Variational Auto Encoder

Ruichen Wang

November 30, 2018

Abstract

Variational auto-encoder [2] is a very powful generative model. It can be used to generate or convert videos, images, texts, sounds etc.

Contents

1 What is VAE?

Variational auto-encoder is a brilliant combination of deep learning and variational inferece. It was proposed by Kingma in 2013. It provides a probabilistic manner for describing an observation in latent space. The encoder is aimed to describe a probability distribution for each latent attribute.

1.1 Intuition

In the past, we want the encoder to learn some dimensions of input as the compressed feature. Using a variational autoencoder, we describe those latent dimensions in probabilistic terms.we'll now instead represent each latent attribute for a given input as a probability distribution. And we will perform random sampling on the distirbution to feed the decoder. We expecting the decoder can accurately reconstruct the input.

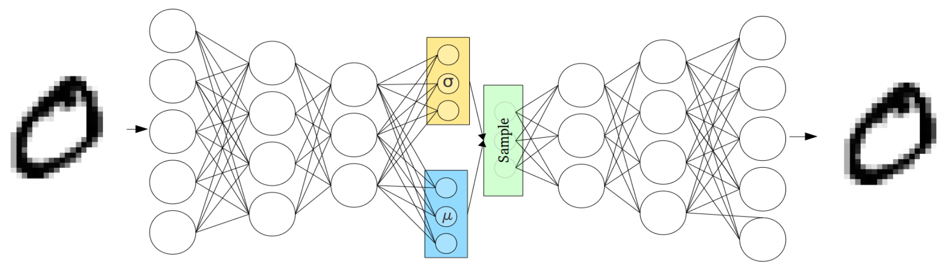

Figure 1: VAE graph model

1.2 Statisical motivation

Suppose there exists some latent variable z controls the observation x. We would like to infer the posterior $p_\theta(z|x)$

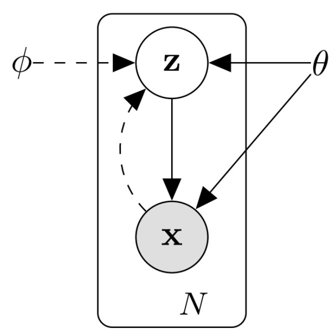

Figure 2: Solid lines denotes the generative model $p_\theta(z)p_\theta(x|z)$. Dash lines denote the variational inference $q_\phi(z|x)$ which is a approximation of intractable $p_\theta(z|x)$.

$$p_\theta(z|x) = \frac{p_\theta(x|z)p_\theta(z)}{p_\theta(x)} = \frac{p_\theta(x|z)p_\theta(z)}{\int p_\theta(z)p_\theta(x|z)dz}$$

As we **do not** make the common simplifying assumptions about the marginal or posterior probabilities, the $\int p_\theta(z)p_\theta(x|z)dz$ is intractable, and EM algorithm or mean-field variational bayesian is also intractable.

So the VAE introduce a recognition model $q_\phi(z|x)$, which is a approximation to the posterior. Note the ϕ can not be computed from some closed-form expectation like mean-filed variational inference. It will be learned jointly with θ.

2 Method

Remember the KL divergence can be used to measure the difference between distributions. We want to minimize the KL below:

$$minD_{KL}(q_\phi(z|x)||p_\theta(z|x))$$

2.1 Derivation of Variational Bound

In this section, we will derivate the objective loss that VAE optimize.

$$\begin{aligned}
D_{KL}(q_\phi(z|x)||p_\theta(z|x)) &= E_{q_\phi(z|x)}\left[log\frac{q_\phi(z|x)}{p_\theta(z|x)}\right] \\
&= E_{q_\phi(z|x)}logq_\phi(z|x) - E_{q_\phi(z|x)}logp_\theta(z|x) \\
&= E_{q_\phi(z|x)}logq_\phi(z|x) - E_{q_\phi(z|x)}[logp_\theta(x,z) - logp_\theta(x)] \\
&= logp_\theta(x) + E_{q_\phi(z|x)}logq_\phi(z|x) - E_{q_\phi(z|x)}logp_\theta(x,z)
\end{aligned}$$

As $logp_\theta(x)$ is fixed. The fomula can also be denoted as:

$$logp_\theta(x) = D_{KL}(q_\phi(z|x)||p_\theta(z|x)) + E_{q_\phi(z|x)}logp_\theta(x,z) - E_{q_\phi(z|x)}logq_\phi(z|x)$$

