Evidence lower bounds with built-in baselines

Justin

December 26, 2022

1 Problem setup

The goal is to maximize the log marginal likelihood,

$$\log p(x) = \log \sum_z p(x, z), \tag{1}$$

for a latent variable model. The derivative of the above is

$$\nabla \log \sum_z p(x, z) = \frac{p(x, z)}{p(x)} \nabla \log p(x, z) = p(z \mid x) \nabla \log p(x, z), \tag{2}$$

the expected gradient under the posterior. When exact marginalization is intractable, a common approach is to introduce a variational approximation to the posterior $q(z \mid x)$ and optimize the lower bound

$$\log p(x) = \log \sum_z q(z \mid x) \frac{p(x, z)}{q(z \mid x)} \geq \sum_z q(z \mid x) \log \frac{p(x, z)}{q(z \mid x)}, \tag{3}$$

which allows for tractable Monte Carlo approximation. [1]

Empirically, we find directly optimizing this lower bound (3) to be more difficult than optimizing the marginal likelihood (1), often requiring a baseline:

$$\begin{aligned}
&\nabla_q \sum_z q(z \mid x) \log \frac{p(x, z)}{q(z \mid x)} \\
&= \nabla_q \sum_z q(z \mid x) \log p(x \mid z) - KL[q(z \mid x) || p(z)] \\
&= \nabla_q \sum_z q(z \mid x)(\log p(x \mid z) - B) - KL[q(z \mid x) || p(z)],
\end{aligned} \tag{4}$$

where the baseline B is not a function of z.[2] Computing this baseline B can be expensive, potentially requiring multiple evaluations of $p(x \mid z)$. A common choice of baseline is the sample-average baseline, where $B = \sum_{z' \in Z} \log p(x \mid z')$ and Z is a set of iid samples.[3] Additionally, even in scenarios where exact marginalization is tractable, prior work has included a baseline. This begs the questions: Is a baseline necessary, and why does optimizing the marginal likelihood not require a baseline?

We show that the gradient of the marginal likelihood (2) already contains a built-in baseline, and use that to derive a simple and computable lower bound of the marginal likelihood (1) whose gradient does not require the manual engineering of a baseline.

[1] The gap between the two is given by $KL[q(z \mid x) || p(z \mid x)]$.

[2] The derivative is a linear operator, and $\nabla \sum_z q(z \mid x) B = B \cdot \nabla \sum_z q(z \mid x) = B \cdot 0$.

[3] A more efficient alternative is the leave-one-out baseline, which is efficient if the results of $p(x \mid z)$ are already available for a set of iid z.

2 Logsumexp

2.1 The gradient of logsumexp approximates the exponentiated regret

The key component of the marginal likelihood is the logsumexp operation, a smooth version of max: $\log \sum_z \exp f(z)$.[4] Similar to the gradient of the marginal likelihood, the gradient of logsumexp is given by

$$\nabla \log \sum_{z' \in Z} \exp f(z') = \frac{e^{f(z)}}{\sum_{z' \in Z} e^{f(z')}} \nabla f(z) = e^{f(z) - \log \sum_{z' \in Z} e^{f(z')}} \nabla f(z) \approx \underbrace{e^{f(z) - \max_{z' \in Z} f(z')}}_{R} \nabla f(z).$$

The last approximation holds because logsumexp is a smooth approximation of max. We can therefore think of $\log \sum_{z' \in Z} \exp f(z') \approx \max_{z' \in Z} f(z)$ as a built-in baseline. R is also the exponentiated regret, the difference between a given $f(z)$ and the best $f(z)$. Since the regret is non-positive, the exponentiated regret is always between 0 and 1.

2.2 Approximating the gradient of logsumexp

Rather than introducing a surrogate loss, such as equation (3), and losing this built-in baseline, we can instead approximate the gradient of logsumexp by using a restriction of logsumexp to $\mathcal{Z}' \subseteq \mathcal{Z}$:

$$\log \sum_{z \in \mathcal{Z}'} \exp f(z). \tag{5}$$

2.3 Gradient estimator analysis

TBD

[4] $\log \sum \exp(f(z))$ is perhaps better known as the log-partition function.