DOT Applied to WGAN

Author:

Paul Arrault
Dual Degree in Artificial Intelligence and Organizational Sciences (IASO)

Supervisor:

Professor Alexandre Vérine

Report Outline:

December 2024

1. Abstract

GANs are highly significant for image generation. As part of a project in the Deep Learning 2 course, we aimed to improve a standard GAN to achieve the best possible results. We will test this on MNIST. Initially, I wanted to apply the Discriminator Optimal Transport (DOT) ([1]), a method that I eventually decided to support with a WGAN. We will first test a WGAN and understand why it should be applied, and then incorporate DOT.

2. Introduction

2.1. Key Concepts

Wasserstein Distance The Wasserstein distance is a metric that quantifies the minimal cost required to transform one distribution P into another Q. It considers the full geometry of the data by examining the distributions as a whole. An important characteristic for this study is that it remains well-defined even when P and Q have no common support. It is defined as:

$$W(P,Q) = \inf_{\gamma \in \Pi(P,Q)} \int_{\mathcal{X}\times\mathcal{Y}} ||x-y|| d\gamma(x,y),$$

where $\Pi(P,Q)$ represents the set of couplings between P, Q. It can also be expressed as an expectation, which is sometimes easier to interpret: $W(P,Q) = \mathbb{E}_{x\sim P}\left[\min_{y\sim Q} ||x-y||\right]$.

Lipschitz Norm The Lipschitz norm measures the maximum rate of change of a function f. For a GAN, an excessively high or low Lipschitz norm often indicates instability. This will be **highly useful** for the two methods we will use, and we will see how to compute it in practice. Mathematically, it is given by:

$$||f||_{\text{Lip}} = \sup_{x,y} \frac{|f(x)-f(y)|}{||x-y||}.$$

2.2. Limitations of Standard GANs

Before discussing the limitations of standard GANs, it is important to recall that their loss function optimization relies on the joint optimization of the generator G and the discriminator D:

$$\min_G \max_D \Big(\mathbb{E}_{x\sim P_{\text{data}}}[\log D(x)] + \mathbb{E}_{z\sim P_z}[\log(1-D(G(z)))]\Big).$$

Jensen-Shannon Divergence (JS) The JS divergence, used to compare distributions (similarly to the Wasserstein distance), is defined as:

$$JS(P||Q) = \frac{1}{2}KL(P||M) + \frac{1}{2}KL(Q||M), \quad \text{where } M = \frac{1}{2}(P+Q), \quad \text{and} \quad KL(P||Q) = \int P(x)\log\left(\frac{P(x)}{Q(x)}\right)dx.$$

When P and Q have no common support, $JS(P||Q)$ becomes ineffective because $KL(P||M)$ diverges. This results in an almost perfect discriminator, leading to vanishing gradients or mode collapse.

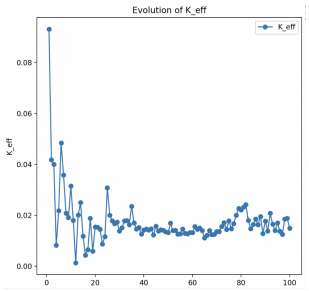
Vanishing Gradient When $D(G(z)) \approx 1$, the gradient for the generator becomes very small. The gradient no longer provides sufficient information, preventing the generator from improving effectively, which leads to learning stagnation:

$$\nabla_G V(G,D) = \nabla_G \mathbb{E}_{z\sim P_z}[\log(1-D(G(z)))] \approx 0.$$

Mode Collapse To counteract the discriminator becoming too strong, the generator specializes in producing a limited subset of the target distribution, sacrificing diversity. In fact, the generator seeks to locally minimize its loss, mathematically: $P_G(x) = \sum_{i=1}^{k} \delta(x - \mu_i)$, where μ_i represents the modes learned by the generator.

Training Instability Achieving balance between the generator G and discriminator D is difficult. Therefore, it may be beneficial to explore alternative methods.

Difficulty in Calculating the Lipschitz Norm The Lipschitz norm is crucial to ensuring stable gradients and thus stable training. However, due to training instability, it quickly becomes unusable. Nevertheless, it is **essential** for the DOT method. Therefore, I opted for a more stable model: the WGAN.



