BEGAN: Boundary Equilibrium Generative Adversarial Networks

August 8, 2022

Useful papers - [BSM17],

1 Introduction

In this work authors sought to solve most common practice problem of GAN - **unstable training**. Firstly, it is necessary to remind important things from Vanilla GAN and WGAN.

The vanilla GAN objective has the following form:

$$\begin{gathered}\mathbb{L}(G, D) = \mathbb{E}_{p_{data}(x)}\left[\log D(x)\right] + \mathbb{E}_{p(z)}\left[\log(1 - D(G(z))\right] \\ \textbf{Discriminator: } \operatorname{argmax}\mathbb{L}(G, D) - ? \\ \textbf{Generator: } \operatorname{argmin}\mathbb{L}(G, D) - ?\end{gathered} \tag{1}$$

Here $G(z)$ has same shape as input x, $D(x) \in [0, 1]$ and gives the probability that sample is from real data. Such approach matches distribution of samples generated by G and real distribution $p_{data}(x)$. Many studies argues that unstable training comes from **disjoint supports** of these distributions. As a result of this, gradients of the discriminator disappear and generator cannot be trained.

In WGAN paper authors aimed to solve mentioned issue proposing model based on Wasserstein distance. Their objective has the following form:

$$\begin{gathered}\mathbb{L}(G, F) = \mathbb{E}_{p_{data}(x)}\left[F(x)\right] - \mathbb{E}_{p(z)}\left[F(G(z)\right] \\ \textbf{Critic: } \operatorname{argmax}\mathbb{L}(G, F) - ? \\ \textbf{Generator: } \operatorname{argmin}\mathbb{L}(G, F) - ?\end{gathered} \tag{2}$$

Looks pretty similar, but with significant difference: 1) instead of discriminator we have critic, $F(x) \in \mathbb{R}$; 2) original problem has no max/min nature. In other words, critic and generator are trying to solve problem jointly. But in original GAN discriminator and generator are opposed. Authors claimed that critic has no problems with gradients like discriminator. I think this is because of the fact that $F(x) \in \mathbb{R}$, that is critic has linear activation function at the end (not sigmoid like discriminator). Important to note that we still trying to match distribution of samples generated by G and real distribution $p_{data}(x)$ (because originally WGAN comes from Wasserstein distance between distributions). So, we **still have a problem with disjoint supports**. It is there, but it does

not affect optimizations as much as in original GAN. By the way, we want to solve this issue. This is where BEGAN comes in (i think :)).

As was said before, in WGAN we match two distributions (let $p_1(x_1), p_2(x_2)$) via Wasserstein distance:

$$\mathbb{W}(p_1, p_2) = \inf_{\gamma \in \Gamma(p_1, p_2)} \mathbb{E}_{\gamma(x_1, x_2)} ||x_1 - x_2|| \tag{3}$$

where γ - transport plan (some joint distribution on x_1, x_2), Γ - set of all transport plans. Using Jensen's inequality:

$$\begin{aligned} \mathbb{W}(p_1, p_2) &= \inf_{\gamma \in \Gamma(p_1, p_2)} \mathbb{E}_{\gamma(x_1, x_2)} ||x_1 - x_2|| \\ &\geq \inf_{\gamma \in \Gamma(p_1, p_2)} |\, \mathbb{E}_{\gamma(x_1, x_2)}\, x_1 - \mathbb{E}_{\gamma(x_1, x_2)}\, x_2| \end{aligned} \tag{4}$$

Here we need to use one important property of transport plan:

$$\begin{aligned} \int dx_1 \gamma(x_1, x_2) = p_2(x_2) \\ \int dx_2 \gamma(x_1, x_2) = p_1(x_1) \end{aligned} \tag{5}$$

It means that if we execute the transport plan by some variable, we obtain distribution of another variable. Then

$$\begin{aligned} &\inf_{\gamma \in \Gamma(p_1, p_2)} |\, \mathbb{E}_{\gamma(x_1, x_2)}\, x_1 - \mathbb{E}_{\gamma(x_1, x_2)}\, x_2| \\ &= \inf_{\gamma \in \Gamma(p_1, p_2)} |\, \mathbb{E}_{p_1(x_1)}\, x_1 - \mathbb{E}_{p_2(x_2)}\, x_2| \\ &= |\, \mathbb{E}_{p_1(x_1)}\, x_1 - \mathbb{E}_{p_2(x_2)}\, x_2| \end{aligned} \tag{6}$$

