IFT 6085 - Lecture 13
Wasserstein GANs

Scribe(s): Yuchen Lu, Kyle Kastner, **Instructor:** Ioannis Mitliagkas

1 Summary

Last lecture, we talked about generative models and maximum likelihood estimation. In this lecture, we are going to focus on another way of learning generative model called Generative Adversarial Network (GAN) [5], and we will be focused on one of the variants, Wasserstein GAN [1].

2 GAN Motivation

In a GAN setting, we have the following definition. P_d is the true data distribution, P_θ is the data distribution parameterized by θ. Normally we use a maximum likelihood estimation (MLE) objective [11] [2] to train P_θ

Definition 1 (MLE Objective)**.**

$$\max_{\theta \in \mathbb{R}^d} \frac{1}{m} \sum_{i=1}^{m} \log P_\theta(x^{(i)})$$

where $x^{(i)}$ be the ith data point.

This objective is related to KL divergence, specifically the MLE is the empirical risk minimizer for the KL-divergence [15].

Theorem 2 (MLE objective and KL divergence relation)**.** *In the limit $m \to \infty$, MLE objective is equivalent to minimizing $KL(P_d|P_\theta)$, which is*

$$\int_x P_d(x) \log \frac{P_d(x)}{P_\theta(x)} dx$$

Proof. Starting from Definition 1, we have

$$\begin{aligned}
\theta^* &= \lim_{m \to \infty} \operatorname*{argmax}_{\theta \in \mathbb{R}^d} \frac{1}{m} \sum_{i=1}^{m} \log P_\theta(x^{(i)}) = \operatorname*{argmax}_{\theta \in \mathbb{R}^d} \int_x P_d(x) \log P_\theta(x) dx \\
&= \operatorname*{argmin}_{\theta \in \mathbb{R}^d} \int_x -P_d(x) \log P_\theta(x) dx \\
&= \operatorname*{argmin}_{\theta \in \mathbb{R}^d} \int_x -P_d(x) \log P_\theta(x) + P_d(x) \log P_d(x) dx \\
&= \operatorname*{argmin}_{\theta \in \mathbb{R}^d} \int_x P_d(x) \log \frac{P_d(x)}{P_\theta(x)} dx = \operatorname*{argmin}_{\theta \in \mathbb{R}^d} KL(P_d|P_\theta)
\end{aligned}$$

□

When $P_\theta(x) = 0$ but $P_d(x) \neq 0$, this metric blows up. A typical fix is to add a small amount of noise, or bound $P_\theta(x) > 0$.

3 Generative Adversarial Networks

As first seen in lecture 12, GAN is an alternative approach, which defines the likelihood function implicitly.
In a GAN, we parameterize P_θ with a generator function G_θ.

Definition 3 (Generator)**.** *A differentiable function from some Z to X.*

$$G_\theta : Z \to X$$

where Z comes from a common distribution like normal or uniform.

After generating data by sampling X, in the original formulation the samples are then scored using a "critic", or disciminator function D_θ, which attempts to classify the generated samples as "fake" (label 0) and "real" (label 1).

Definition 4 (Disciminator)**.** *A differentiable function from some mixture of sampled and real data X to a set of probabilities P_r in $[0, 1]$.*

$$D_\theta : X \to P_r$$

The combined objective of the generator and discriminator forms a minimax game.

Definition 5 (GAN Objective)**.**

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

Both the generator G_θ and the discriminator D_θ are often parameterized using neural networks, and the composite objective trained by standard neural network methods (generally gradient descent and backpropagation [8]).

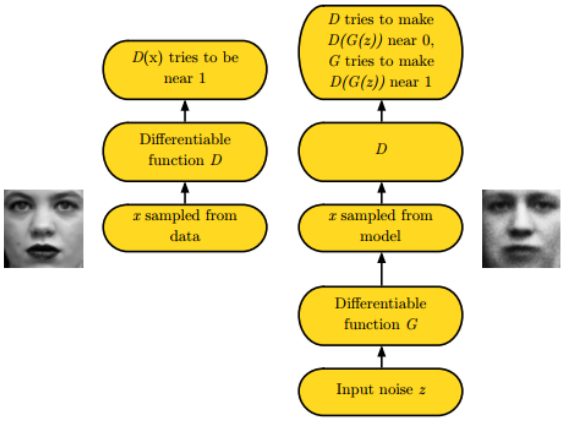

Figure 1: A graphical depiction of GAN, from Goodfellow [4], figure 12

This formulation (as we saw in lecture 12) also means that for a fixed generator D_θ, there exists an optimal discriminator [10].

Definition 6 (Optimal Discriminator)**.**

$$D^*_\theta = \frac{P_r(x)}{P_r(x) + P_\theta(x)}$$

In order to train GANs, we have many choices. We can use other standard losses [9], divergence functions [12], energy functions [16], or distance functions $d(P_r, P_\theta)$, with one possible distance function being the Wasserstein distance [1].

4 Wasserstein Distance

We have the following distance for two distributions. For any two distributions P_r, P_d.