3. Wasserstein GANs

3.1. Model

Wasserstein GANs ([2]) (WGANs) introduce a new loss function formulation to address the issues of standard GANs. Their primary goal is to ensure stable training by replacing the JS divergence with another measure. The Wasserstein distance was chosen; since its basic form is "rather hostile," a good method is to modify it using the Kantorovich-Rubinstein theorem as:

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

Thus, we aim to approximate the function $f(x)$ with $D_\theta(x)$. The generator L_G and discriminator L_D losses are now reformulated as follows:

$$L_G = -\mathbb{E}_{x \sim Q}[D(x)], \quad L_D = \mathbb{E}_{x \sim Q}[D(x)] - \mathbb{E}_{x \sim P}[D(x)].$$

3.1.1. Enforcing the Lipschitz Constraint

The discriminator D must comply with the Lipschitz constraint, ensuring it is less than or equal to 1. Several methods exist for this, with the main ones being gradient clipping, gradient penalty, and spectral normalization.

I initially opted for **spectral normalization**. For a linear transformation $l_{W,b}$, $||l_{W,b}||_{Lip} = \sigma(W)$, where $\sigma(W)$ is the largest singular value of W. Spectral normalization SN is defined as: $SN(l_{W,b}) = l_{\frac{W}{\sigma(W)},b}$, ensuring $||l_{W/\sigma(W)}||_{Lip} = 1$.

In the context of neural networks, this normalization guarantees that the network's Lipschitz norm is bounded due to this property:

$$||f \circ g||_{Lip} \leq ||f||_{Lip} \cdot ||g||_{Lip}.$$

Applied to a neural network with activations whose norms are less than 1 and spectral normalization in each layer, we get:

$$\|f_{\text{nn}}\|_{\text{Lip}} \leq \prod_{l=1}^{L} \|l_{W_l/\sigma(W_l)}\|_{\text{Lip}} = 1$$

Thus, this normalization ensures the network adheres to the Lipschitz constraint, stabilizing GAN training (figure 1). However, this method does not provide control over its value, which can slow training with a norm close to 0 (figure 2).

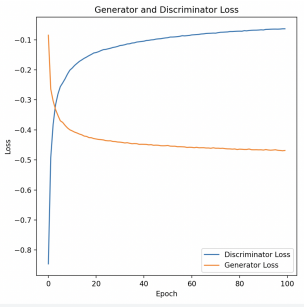


Figure 1

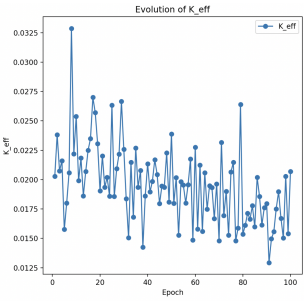


Figure 2

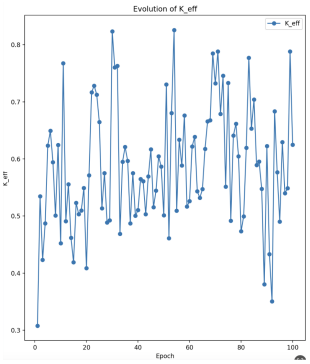


Figure 3

3.1.2. Choosing Gradient Penalty

An alternative is the use of Gradient Penalty (GP). This method not only enforces the constraint but ensures it explicitly converges toward 1. I chose this method because **having a spectral norm equal to 1** is crucial for optimal transport.

The new loss function for the discriminator then becomes:

$$L_D = \mathbb{E}_{x\sim Q}[D(x)] - \mathbb{E}_{x\sim P}[D(x)] + \lambda \cdot \mathbb{E}_{\hat{x}\sim P_{\hat{x}}}\Big(\big(\|\nabla_{\hat{x}} D(\hat{x})\|_2 - 1\big)^2\Big),$$

where $P_{\hat{x}}$ is a distribution linearly interpolating between real and generated data. This method has several advantages:

- Smoother than gradient clipping, avoiding the instability issues observed with the latter.
- Ensures a norm close to 1 for the discriminator, which is particularly useful for DOT! (see section 4)

In practice, while variations around the target norm are expected, they remain an empirical estimate of the discriminator's overall behavior (figure 3).

3.2. Results

The model showed promising results. Training was very stable, with no signs of *mode collapse*, and it produced significant diversity in the generated digits. The evolution during training is illustrated below. Each digit evolves slowly but converges to a reliable output.

Training Parameters: D vs G ratio: 5 (first 50 epochs), then 1; Epochs: 204; Learning rate: 10^{-4}; Optimizer: Adam ($\beta_1 = 0.0$, $\beta_2 = 0.9$); Architecture: WGANGP.