So, we obtain that Wasserstein distance between two distributions can be evaluated like a difference of expectations of variables (or simply mean) from these distributions. That sounds good, but it is does not solve our problem of disjoints supports of two distributions. So, what authors said: **lets consider distance not between real distribution and distribution of generator but difference between distributions of error of autoencoder**. Sounds tricky, lets consider in more detail. Lets say that $p_1 = p_{data}$, $p_2 = p_G$. So, we change p_{data}, p_G to some q_{data}, q_G, where q_{data} and q_G it is distributions of errors of autoencoder for real data and fake data, respectively. These distributions can be obtained by following transformations:

$$x \sim p_{data(G)}(x) \to x_{ae} = \mathbf{AE}_\phi(x) \to l = |x - x_{ae}| \to l \sim q_{data(G)}(l) \tag{7}$$

In such a way we obtain samples from distributions of error. Then Wasserstein distance has the following form:

$$\begin{aligned} \mathbb{W}(q_{data}, q_G) &\geq \left|\mathbb{E}_{q_{data}(l)}\, l - \mathbb{E}_{q_G(l)}\, l\right| \\ &= \left|\mathbb{E}_{p_{data}(x)}\, |x - \mathbf{AE}_\phi(x)| - \mathbb{E}_{p_G(x)}\, |x - \mathbf{AE}_\phi(x)|\right| \\ &= \left|\mathbb{E}_{p_{data}(x)}\, |x - \mathbf{AE}_\phi(x)| - \mathbb{E}_{p(z)}\, |\mathbf{G}_\theta(z) - \mathbf{AE}_\phi(\mathbf{G}_\theta(z))|\right| \\ &= \mathbb{E}_{p(z)}\, |\mathbf{G}_\theta(z) - \mathbf{AE}_\phi(\mathbf{G}_\theta(z))| - \mathbb{E}_{p_{data}(x)}\, |x - \mathbf{AE}_\phi(x)| \end{aligned} \tag{8}$$

Here we used LOTUS (two times), and also authors open the module as a negative value. So similar to the previous ones, we can write that BEGAN objective:

$$\begin{gathered}\mathbb{L}(G, AE) = \mathbb{E}_{p(z)} \left|\mathbf{G}(z) - \mathbf{AE}(\mathbf{G}(z))\right| - \mathbb{E}_{p_{data}(x)} \left|x - \mathbf{AE}(x)\right| \\ \textbf{Autoencoder} : \operatorname{argmax} \mathbb{L}(G, AE) - ? \\ \textbf{Generator} : \operatorname{argmin} \mathbb{L}(G, AE) - ?\end{gathered} \tag{9}$$

So, discriminator (autoencoder) is trying to make errors for real samples lower while for fake samples higher. Generator works opposite. Such approach more robust that vanilla GAN due to: 1) it is more likely that supports of noise distributions will be intersected; 2) autoencoder is very easy train as opposed to classifier in GAN (as you can remember we need discriminator to be optimal).

A fair question may arise: as we said, we match distributions of errors for real and fake samples, but why does the equality of these distributions guarantee us the equality of p_{data} and p_G? Ok, lets $q_{data} = q_G$, then

$$\begin{gathered}\mathbb{W}(q_{data}, q_G) = 0 \rightarrow \\ \left|\mathbb{E}_{p_{data}(x)} \left|x - \mathbf{AE}_\phi(x)\right| - \mathbb{E}_{p_G(x)} \left|x - \mathbf{AE}_\phi(x)\right|\right| = 0\end{gathered} \tag{10}$$

When it might happens:

- If $p_{data} = p_G$, that is what we need
- If autoencoder works perfectly, i.e. $x = \mathbf{AE}_\phi(x), \mathbf{G}_\theta(z) = \mathbf{AE}_\phi(\mathbf{G}_\theta(z))$. But this is might happened if and only if $p_{data} = p_G$ because our discriminator force to maximize error on fake samples, so autoencoder becomes ideal if fake samples equals to real ones
- Lastly, lets say that autoencoder works not perfectly

$$\begin{gathered}\left|\mathbb{E}_{p_{data}(x)} \left|x - \mathbf{AE}_\phi(x)\right| - \mathbb{E}_{p_G(x)} \left|x - \mathbf{AE}_\phi(x)\right|\right| = 0 \\ \int p_{data}(x) f(x) dx = \int p_G(x) f(x) dx\end{gathered} \tag{11}$$

  But here we cannot say anything, that is, i cannot prove that p_{data} should equal p_G :((, because integrals might equals if integrand functions are different. In simple words, might happened that autoencoder works not perfectly and distribution of errors equals.

