Notes on WGAN

Xavier Garcia

February 10, 2019

The purpose of these notes is to explain a few of the confusing parts of the paper "Wasserstein GAN". We begin by discussing the simplest setup for generative models, working our way to the regular GAN and finally WGAN.

1 An Overture to Approximating Distributions

We begin by establishing some notations as well as the setting for most of these notes. We assume that we have some data $x_1, ..., x_N \in \mathbb{R}^m$ and we wish to understand how it's generated. We will take a probabilistic approach, and view each x_i as samples from a random variable X, and hence the problem thus reduces to understanding the distribution of X.

1.1 Parametrized distributions and maximizing likelihood

The traditional approach to this problem is to assume that the distribution X belongs to a parametrized family of distributions $\mathbb{P}_\theta$ where $\theta \in \mathbb{R}^d$. By this, I mean that for every θ, there exists a distribution $\mathbb{P}_\theta$ and in particular, there exists a $\theta^* = \theta^*(X)$ such that

$$\mathbb{P}(X \in \cdot) = \mathbb{P}_{\theta^*}(\cdot).$$

With this assumption, the goal becomes then to find this θ^*. This assumptions also allows us to define the (log)-likelihood function:

$$\ell(\theta | x_1, ..., x_n) = \sum_{i=1}^{N} \log p_\theta(x_i) \tag{1}$$

Notice that under the assumption that X was discrete, the equation on the right (1) is exactly equal to logarithm of the probability that N independent samples of X yield the sequence $x_1, ..., x_n$. Since that statement doesn't make sense in the continuous setting, we use (1) as a proxy. A theoretically more satisfying explanation for this function follows from the fact if we had infinite data, then the following formula holds:

$$\lim_{n\to\infty} \frac{1}{N} \ell(\theta | x_1, ..., x_N) = H(X) - D_{KL}(X || \mathbb{P}_\theta)$$

where $H(X)$ is the entropy of X and D_{KL} is the Kullback-Liebler (KL) divergence. In particular, since

$$\theta^* = \text{argmin}_\theta D_{KL}(X||\mathbb{P}_\theta)$$

this implies that we can think of maximizing log-likelihood as minimizing the KL divergence between our distributions and hence yielding a tractable way to recover θ^* without having to look at distributions directly.

As a way to demystify these definitions, we provide an example. Suppose you had data $(x_1, y_1), ..., (x_N, y_N)$, where $x_i \in \mathbb{R}$ and $y_i \in \{0, 1\}$, and your goal was to understand y given x i.e. we are interested in understanding the variable $y|x$. One way to do this is by logistic regression i.e. assume that $\mathbb{P}(y = 1|x) = \sigma(wx + b)$ where σ is the logistic function and $\theta := (w, b)$ are parameters. In particular, this shows that logistic regression is actually a simple example of this framework.

1.2 On the existence of densities

The method described in the previous section is quite general, given the assumption of the existence of a density and that it belongs to a computationally tractible family of parametrized distributions or is at the very least well-approximated by one such family. In this section, we discuss how restrictive are these assumptions.

We begin with the assumption of the existence of a density. For convenience, we will call distributions which admit a density "continuous". On one hand, it's easy to think that the answer should be 'not very strong'. After all, the entirety of introductary probability theory classes and most machine learning models are based on this assumption. There are also justifiable reasons to believe that this assumption is not very strong, such as the following:

Lemma 1.1. *The set of continuous random variables is dense.*

Proof. For any random variable X and any $\epsilon > 0$, the random variable $X_\epsilon := X + \epsilon Z$ is continuous, where $Z \sim N(0, 1)$ is a standard Gaussian random variable. □

In particular, this says that even if your variable X is not continuous, there exists an an arbitrarily close approximation X_ϵ which is in fact continuous.

This lemma is very seductive. In fact, all usage of mean square error as a loss function actually depend on this fact. Nevertheless, if your data is hypersensitive to noise, then no matter how close your approximation is theoretically, in practice the data generated by the approximation will not be representative of the real distribution. For example, if you have a random variable $X \in \mathbb{R}^2$ such that it is constrained to be on the circle i.e. $||X|| = 1$, then with probability one we have $||X_\epsilon|| \neq 1$. In particular, this shows that none of these approximation would yield data that could ever actually manifest from our original random variable. The problem here is that the circle is a lower-dimensional space and

thus any amount of noise will place the variable out of this space. In particular, the injection of noise through the ϵZ term does not respect the restriction $||X|| = 1$. In real life, this phenomenon can best be seen through the production of "blurry" images.