4. Discriminator Optimal Transport

DOT is based on the following premise: in an autoencoder, the encoder and decoder are used jointly, so why not exploit D in post-training? DOT aims to enhance the quality of generated images and correct errors. Optimal transport uses the Wasserstein distance (hence my choice of WGANGP).

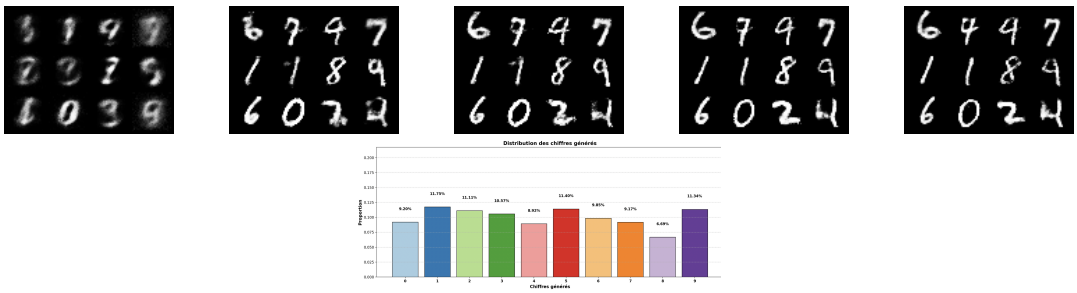


Figure 4: Evolution during training and diversity of generated digits.

4.1. Theory

The method is based on the following two theorems:

Theorem 1:
Assumptions: π^* and D^* are optimal solutions, π^* is an optimal deterministic transport described by $T : X \to X$.
Results:
$\|D^*\|_{\text{Lip}} = 1$,
$T(y) = \arg\min_x\{\|x - y\|_2 - D^*(x)\}$,
$p(x) = \int dy\, \delta(x - T(y))q(y)$

Theorem 2:
Assumptions: Every GAN objective function with a gradient penalty provides a lower bound for the average divergence between p and p_G. $\tilde{D} = D/K$, where K is the Lipschitz coefficient of D.
Results:
$V_D(G, D) \leq K\left(\mathbb{E}_{x\sim p}[\tilde{D}(x)] - \mathbb{E}_{y\sim p_G}[\tilde{D}(y)]\right)$

These two theorems highlight the link between optimal transport and GAN discriminators (1). They show that maximizing the discriminator's loss is equivalent to minimizing the Wasserstein distance and that this method works for any GAN (2). This explains why it was crucial for the Lipschitz coefficient to converge to 1 and why its precise calculation is so important.

Calculating the Lipschitz Coefficient: The Lipschitz norm is calculated using two formulas:

$$K_{\text{eff}} = \max\left\{\frac{|D(x) - D(y)|}{\|x - y\|_2} \mid x, y \sim p_G\right\}$$

(image space) and

$$k_{\text{eff}} = \max\left\{\frac{|D \circ G(z) - D \circ G(z')|}{\|z - z'\|_2} \mid z, z' \sim p_Z\right\}$$

(latent space). For my experiments, I used 50,000 pairs, repeated the calculation 10 times, and averaged the maximum values to avoid outliers.

It is particularly important to note that for the ideal version, we have:

$$T_D(y) = \arg\min_x\{\|x - y\|_2^2 - \frac{1}{K}D(x)\}.$$

4.2. Target Optimal Transport

Optimal transport in image space, defined as

$$T_D^{\text{eff}}(y) = \arg\min_x\left\{\|x - y\|_2^2 - \frac{1}{K_{\text{eff}}}D(x)\right\},$$

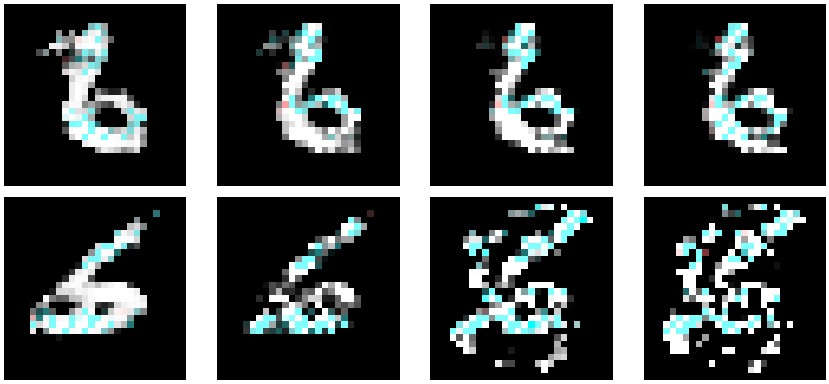
optimizes the alignment between generated samples and the target distribution.

