and $z \in \mathcal{Z}$ fixed, $P(X|z)$ is considered a random variable. Starting from Kullback-Leibler divergence between $P(X|z)$ and $Q(z)$,

$$\mathbb{KL}(Q(z)||P(z|X)) = \mathbb{E}_{z\sim Q}[\log Q(z) - \log P(z|X)] \ . \tag{18}$$

Using Bayes' rule,

$$\mathbb{KL}(Q(z)||P(z|X)) = \mathbb{E}_{z\sim Q}[\log Q(z) - \log P(X|z) - \log P(z)] + \log P(X) \tag{19}$$

$$\log P(X) - \mathbb{KL}(Q(z)||P(z|X)) = \mathbb{E}_{z\sim Q}[\log P(X|z) - \mathbb{KL}(Q(z)||P(z))] \ . \tag{20}$$

Since we are in making Q dependent on X, we re-write the equation as

$$\underbrace{\log P(X) - \mathbb{KL}(Q(z|X)||P(z|X))}_{\text{a quantity we'd like to maximize}} = \underbrace{\mathbb{E}_{z\sim Q}[\log P(X|z)] - \mathbb{KL}(Q(z|X)||P(z))}_{\text{a quantity optimizable via SGD}} \ . \tag{21}$$

1. The divergence $\mathbb{KL}(Q(z|X)||P(z|X))$ is an error term that makes Q produce z that can reproduce a given X. It forces $Q(z|X)$ to be close to $P(z|X)$. Assuming an arbitrarily high-capacity model, $Q(z|X)$ hopefully matches $P(z|X)$. $P(z|X)$ describes the values of $z \in \mathcal{Z}$ that are likely to give rise to a sample like X in our model.
2. The term $\mathbb{E}_{z\sim Q}[\log P(X|z)]$ is an encoder/decoder term: z is produced from X via Q and we want to maximize the probability of occurrence of some X given the sampled latent variable $z \in \mathcal{Z}$.

Now, we shall give a form to $Q(z|X)$. The usual choice for a VAE is

$$Q(z|X) = \mathcal{N}(z|\mu(X;\vartheta), \Sigma(X;\vartheta)) \ . \tag{22}$$

The choice of a Normal distribution is motivated by the fact that $\mathbb{KL}(Q(z|X)||P(z))$ has a closed-form solution.

$$\mathbb{KL}(Q(z|X)||P(z)) = \frac{1}{2}\left(\mathrm{tr}(\Sigma(X)) + \mu(X)^\top \mu(X) - k - \log\det(\Sigma(X))\right) \tag{23}$$

Now that we have a closed-form expression for the second term, we need to evaluate $\mathbb{E}_{z\sim Q}[\log P(X|z)]$. We can obtain an approximation of it by sampling from $P(z)$ and compute

$$\nabla_\theta \left[\log P(X|z) - \mathbb{KL}(Q(z|X)||P(z))\right] \ . \tag{24}$$

The issue here is that sampling from $P(z)$ is a **non-differentiable** operation, which prevents to backpropagate the loss to $\mu(X;\vartheta)$ and $\Sigma(X;\vartheta)$, which parametrizes Q. To circumvent this issue, we use a **reparametrization trick**. We sample $\epsilon \sim \mathcal{N}(0, I)$, then compute $z = \mu(X;\vartheta) + \Sigma^{1/2}(X;\vartheta) \cdot \epsilon$.

3.3 Intepreting the objective

3.3.1 The error term $\mathbb{KL}(Q(z|X)||P(z|X))$

This error term is of utmost importance to ensure that our VAE samples from the correct distribution.

Theorem 1 (Convergence to true distribution.). *Let $P(X)$ be the modelled distribution and $P^*(X)$ the true distribution. Let n be the number of samples $z \sim P(z)$, then*

$$P(X) \xrightarrow{\mathcal{L}} P^*(X) \Longleftrightarrow \mathbb{KL}(Q(z|X)||P(z|X)) \rightarrow_{n\infty} 0 \ . \tag{25}$$

Intuitively, $\mathbb{KL}(Q(z|X)||P(z|X))$ pulls our model of $P(X)$ towards parametrization that makes $P(z|X)$ Gaussian.

3.3.2 An interpretation from information theory

1. $\mathbb{KL}(Q(z|X)||P(z))$: expected information required to convert an uninformative sample from $P(z)$ to a sample from $Q(z|X)$. It is the extra information that we get about X when z comes from $Q(z|X)$ instead of $P(z)$.
2. $P(X|z)$: amount of information required to reconstruct X from z with an ideal encoding.
3. $-\log P(X)$: total number of bits required to construct X given an ideal encoding using our model.

Therefore, using Eq. (21), we notice that minimizing $-\log P(X)$ is a two-step process:

1. Use some bits to construct z (minimize $\mathbb{KL}(Q(z|X)||P(z))$)
2. Use some bits to reconstruct X from z (minimize $P(X|z)$).

Finally, $\mathbb{KL}(Q(z|X)||P(z))$ is a “price” we have to pay for Q being a sub-optimal encoding.

References

[1] K. P. Murphy. *Machine learning : a probabilistic perspective.* MIT Press, 2013.