Wasserstein Generative Adversarial Networks

Summary Notes by Max Guo

July 7, 2022

(?) - denotes a lack of familiarity or understanding of a particular concept at time of reading

(?) - denotes a confusion as to why the authors included this

1 Information

- **Year**: 2017
- **Conference**: ICML
- **Authors**:

Name	Institute
Martin Arjovsky	Courant Institute of Mathematics
Soumith Chintala	Facebook AI Research
Leon Bottou	Courant Institute of Mathematics, Facebook AI Research

2 Research Problem

How do we learn a probability distribution and improve upon the shortcomings of regular GANs?

3 Existing Approaches and Shortcomings

Existing approaches to learning a probability distribution:

- **Learning a Probability Density**:
  - Define a parametric family of densities $(P_\theta)_{\theta \in \mathbb{R}^d}$, and find the θ that maximizes the likelihood of the data. Equivalent to minimizing the KL divergence $KL(P_r||P_\theta)$, where P_r is the real data distribution.
    * **Problem**: P_θ might be 0 in some places where $P_r > 0$, so KL is not defined.
    * **Remedy**: Add a noise term to the model distribution
    * **Problem with Remedy**: Noise degrades quality of samples; need a high amount of noise.
  - Define a random variable Z with fixed distribution $p(z)$ and obtain P_θ via passing z through $g_\theta : \mathcal{Z} \to \mathcal{X}$. Can vary θ to obtain distribution close to the real data distribution P_r.
    * **Benefits**:
      · Enables representations of distributions on low-dimensional manifolds
      · Generating samples is better than knowing density value (generally, sampling from arbitrary high-dimensional density is hard)
    * Examples: VAEs (Kingma and Welling, 2013) and GANs (Goodfellow et al., 2014)
      · VAEs need to fiddle with additional noise terms (approximate likelihood of examples)
      · Training GANs is "delicate and unstable"

4 High Level Contribution

The paper analyzes how the **Earth Mover (Wasserstein-1) distance** compares theoretically to other popular probability distances and defines the **Wasserstein-GAN** to minimize an approximation of the EM distance. They show that the Wasserstein-GAN **remedies some GAN training problems**, including: mode dropping, balancing training of generator and discriminator, and requiring careful design of network architecture.

5 Technical Contributions

5.1 Distances between Probability Distributions

- **Total Variation** distance:
$$\delta(\mathbb{P}_r, \mathbb{P}_g) = \sup_{A \in \Sigma} |\mathbb{P}_r(A) - \mathbb{P}_g(A)| \tag{1}$$
- **Kullback-Leibler** divergence:
$$KL(\mathbb{P}_r || \mathbb{P}_g) = \int P_r(x) \log \left(\frac{P_r(x)}{P_g(x)} \right) \mathrm{d}\mu(x) \tag{2}$$
- **Jensen-Shannon** divergence:
$$JS(\mathbb{P}_r, \mathbb{P}_g) = KL(\mathbb{P}_r || \mathbb{P}_m) + KL(\mathbb{P}_g || \mathbb{P}_m) \tag{3}$$
where $\mathbb{P}_m = (\mathbb{P}_r + \mathbb{P}_g)/2$.
- **Earth-Mover** (Wasserstein-1) distance:
$$W(\mathbb{P}_r, \mathbb{P}_g) = \inf_{\gamma \in \Pi(\mathbb{P}_r, \mathbb{P}_g)} \mathbb{E}_{(x,y)}[||x - y||] \tag{4}$$
$\Pi(\mathbb{P}_r, \mathbb{P}_g)$ is the set of joint distributions whose marginals are $\mathbb{P}_r$ and $\mathbb{P}_g$.

5.2 Theoretical Results

- **Theorem 1**:

  Let $\mathbb{P}_r$ be a fixed distribution over set $\mathcal{X}$. Let Z be a random variable over another space $\mathcal{Z}$. Let $\mathbb{P}_\theta$ denote the distribution of $g_\theta(Z)$, $g : (z, \theta) \in \mathcal{Z} \times \mathbb{R}^d \mapsto g_\theta(z) \in \mathcal{X}$. Then:

  1. If g is continuous in θ, $W(\mathbb{P}_r, \mathbb{P}_\theta)$ is continuous in θ.
  2. If g is locally Lipschitz and satisfies regularity conditions, then $W(\mathbb{P}_r, \mathbb{P}_\theta)$ is continuous everywhere and differentiable almost everywhere.
  3. The above are false for JS divergence and forward or reverse KL.

