Mixture GAN

Tikhon Mavrin [1]

Abstract

In the current article, we will look at the mixture GAN – generative adversarial network, which learns to distinguish not the true distribution and the generated one, but their mixtures.

1. Problem statement

In GAN we play min-max game:

$$\min_G \max_D \mathbb{E}_{x\sim p_{\text{data}}(x)}[\log D(x)]+\mathbb{E}_{z\sim p_z(z)}[\log(1-D(G(z)))] \tag{1}$$

It is equal to:

$$\min_G \max_D \mathbb{E}_{x\sim p_{\text{data}}(x)}[\log D(x)]+\mathbb{E}_{x\sim p_g(x)}[\log(1-D(x))] \tag{2}$$

For G fixed, the optimal discriminator D is

$$D^*_G(x)=\frac{p_{\text{data}}(x)}{p_{\text{data}}(x)+p_g(x)} \tag{3}$$

When $p_g(x)=0$ or $p_{\text{data}}(x)=0$, the derivative of $D^*_G(x)$ equals to zero, which can lead to optimization problems.

1.1. Our approach

Instead of learning to distinguish the true distribution and the generated one, we will work with their mixtures:

$$\mathbb{E}_{t\sim\mu_{[0,1]}}\left[\mathbb{E}_{x_d\sim p_{\text{data}}(x_d),x_g\sim p_g(x_g),x_t=\frac{1+t}{2}x_d+\frac{1-t}{2}x_g}[\log D(x_t,t)]+\mathbb{E}_{x_d\sim p_{\text{data}}(x_d),x_g\sim p_g(x_g),x_t=\frac{1-t}{2}x_d+\frac{1+t}{2}x_g}[\log(1-D(x_t,t))]\right] \tag{4}$$

In the special case where $\mu_{[0,1]}$ is a delta function of 1, we get the vanilla gan.

When t is equal to one, then it is GAN inside, when t is equal to zero, mixtures are the same and we have:

$$D^*_G(x_0,0)=\frac{1}{2} \tag{5}$$

Now for every t we sample x_t from mixtures, not initial distributions, which can help with derivative problem.

1.2. Comparison of approaches

This vanishing gradient problem can be partially solved using WGAN with gradient clipping or using WGAN-GP with regularizations. In WGAN-GP also use mixtures for regularization. Or you can use spectral normalization.

Also people use Non-Saturating (Modified Minimax) Loss in vanilla GAN.

Our approach in future can use WGAN ideas and add gradient regularizations on both x and t dimensions. So I hope that we can solve vanishing gradient problem with our method and also use WGAN ideas here.

1.3. Baseline

As a baseline we trained model with different we trained models with different $\mu_{[0,1]}$. All models were able to learn the distributions with varying quality and converge to an acceptable result.

[1]Skoltech. Correspondence to: Tikhon Mavrin <Tikhon.Mavrin@skoltech.ru>.