Definition 7 (Total Variation)**.**

$$\delta(P_r, P_d) = \sup_A |P_r(x) - P_d(x)|$$

Definition 8 (KL Divergence)**.**

$$KL(P_r|P_d) = \int_x P_r(x) \log \frac{P_r(x)}{P_d(x)} dx$$

Definition 9 (Jenson-Shannon Divergence)**.** *Let P_m be the mixture of two distribution, such that $P_m = \frac{P_r+P_d}{2}$. Then*

$$JS(P_r, P_d) = \frac{1}{2} KL(P_r|P_m) + \frac{1}{2} KL(P_d|P_m)$$

Definition 10 (Wasserstein Divergence)**.** *Let $\Pi(P_r, P_d)$ be the set of all distribution whose marginals are P_r and P_d. Then*

$$W(P_r, P_d) = \inf_{\gamma \in \Pi(P_r, P_d)} \mathbb{E}_{(x,y)\sim\gamma}[||x - y||]$$

To illustrate the advantage of Wasserstein distance, we use the following toy example [7].

Example 11 (Simple Example)**.** *Considering two distributions P_θ, P_d defined over $\mathbb{R}^2$, where the true distribution is the line $(0, y)$ with y is uniformly from $U[0, 1]$. We have $P_\theta = (\theta, y)$ with y also sampled uniformly from $U[0, 1]$. We are trying to adjust θ according to different distance function.*

It can be easily shown that

$$\delta(P_d, P_\theta) = \begin{cases} 1 & \theta \neq 0 \\ 0 & \theta = 0 \end{cases}$$

$$KL(P_d|P_\theta) = P(P_\theta|P_d) = \begin{cases} +\infty & \theta \neq 0 \\ 0 & \theta = 0 \end{cases}$$

$$JS(P_d, P_\theta) = \begin{cases} \log 2 & \theta \neq 0 \\ 0 & \theta = 0 \end{cases}$$

If learning with gradient descent, then these divergence would give 0 everywhere, which explains why GAN learning would be hard using these metric. Since two distribution is just translation of each other, Wasserstein distance is

$$W(P_d, P_\theta) = |\theta|$$

5 Wasserstein GAN

Computing the Wasserstein distance is intractable in general by Definition 10, but we have the following theorem.

Theorem 12 (Kantorovich-Rubinstein duality)**.** *Wasserstein distance W is equivalent to*

$$W(P_r, P_\theta) = \sup_{||f||_L \leq 1} \mathbb{E}_{x\in P_r}[f(x)] - \mathbb{E}_{x\in P_\theta}[f(x)]$$

where the superemum is taken over all 1-Lipchitz functions.

The intuitive definition of this formulation is that every Lipschitz contiguous function has some support, but different fuctions have a different discrepancy. The supremum gives the worst case over all Lischitz functions, and by improving this value we improve the potential critic.
The Wasserstein distance is part of a large family, and is related to a number of other methods from two sample testing [13] and maximum mean discrepancy methods [14], [3]. There are also more recent methods related to the improved training of such networks [6].

References

[1] M. Arjovsky, S. Chintala, and L. Bottou. Wasserstein generative adversarial networks. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, pages 214–223, 2017. URL http://proceedings.mlr.press/v70/arjovsky17a.html.

[2] S. Boucheron, O. Bousquet, and G. Lugosi. Theory of classification: A survey of some recent advances. *ESAIM: probability and statistics*, 9:323–375, 2005.

[3] W. Bounliphone, E. Belilovsky, M. B. Blaschko, I. Antonoglou, and A. Gretton. A Test of Relative Similarity For Model Selection in Generative Models. *ArXiv e-prints*, Nov. 2015.

[4] I. Goodfellow. NIPS 2016 Tutorial: Generative Adversarial Networks. *ArXiv e-prints*, Dec. 2017.

[5] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 2672–2680. Curran Associates, Inc., 2014. URL http://papers.nips.cc/paper/5423-generative-adversarial-nets.pdf.

[6] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. Courville. Improved Training of Wasserstein GANs. *ArXiv e-prints*, Mar. 2017.

[7] A. Irpan. Overview of wasserstein gan, 2017. URL https://www.alexirpan.com/2017/02/22/wasserstein-gan.html.

[8] Y. Lecun. A theoretical framework for back-propagation. In D. Touretzky, G. Hinton, and T. Sejnowski, editors, *Proceedings of the 1988 Connectionist Models Summer School, CMU, Pittsburg, PA*, pages 21–28. Morgan Kaufmann, 1988.

[9] X. Mao, Q. Li, H. Xie, R. Y. K. Lau, Z. Wang, and S. P. Smolley. Least Squares Generative Adversarial Networks. *ArXiv e-prints*, Nov. 2016.

[10] L. Metz, B. Poole, D. Pfau, and J. Sohl-Dickstein. Unrolled Generative Adversarial Networks. *ArXiv e-prints*, Nov. 2016.

[11] K. P. Murphy. *Machine learning : a probabilistic perspective*. 2013.

[12] S. Nowozin, B. Cseke, and R. Tomioka. f-GAN: Training Generative Neural Samplers using Variational Divergence Minimization. *ArXiv e-prints*, June 2016.

[13] A. Ramdas, N. Garcia, and M. Cuturi. On Wasserstein Two Sample Testing and Related Families of Nonparametric Tests. *ArXiv e-prints*, Sept. 2015.

[14] D. J. Sutherland, H.-Y. Tung, H. Strathmann, S. De, A. Ramdas, A. Smola, and A. Gretton. Generative Models and Model Criticism via Optimized Maximum Mean Discrepancy. *ArXiv e-prints*, Nov. 2016.

[15] V. Vapnik. Principles of risk minimization for learning theory. In *Advances in neural information processing systems*, pages 831–838, 1992.

[16] J. Zhao, M. Mathieu, and Y. LeCun. Energy-based Generative Adversarial Network. *ArXiv e-prints*, Sept. 2016.