- As a corollary, if g is a feedforward NN, then $W(\mathbb{P}_r, \mathbb{P}_g)$ is continuous everywhere and differentiable almost everywhere.
- **Theorem 2** demonstrates that the order of strength of the distances and divergences is KL, JS and TV, and EM is the weakest.
  - For example, if a sequence of probabilities and another fixed probability distribution are measured by KL to have divergence going to 0, JS and TV, and EM will also have the same result.
  - However, there exists (simple) examples where EM goes to 0 but the others don't.
  - $\implies$ when learning distribution on low dimensional manifolds, shouldn't use KL, JS, or TV.

5.3 More on Wasserstein Distance

- Calculating the Wasserstein Distance via the definition above is intractable, so instead use:

$$W(\mathbb{P}_r, \mathbb{P}_\theta) = \sup_{||f||_L \leq 1} \mathbb{E}_{x \sim \mathbb{P}_r}[f(x)] - \mathbb{E}_{x \sim \mathbb{P}_\theta}[f(x)] \tag{5}$$

- Another theorem showing that htere is a solution to this equation, and that:

$$\nabla_\theta W(\mathbb{P}_r, \mathbb{P}_\theta) = -\mathbb{E}_{z \sim p(z)}[\nabla_\theta f(g_\theta(z))] \tag{6}$$

6 Wasserstein GAN

Algorithm 1 WGAN, our proposed algorithm. All experiments in the paper used the default values $\alpha = 0.00005$, $c = 0.01$, $m = 64$, $n_{\text{critic}} = 5$.

Require: : α, the learning rate. c, the clipping parameter. m, the batch size. n_{critic}, the number of iterations of the critic per generator iteration.
Require: : w_0, initial critic parameters. θ_0, initial generator's parameters.

```
while θ has not converged do
    for t = 0, ..., n_critic do
        Sample {x^(i)}_{i=1}^m ~ P_r a batch from the real data.
        Sample {z^(i)}_{i=1}^m ~ p(z) a batch of priors.
        g_w ← ∇_w[ (1/m) Σ_{i=1}^m f_w(x^(i))
                     − (1/m) Σ_{i=1}^m f_w(g_θ(z^(i))) ]
        w ← w + α · RMSProp(w, g_w)
        w ← clip(w, −c, c)
    end for
    Sample {z^(i)}_{i=1}^m ~ p(z) a batch of prior samples.
    g_θ ← −∇_θ (1/m) Σ_{i=1}^m f_w(g_θ(z^(i)))
    θ ← θ − α · RMSProp(θ, g_θ)
end while
```

Notes:

- Train critic (discriminator) n_{critic} times, train generator once
- In order to have parameters w lie in a compact space (for the Lipschitz condition), clamp weights to a fixed box ($\mathcal{W} = [-0.01, 0.01]^l$).
- Because EM distance is continuous and differentiable, should train the critic to optimality.
- JS (normal GAN) results in vanishing gradients and mode collapse
- EM results in clean gradients everywhere and no mode collapse.

7 Empirical Results

- **Datasets**
  - Mixture of Gaussians
  - Image generation (LSUN-Bedrooms dataset)
- **Baselines**
  - DCGAN (GAN with convolutional architecture) (Radford et al., 2015) trained with standard GAN procedure
- **Benefits**
  - Meaningful loss metric (estimate of the EM distance) that correlates with the generated sample quality. Estimate goes down with higher sample quality.
    - This is not empirically true for JS, the baseline.
  - Improved stability - more robust to the architecture of the generator.
- **Observations**
  - WGAN is unstable with Adam or momentum, so used RMSProp.

8 My Questions and Thoughts

- This paper is quite mathematical in the theory portions, and I would probably need to take deeper mathematical or statistical courses to fully understand the supplementary proofs.
- A YouTube video tutorial on WGANs was very helpful for my understanding of this paper. There was also an implementation of WGANs using PyTorch on the video.
- The clipping of the weights was mentioned in the video as a crude way of enforcing the Lipschitz condition, and there are other papers describing how to do this in more principled ways.