The other problem with requiring densities is that you also need the parametrized family of densities to be amenable to some optimization schemes. Very flexible distributions such as mixtures of Gaussians will have issues when we try to maximize the likelihood and we will be forced to resort to methods such as expectation-maximization and the likes, which could pose a formidable threat when it comes to actually doing this in practice.

1.3 An alternative approach: Continuous functions of Gaussian variables

To overcome the problems from the previous subsection, we need to choose a more sophisticated approximation scheme. In particular, we need an approximation scheme which should remain dense but also be able to respect restriction to our variables. Instead of thinking of the Gaussians as noise to be added, we should instead of thinking of Gaussians as a noble distribution in its own right. In particular, we could consider continuous functions of Gaussians as our approximators. Our goal then would be to find a function f such that $f(Z) \approx X$ in distribution.

Lemma 1.2. *The set of distributions*

$$\{X : X = f(Z) \text{ where } Z \sim \mathcal{N}(0,1) \text{ and } f : \mathbb{R} \to \mathbb{R}\}$$

is dense in the space of distributions.

Proof. We give a sketch of the proof. Given a continuous random variable X with continuous cumulative distribution function (c.d.f) G, we can define $H(x) = \inf_t\{t : G(t) > x\}$ and $f : H \circ \Phi$, where Φ is the c.d.f. of Z. Then

$$\begin{aligned}
\mathbb{P}(f(X) \leq x) &= \mathbb{P}(\Phi(X) \leq G(X)) \\
&= \mathbb{P}(X \leq \Phi^{-1}G(X)) \\
&= \Phi(\Phi^{-1}G(X)) \\
&= G(X)
\end{aligned}$$

This implies we can write every continuous real-valued random variable this way and as we've seen from the previous section, those form a dense set.

□

While this proof holds only in $\mathbb{R}$, a general theorem is also true. This class of approximators is far more flexible than before, and more respectful of constraints imposed by the data. For example, we can choose functions which satisfy $||f(X)||^2 = 1$ and other constraints of the like. In particular, we've avoided the problem of requiring densities. Instead of parametrizing the distribution through the density directly, we can choose f from a parametrized family (e.g. a neural network) and obtain a family of distributions this way.

2 Wasserstein GAN

In the previous section, we outlined a different method of approximating distributions by choosing a parametrized family of functions f_θ and looking at the random variable $f_\theta(Z)$ for Z a standard Gaussian random variable. In this section, we will discuss how to train such models, as well as introduce the Wasserstein GAN.

2.1 The failure of KL Divergence

To begin training, we must develop an objective to optimize. Motivated by the methods in the previous section, we could choose the KL divergence as a loss function. In particular, we have:

$$L(\theta|X) := D_{KL}(X||f_\theta(Z)) \tag{2}$$

Symbolically, this is appealing, but mathematically, it's unclear how we could even do this. After all, not only have not assumed that X has a density, but even if we did, we don't know whether $f_\theta(X)$ has a density nor what properties would that density have, which makes defining this KL divergence seemingly impossible, even more so computing it. To overcome this, the Generative Adversarial Network (GAN)architecture was formulated by Goodfellow et. al. , which introduces the notion of a discriminator network in order to make Equation (2) make sense. We will not delve into that framework in these notes. Instead, we invite the reader to question whether such a methodology even makes sense. After all, the goal itself is unsound. The sole reason why we chose to consider functions of Gaussian random variables as opposed to parametrizing the density was to avoid having a density. Since the KL divergence necessitates densities, any attempts at mimicking the KL divergence seem like they are missing the point.

From a mathematical standpoint, the KL divergence has many defects which make it troubling. For starters, the KL divergence is not a metric, since it's not symmetric. Second of all, it will diverge for distributions with disjoint support. While the surrogate loss function introduced in the original GAN framework manages to fix both of these issues, it did not fix the biggest theoretical issue: as a function of θ, the function

$$\theta \mapsto D_{KL}(X||f_\theta(Z))$$

could fail to be continuous. This is extremely troubling, since it suggests that even in the optimal case where we have infintie data, our loss function proves to be ill-defined.