Does this looks familiar to you? It is the evidence lower bound(ELBO). minimize the KL divergence is equal to maximize the ELBO. So now we convert our goal to :

$$\begin{aligned}
max\mathcal{L} &= E_{q_\phi(z|x)}logp_\theta(x,z) - E_{q_\phi(z|x)}logq_\phi(z|x) \\
&= E_{q_\phi(z|x)}[logp_\theta(x|z) + logp_\theta(z)] - E_{q_\phi(z|x)}logq_\phi(z|x) \\
&= E_{q_\phi(z|x)}logp_\theta(x|z) - D_{KL}(q_\phi(z|x)||p_\theta(z))
\end{aligned}$$

As you can see, the first term is the negative cross entropy $-H(q_\phi(z|x), p_\theta(x|z))$, which measures the reconstruction likelihood. The second term can be viewed as the regulation of $q_\phi(z|x)$, which encouraging the prior $p_\theta(z)$ to be closed to the aprroximate posterior $q_\phi(z|x)$.

2.2 Reparameterization Trick

In the loss function, we invoked a distribution $q_\phi(z|x)$, which will generate sameple latent variables from observation. As z is sampled from some latent distirbution, it is not able to calculate the gradient. We use a method called **reparameterization trick** to rewrite the expectation in order to backpropagate.

2.2.1 Example

Assume we have a normal distribution q that is parameterized by θ, specially $q_\theta(x) = N(\theta, 1)$. We want to solve the below problem:

$$\arg\min_\theta E_q(x^2)$$

It is quite obvious that $E(x^2) = E(x)^2 + D(x) = \theta^2 + 1$. $\theta = 0$ is the answer. We want to see how reparameterization trick can help us solve this problem in calculating the gradients.

$$\begin{aligned}\nabla_\theta E_q[x^2] &= \nabla_\theta \int q_\theta(x) x^2 dx \\ &= \int x^2 \nabla_\theta q_\theta(x) \frac{q_\theta(x)}{q_\theta(x)} dx \\ &= \int q_\theta(x) x^2 \nabla_\theta log q_\theta(x) dx \\ &= E_q[x^2 \nabla_\theta log q_\theta(x)] \\ &= E_q[x^2(x-\theta)]\end{aligned}$$

As you can see the expectation is based on q_θ. Using reparameterization trick can rewrite the expectation so that the distribution is independent with θ.

We make $p(\epsilon) \sim N(0,1)$, $x = \theta + \epsilon$. Hence:

$$\nabla_\theta E_q[x^2] = E_p[\nabla_\theta(\theta + \epsilon)^2] = E_p(2(\theta + \epsilon))$$

Now we can also get our result $\theta = 0$. Using reparameterization trick can also make the variance of gradience more stable.

2.2.2 Reparametrization for the Posterior

First we sample a noise variable

$$\epsilon \sim p(\epsilon) = N(0,1)$$

Then we apply a transform $g_\phi(\epsilon, x)$ that maps the random noise to a complex distribution.

$$z = g_\phi(\epsilon, x)$$

Here we choose gaussian $z \sim q_{\mu,\sigma}(z) = N(\mu, \sigma)$, which is also can be denoted as :

$$z = g_{\mu,\sigma}(\epsilon) = \mu + \epsilon \cdot \sigma$$

The biggest advantage of this approach is that we many now write the gradient as:

$$\nabla_\phi E_{q(z|x)}[f(x,z)] = \nabla_\phi E_{p(\epsilon)}[f(x, g(\epsilon, x))] = E_{p(\epsilon)}[\nabla_\phi f(x, g(\epsilon, x))]$$

So we can write the frist term of ELBO as :

$$E_{q_\phi(z|x)} log p_\theta(x|z) = E_{p(\epsilon)}[log p_\theta(x|g(\epsilon, x))] = \frac{1}{L} \sum_{l=1}^{L} log p_\theta(x|g(\epsilon^l, x))$$

Now we take the sampling operation outside the network. Now we just perform sampling operation on ϵ instead of z.

2.3 KL Divergence

Now let's decide which prior knowledge $p_\theta(z)$ to use, as $p_\theta(z)$ comes from our assumptions. It can be any arbitrary function. However, there are some pre-requisite. First it should be flexible enough to represent the richness of the data. Second , it should be easy to sample.