Algorithm 1 Optimal Transport in Image Space (TOT)

Require: trained D, coefficient K_{eff}, sample y, learning rate ϵ, small vector δ.

Initialize $x \leftarrow y$.
for $n_{\text{trial}} = 1, \ldots, N_{\text{updates}}$ **do**
  Update $x \leftarrow x - \epsilon \nabla_x \left(\|x - y\|_2^2 + \delta - \frac{1}{K_{\text{eff}}} D(x) \right)$.
end for
return x.

Results: Minimal improvements (*first row of the representation read from left to right*). It works for small dimensions but is not suited for our case (dimension 784). The method is very unstable; increasing the number of iterations by 2 and the learning rate by 0.05 produces unreadable noise (*second row*).



4.3. Latent Space DOT

Optimal transport in latent space, defined as

$$T^{\text{eff}}_{D \circ G}(z_y) = \arg\min_z \left\{ \|z - z_y\|_2^2 - \frac{1}{k_{\text{eff}}} D \circ G(z) \right\},$$

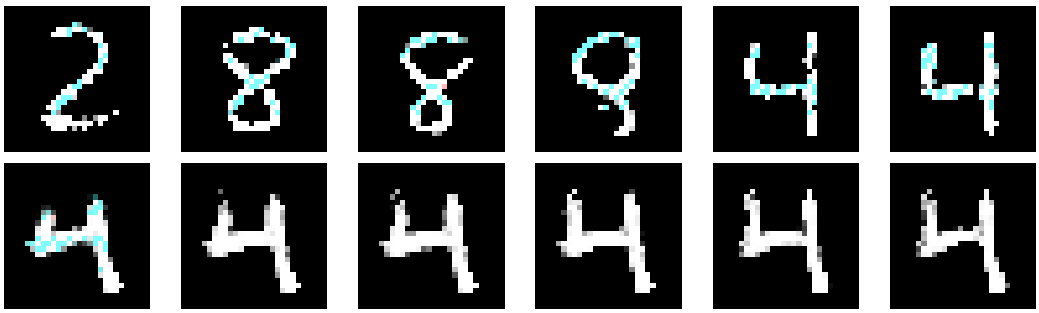
aims to adjust latent representations to improve the quality of generated samples.

Algorithm 2 Optimal Transport in Latent Space (Latent Space DOT)

Require: Generator G, Discriminator D, coefficient k_{eff}, sample z_y, learning rate ϵ, small vector δ.

Initialize $z \leftarrow z_y$.
for $n_{\text{trial}} = 1, \ldots, N_{\text{updates}}$ **do**
  $g \leftarrow \nabla_z \left\{ \|z - z_y\|_2^2 + \delta - \frac{1}{k_{\text{eff}}} D \circ G(z) \right\}$.
  if noise is generated by $\mathcal{N}(0, I_{D \times D})$ **then**
    $g \leftarrow g - (g \cdot z) z / \|D\|$
  end if
  Update $z \leftarrow z - \epsilon g$.
  if noise is generated by $\mathcal{U}([-1, 1])$ **then**
    Clip $z \in [-1, 1]$.
  end if
end for
return $x = G(z)$.

Results: Precision is improved. With a high learning rate, diversity decreases but precision increases (*first row*). The goal is to find a balance to maintain diversity while improving precision (*second row*).



4.4. Comparison

I opted for Latent Space DOT. The goal was not hyperparameter tuning but understanding the consequences of modifications. Increasing learning rates improved precision (Figure 2) but reduced diversity. By finding a balance, it was possible to minimize diversity loss while retaining precision (Figure 3). Ultimately, the *goal was achieved*, with precision improved over WGANGP, which served as our benchmark (Figure 1). For a detailed exploration of latent space shifts and trade-offs between precision and recall, visit my Canva.

