- Importance Weighted Autoencoders
- Yuri Burda, Ruslan Salakhutdinov

Motivation

Let's consider the VAE objective:

$$\log p_\theta(x) - D_{KL}[q_\phi(z|x) \| p_\theta(z|x)] = E_{z\sim q_\phi(z|x)}[\log p_\theta(x|z)] - D_{KL}[q_\phi(z|x) \| p_\theta(z)]$$

The idea is to approximate $p_\theta(z|x)$ with $q_\phi(z|x)$. However, let's now consider what will happend if q_ϕ is not expressive enough, i.e., the true posterior can be easily approximated with simple regression from observations.

Now consider, given fixed ϕ that is the best in terms of approximating p_θ, we try to optimize θ to optimize ELBO, and consider how this affects $p_\theta(x)$. Ideally, if we don't have the KL term on the left, we will achieve θ that maximizes $p_\theta(x)$ by maximizing ELBO. However, in practice, we have another pressure of optimizing θ to make $q_\phi(z|x)$ and $p_\theta(z|x)$ close, we may be achieve best θ that maximizes $p_\theta(x)$. And further, the worse that $q_\phi(z|x)$ models best $p_\theta(z|x)$, the worse this effect will be.

Here have see that the requirement that strong penalty on the simalarity between $q_\phi(z|x)$ and $p_\theta(z|x)$ maybe too harsh. Suppose now our goal is to optimize $p_\theta(x)$, even though $q_\phi(z|x)$ is not so good, intuitively we should believe that, given x, z that has a higher $q_\phi(z|x)$ should also possess a high value of $p_\theta(z|x)$, (i.e., is a good explaination of the data and thus **suitable for estimating** θ, even though overall similarity between $q_\phi(z|x)$ and $p_\theta(z|x)$ is not so good. So maybe place more weight this z's will result in better result.

VAE Objective

The VAE objective can be written as

$$\log p(x) = \log E_{q(h|x)}[\frac{p(x,h)}{q(h|x)}] \geq E_{q(h|x)}[\log \frac{p(x,h)}{q(h|x)}] = \mathcal{L}(x)$$

Using reparametrization trick, the lower bound can be written as

$$E_{\epsilon\sim\mathcal{N}(0,I)}[\log \frac{p(x,h(\epsilon,x,\theta)|\theta)}{q(h(\epsilon,x,\theta)|\theta)}]$$

And the gradient estimation using Monte Carlo:

$$\frac{1}{k}\sum_{i=1}^{k} \nabla_\theta \log w(x, h(\epsilon_i, x, \theta), \theta)$$

where $w(x,h,\theta) = p(x,h|\theta)/q(h|x,\theta)$. Each of these is an estimate of $\mathcal{L}(x)$.

Importance Weighted Autoencoder

The fundamental difference is the lower bound used. In IWAE, this is $\mathcal{L}_k(x)$, where k is a hyperparameter:

$$\mathcal{L}_k(x) = E_{h_1,...,h_k \sim q(h|x)}[\log \frac{1}{k} \sum_{i=1}^{k} \frac{p(x,h_i)}{q(h_i|x)}]$$

Now comes the two major results about this lower bound.

- It is a tigher lower bound. For all k,

  $$\log p(x) \geq \mathcal{L}_{k+1} \geq \mathcal{L}_k$$

  And the larger k, the tigher.
- The gradient of this objective with respect to θ can be written as

  $$\sum_{i=1}^{k} \tilde{w}_i \nabla_\theta \log w(x, h(\epsilon_i, x, \theta), \theta)$$

  where $\tilde{w}_i = \frac{w_i}{\sum_{i=1}^{k} w_i}$ is the normalize weight for its gradient.

The second one has the good interpretation that it places higher weights on better samples, and tends to optimize θ using these samples.

Experiments

On MNIST, the author shows that:

- Larger k does not improve the performance of VAE much
- Instead, larger k results in siginificantly better result for IWAE

The author also measures the percent of "active dimensions" in the latent variable. And they discover that using IWAE objective allows more active dimensions.