ASSIGNMENT 4: THEORY OF GENERATIVE MODELS [IFT6135]

JOSEPH D. VIVIANO

1. Reparameterization Trick of Variational Autoencoders

1.1. **Transformation of Gaussian Noise.** First let's define z as a linear transformation applied to a gaussian distribution with zero mean ϵ:

$$z = \mu(x + \sigma(x \odot \epsilon$$

The expectation of z is as follows. Crucially, note that the expectation of our normal distribution $E\epsilon = 0$:

$$\begin{aligned} E(z) &= E(\mu(x) + E\left(\sigma(x \odot \epsilon\right) \\ &= \mu(x) + \sigma(x \odot 0) \end{aligned} \tag{1.1}$$

Therefore, it follows that $E(z) = \mu(x)$. We can do a similar calculation for $\sigma^2(z)$ to show that $\sigma^2(z) = \sigma^2(x)$ and therefore z has a gaussian with $z \sim \mathcal{N}(\mu(x, \sigma^2(x)$.

Now let's look at the k^2 dimensional output of a neural network: $z = \mu(x + S(x \odot \epsilon$, where $S(x)$ is our output. This gives us the same as for eq 1.1, replacing σ for S:

$$\begin{aligned} E(z) &= E(\mu(x) + E\left(S(x \odot \epsilon\right) \\ &= \mu(x) \end{aligned} \tag{1.2}$$

In contrast, the calculations for variance leads to a due to the squaring of σ. When one replaces S for σ, one is left with the term $S(x)S(x)^T$, so $\sigma^2(z) = \left(S(x)S(x)^T\right)$. Whereas before the variance term of our distribution was $\sigma^2(x)$, it is now $\left(S(x)S(x)^T\right)$, as in $z \sim \mathcal{N}(\mu(x), \left(S(x)S(x)^T\right))$.

1.2. **Encoders vs. Mean Fields.** In general, the inference network used in in a variational autoencoder will outperform traditional mean field methods which factorize the variational distribution as a product of distributions. More specifically, mean field variational inference assumes the variational family factorizes:

Date: April 2018.

$$q(z_1, z_2, ..., z_m) = \prod_{j=1}^{m} q(z_j)$$

with each variable being independent. In practice, however, this is rarely true, and the dependencies between these variables makes the posterior hard to work with. The mean field approach requires solutions to the expectations taken on the posterior, which is generally intractable. The variational autoencoder gets around this requirement by only approximating the posterior using the probabilistic encoder $q_\phi(z|x)$, which approximates the true generating model $p_\theta(x, z)$, and where the parameters ϕ and θ are optimized jointly by the variational autoencoder algorithm.

2. Importance Weighted Autoencoder

2.1. **IWLB as a Lower Bound on log Likelihood.** We can show that IWLB is a lower bound on the log likelihood $\log p(x)$ by using Jensen's inequality (i.e., that a secant line of a concave function lies below the graph), the fact that log is concave, and the fact that $p(x, z_i) = p(x|z_i)p(x)$. Basicially, pull the log and denominator out of the bracket, and push the expectation to $p(x)$ (since $p(x) = Ep(x)$) to resolve to $\log(p(x))$:

$$\begin{aligned} \mathcal{L}_k &= E\left[\log \frac{1}{k}\sum_{i=1}^{k} \frac{p(x, z_i)}{q(z_i|x)}\right] \\ &\leq \log \frac{1}{k}\sum_{i=1}^{k} E(p(x)) \\ &= \log(p(x)) \end{aligned} \tag{2.1}$$

2.2. **IWLB with k=2 is tighter than ELBO with k=1.** First, observe the VAE loss function. Isn't it nice?

$$\begin{aligned} \log p(x) &\geq E_{q(z|x)}\left[\log \frac{p(x, z)}{q(z|x)}\right] \\ &= \log(p(x)) - D_{KL}\left(q(z|x)||p(z|x)\right) \\ &= \mathcal{L}_1 \end{aligned} \tag{2.2}$$

If we can demonstrate that $\mathcal{L}_K$ approaches $\log p(x)$ as K grows, then we can claim that $\mathcal{L}_2$ is a tighter bound than ELBO. I'm sorry for the crappy formatting below but exams are killing me and I hope you understand ;).