Here we can just choose $p_\theta(z) \sim N(0,1)$, which means our assumption posterior $q_\phi(z|x)$ should be closed to $N(0,1)$. The KL divergence can be denoted as :

$$\begin{aligned} D_{KL}(q_\phi(z|x)||p_\theta(z)) &= D_{KL}[N(\mu(x), \sigma(x)||N(0,1))] \\ &= \frac{1}{2}(\sigma^2 + \mu^2 - log\sigma^2 - 1) \end{aligned}$$

Now the full loss function will be :

$$\arg\max_{\theta,\mu,\sigma} \mathcal{L} = \frac{1}{L}\sum_{l=1}^{L} logp_\theta(x|g(\epsilon^l, x)) - \frac{1}{2}\sum_{j=1}^{J}(\sigma_j^2 + \mu_j^2 - log\sigma_j^2 - 1)$$

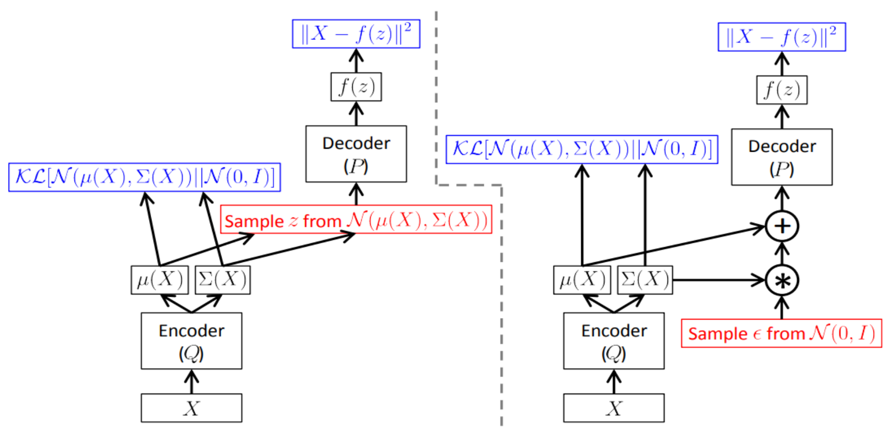

Figure 3: Left without reparameterization trick. Right with it. The sampling operation is equivalent. But backpropagation can only be applied to the right.[1]

References

[1] Carl Doersch. Tutorial on variational autoencoders. *CoRR*, abs/1606.05908, 2016.

[2] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. *CoRR*, abs/1312.6114, 2013.

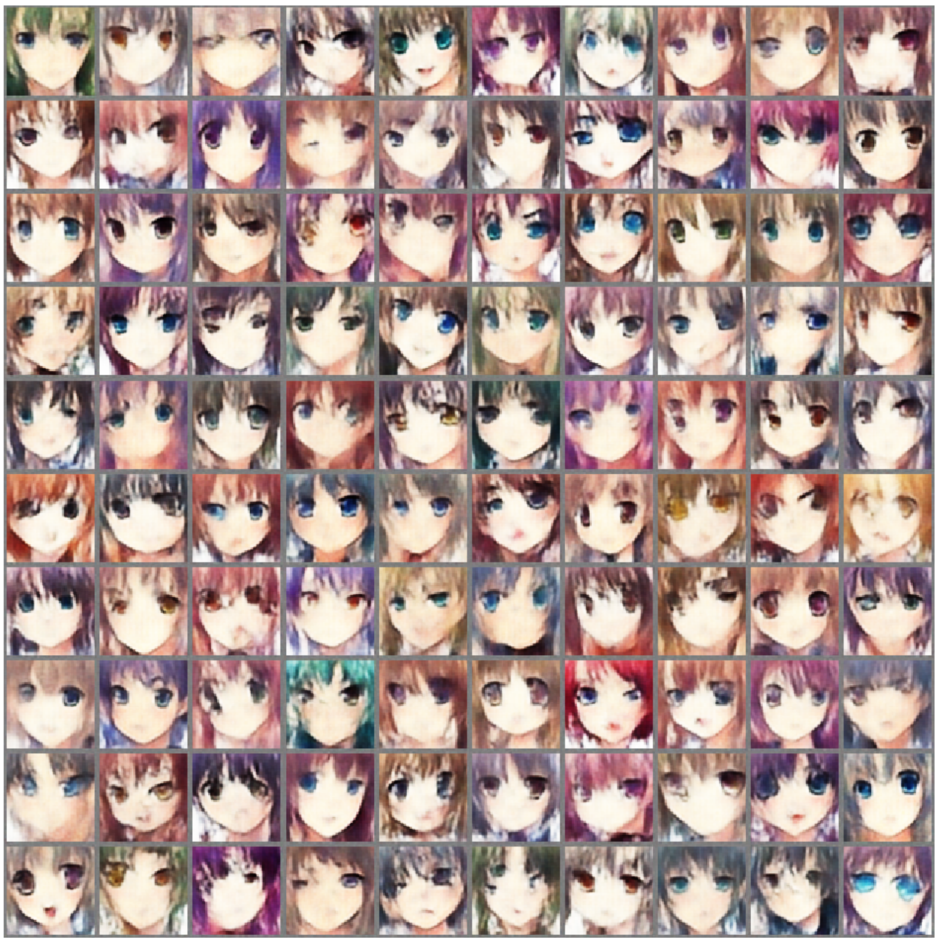

Figure 4: Generated images