Recall that precision and recall are defined as:

$$\bar{\alpha} = \hat{P}(\text{Supp}(P)) \quad \text{and} \quad \bar{\beta} = P(\text{Supp}(\hat{P})).$$

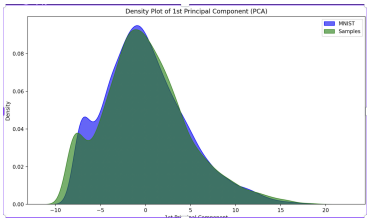


Figure 1: WGANGP

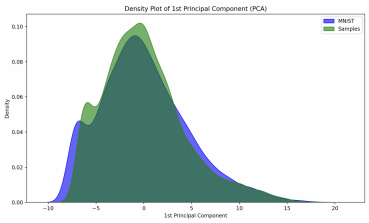


Figure 2: Learning rate = 0.5

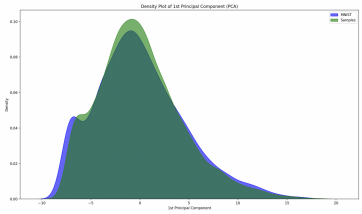
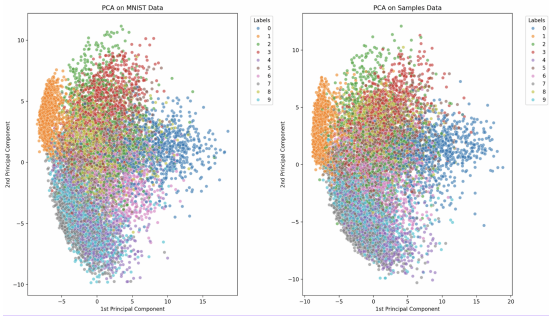
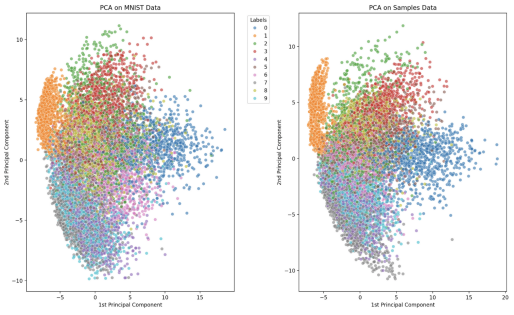


Figure 3: Learning rate = 0.005

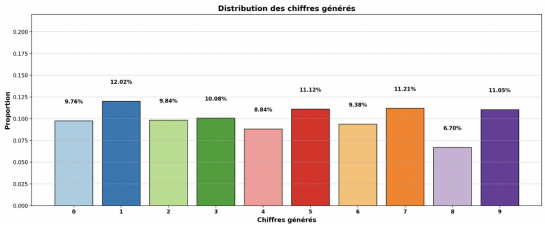
I didn't focus on hyperparameter tuning, but my best result was achieved with a learning rate of 0.05, a Lipschitz norm of 0.47, 25 updates, and $\delta = 10^{-3}$. Visualizations of latent spaces for generated data compared to MNIST are shown below, along with a graph on sample diversity. Metrics: FID: 19.65, precision = 0.52, recall = 0.32.



WGANGP



WGANGP + DOT



Graph: Diversity of Generated Samples

5. Discussion and Perspectives

The results obtained are generally satisfactory. However, several avenues for improvement can be explored. First, fine-tuning, particularly of the learning rate, the Lipschitz norm, and the number of updates, could enhance both the stability and precision of the results. The calculation of the Lipschitz norm could also be improved; this computation must be carefully designed in its execution to ensure the implementation of DOT.

Next, the trade-off between precision and recall remains a major challenge. Finding a balance between these two objectives is critical, and fine-tuning is again a potential solution. Despite the encouraging results, some artifacts persist in the generated images, indicating residual instabilities (a consequence of model errors, as DOT assumes the discriminator is "*perfect*") or poor convergence in latent space.

Additionally, better normalization of latent vectors could improve the quality of generated samples by ensuring a more homogeneous distribution of representations in latent space. While the method was successfully applied to WGANGP, it would be interesting to evaluate it on other GAN variants, such as StyleGANs, BigGANs, or FGANs, to test its generalizability and adaptability.

Finally, DOT, thanks to its simplicity and post-training effectiveness, constitutes a fast and intuitive tool for visualizing changes in latent space and generated images.

References

- [1] DOT: Gabriel Peyré, Marco Cuturi, et al. "Computational Optimal Transport." https://arxiv.org/abs/1910.06832
- [2] WGAN: Martin Arjovsky, Soumith Chintala, Léon Bottou. "Wasserstein GAN." https://arxiv.org/abs/1701.07875