Lets look at the discriminator loss in more detail. As it can be seen we need to maximize $\mathbb{E}_{p(z)} \left|\mathbf{G}(z) - \mathbf{AE}(\mathbf{G}(z))\right|$ and minimize $\mathbb{E}_{p_{data}(x)} \left|x - \mathbf{AE}(x)\right|$. But these goals are opposite. The first one is responsible for **distinguish properties** and the second one for **auto encoding properties**. In other words, we push discriminator in such a way that for fake samples it makes bad reconstruction (because it is fake) and for real samples it makes good reconstruction. Authors proposed following ratio to control these terms:

$$\gamma = \frac{\mathbb{E}_{p(z)} \left|\mathbf{G}(z) - \mathbf{AE}(\mathbf{G}(z))\right|}{\mathbb{E}_{p_{data}(x)} \left|x - \mathbf{AE}(x)\right|} \tag{12}$$

I do not understand it: Lower values of γ lead to lower image diversity because the discriminator focuses more heavily on auto-encoding real images. We will refer to γ as

the diversity ratio. There is a natural boundary for which images are sharp and have details

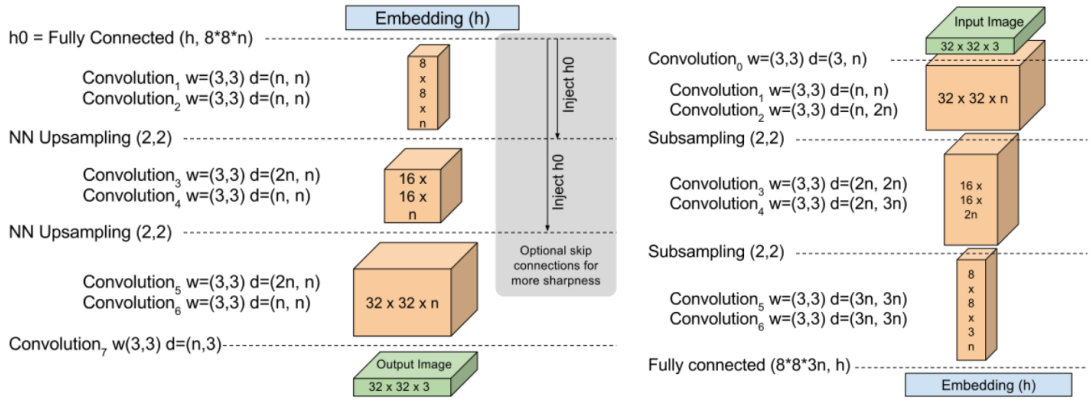
So, our final objective:

$$\begin{aligned}
\phi^{k+1} &= \phi^k - \nabla_\phi \left[\frac{1}{n} \sum_{i=1}^{n} |x_i - \mathbf{AE}_\phi(x_i)| - k_t \frac{1}{n} \sum_{i=1}^{n} |\mathbf{G}_\theta(z_i) - \mathbf{AE}_\phi(\mathbf{G}_\theta(z_i))| \right] \\
\theta^{k+1} &= \theta^k - \nabla_\theta \left[\frac{1}{n} \sum_{i=1}^{n} |\mathbf{G}_\theta(z_i) - \mathbf{AE}_\phi(\mathbf{G}_\theta(z_i))| \right] \\
k_{t+1} &= k_t + \lambda_k \left(\gamma \frac{1}{n} \sum_{i=1}^{n} |x_i - \mathbf{AE}_\phi(x_i)| - \frac{1}{n} \sum_{i=1}^{n} |\mathbf{G}_\theta(z_i) - \mathbf{AE}_\phi(\mathbf{G}_\theta(z_i))| \right)
\end{aligned} \tag{13}$$

$k_0 = 0, \lambda_k = 0.001$. Authors proposed convergence measure as (again do not understand):

$$\mathbb{M}_{global} = \mathbb{E}_{p_{data}(x)} |x - \mathbf{AE}(x)| + \left| \gamma \, \mathbb{E}_{p_{data}(x)} |x - \mathbf{AE}(x)| - \mathbb{E}_{p(z)} |\mathbf{G}(z) - \mathbf{AE}(\mathbf{G}(z))| \right| \tag{14}$$

The architecture of generator and discriminator has the following form 1: Discriminator



(a) Generator/Decoder (b) Encoder

Figure 1: Generator and discriminator architecture

consist of encoder and decoder, generator only of decoder.

References

[BSM17] David Berthelot, Thomas Schumm, and Luke Metz. Began: Boundary equilibrium generative adversarial networks. *arXiv preprint arXiv:1703.10717*, 2017.