2.2 The Wasserstein distance

All of these problems revolve around the necessity of a density to have a KL divergence. Since our goal was to remove this necessity, we should seek out a

different metric that respects this goal. A careful study of probability theory reveals that there are actually a variety of metrics for distributions. The one we will focus on for the purposes of these notes is the Wasserstein distance $\mathcal{W}(\cdot, \cdot)$, also known as the Earth Mover (EM) distance. To define this, we need a new more definitions. Given two random variables X and Y, we call $Z = (Z_X, Z_Y)$ a **coupling** of X and Y if Z_X has the same distribution as X and Z_Y has the same distribution as Y. Trivially, $Z = (X, Y)$ is an example of a coupling, but it is not the only one. For example, if X and Y have the same distribution and are indepenent, then (X, Y), (X, X), and (Y, Y) are all examples of couplings but all different random variables and not all sharing the same joint distribution. For two random variables X and Y, we let $\Gamma(X, Y)$ denote the space of couplings of X and Y. With this definitions in hand, we can define the Wasserstein distance $\mathcal{W}(X, Y)$ by the equation:

$$\mathcal{W}(X, Y) := \inf_{Z \in \Gamma(X,Y)} \mathbb{E}[||Z_X - Z_Y||].$$

Let's take a moment to make a few observations. First, it's quite possible that this infimum is not actually attained for any one coupling, which is why we use the word 'infimum' as opposed to 'minimum'. Second, notice that this definition makes no assumption on a density at all, but this comes at the cost of having to take the infimum over the space of couplings, which could be quite complicated. Third, we also note this metric also arises naturally as the weak* topology of the space of probability distributions, but we won't delve into this remark any further. Finally, and most importantly, if for a fixed z, $\theta \mapsto f_\theta(z)$ is locally Lipschitz, then

$$\theta \mapsto \mathcal{W}(X, f_\theta(Z))$$

is continuous everywhere and differentiable almost everywhere.

2.3 Computing the Wasserstein metric and its derivative

Although the Wasserstein distance has desirable theoretically properties, computing seems intractable. Even if we knew the distribution of X, computing the Wasserstein distance could be quite difficult. Even more troubling, we can't simply approximate through a Monte Carlo scheme using data as we normally do due to the infimum computation. Therefore, we need to find a different representation of the Wasserstein distance that's more amenable to approximation. To do this, we invoke the Kantorovich-Rubenstein duality that allows us to write the following equivalent formulation for the Wasserstein distance:

$$\mathcal{W}(X, Y) = \sup_{g: ||g||_L \leq 1} \mathbb{E}[g(X)] - \mathbb{E}[g(Y)]$$

where the supremum is over Lipschitz funtions g with Lipschitz constant $||g||_L$ bounded from above by 1. This representation is more amenable to approximation because now we can replace the supremum overall Lipschitz functions to a supremum over functions g_ϕ, parametrized by some parameters ϕ which satisfy $||g_\phi||_L < K$ for some K.

We now proceed to start making assumptions to make this problem tractable. First, we assume that there exists an optimal ϕ^* such that

$$\begin{aligned}\mathcal{W}(X,Y) &= \sup_{g:||g||_L<1} \mathbb{E}[g(X)] - E[g(Y)] \\ &= \mathbb{E}[g_{\phi^*}(X)] - \mathbb{E}[g_{\phi^*}(Y)].\end{aligned}$$

With such an assumption, we can then easily compute the following gradient:

$$\nabla_\theta W(X, f_\theta(Z)) = -\mathbb{E}[\nabla_\theta g_{\phi^*}(f_\theta(Z))]$$

This expression is now amenable to the usual Monte Carlo scheme i.e. given samples $z_1, ..., z_m \sim Z$, we can write

$$\nabla_\theta W(X, f_\theta(Z)) \approx -\frac{1}{m}\sum_{i=1}^{m} \nabla_\theta g_{\phi^*}(f_\theta(z_i))$$

and similarly for the Wasserstein metric.

Finally, we look at how limiting is the constraint that the family of g_ϕ have Lipschitz constant bounded by one. We can relax this condition by considering Lipschitz function bounded by some fixed K, then later dividing by K. It's an instructive exercise to convince one's self that feedfoward neural networks are also Lipschitz functions, whose Lipschitz constant be bounded above by a continuous function of its parameters. This implies that if we restrict the parameters ϕ to lie in a compact set, then we can find a global Lipschitz bound that does not depend on the parameters.

With all these steps defined, we can explain the algorithm in simple words. We begin with some samples $x_1, ..., x_m$ and intial parameters ϕ and θ. We then sample $z_1, ..., z_m \sim Z$ and compute $f_\theta(z_i)$. Next, we try to approximate the Wasserstein metric by using the performing gradient descent on the ϕ variable for the loss function

$$L_{\text{WGAN}}(\phi, \theta) = \frac{1}{m}\sum_{i=1}^{m} g_\phi(x_i) - g_\phi(f_\theta(x_i)).$$

We can either perform the full gradient descent or take a few steps. Either way, we update our ϕ. To ensure out ϕ remain in a compact set, we clip the weights so that $\phi_i \in (-0.01, 0.01)$ for each i. Then, we try to minimize the approximated Wasserstein distance by minimizing the same loss function, but this time as a function of θ.