Suppose $\frac{p(x,y)}{p(y|x)}$ is smaller than $\frac{p(a,b)}{p(b|a)}$. Therefore, $E\left[\log \frac{1}{2}\left(\frac{p(x,y)}{q(y|x)} + \frac{p(a,b)}{q(b|a)}\right)\right] \leq E\left[\log \frac{1}{2}\left(\frac{p(x,y)}{q(y|x)} + \frac{p(x,y)}{q(y|x)}\right)\right]$.

If $\frac{p(x,y)}{p(y|x)}$ is the VAE loss function above, we can say with certianty that $E\left[\log \frac{1}{2}\left(\frac{p(x,y)}{q(y|x)} + \frac{p(a,b)}{q(b|a)}\right)\right] > \mathcal{L}_1$. It is also the exact formula for $\mathcal{L}_2$.
When we sub in the smaller $\frac{p(x,y)}{q(y|x)}$ in place of the larger $\frac{p(a,b)}{q(b|a)}$, our answer shows that $\mathcal{L}_1 \leq \mathcal{L}_2$.

3. Maximum Likelihood for Generative Adversarial Networks

The original GAN objective can be writen as:

$$\max_D \mathbf{E}_{p\,D}(x)\big[\log D(x)\big] + \mathbf{E}_{p_G}\big[\log(1 - D(G(z)))\big]; \quad \max_G \mathbf{E}_{p_G}\big[\log D(G(z))\big]$$

Note that the definition of an optimial discriminator using this notation is:

$$D^*(\mathbf{x}) = \frac{p_D(\mathbf{x})}{p_G(\mathbf{x}) + p_D(\mathbf{x})} \tag{3.1}$$

The goal here is to find the maximum likelihood objective (cost function) instead of the negative log liklihood objective to apply to samples coming from the generator $\mathbf{E}_{p_G}\big[f(D(G(z)))\big]$, where $f(D(G(z)))$ must be found.

As a reminder, the maximum likelihood estimate in this case would be:

$$\hat{\theta} \in \{\arg\max_{\theta \in \Theta} \{(\theta\,; x)\}$$

Each step of learning in a GAN consists of reducing the expectation $f(x)$ run on a bunch of samples pulled from generator G, which we express as

$$\mathbf{E}_{x\ p_G} f(x) \tag{3.2}$$

here, p_G represents a sample pulled from the probability distribution generated by G. First, let's take partial derivative with respect to the weights θ on a sample $x\ p_G$ from the generator and represent it as an integral. Then we applied Leibniz's rule, and sub in the identity $\frac{\partial}{\partial\theta} p_G(x) = p_G(x)\frac{\partial}{\partial\theta} log p_G(x)$:

$$\begin{aligned} \frac{\partial}{\partial\theta}\mathbf{E}_{x\ p_G} f(x) &= \int f(x)\frac{\partial}{\partial\theta} p_G(x) \\ &= \int f(x)\frac{\partial}{\partial\theta} p_G(X)dx \\ &= \int f(x) p_G(x)\frac{\partial}{\partial\theta} log p_G(x) \end{aligned} \tag{3.3}$$

While tells us that we can express the aformentioned expectation (3.2) as:

$$\mathbf{E}_{x\ p_G} f(x) \frac{\partial}{\partial \theta} log p_G(x) \tag{3.4}$$

This tells us the maximum likelihood can be found given:

$$f(x) = -\frac{p_D(x)}{p_G x} \tag{3.5}$$

Now if we assume that our optimal discriminator $D^*(\mathbf{x})$ from (3.1) is the logistic sigmoid $\sigma(a(x))$ then

$$\sigma(a(x)) = \frac{p_D(\mathbf{x})}{p_G(\mathbf{x}) + p_D(\mathbf{x})} \tag{3.6}$$

And it follows that

$$f(x) = -\exp(a(x)) \tag{3.7}$$

which is our function f such that the objective corresponds to maximum liklihood.

Université de Montréal
E-mail address: joseph@viviano.ca