Optimal Transport applied to Generative Adversarial Networks
Final Project Report - Deep Learning
MVA, 2020

DÉMARRE Camille
École Polytechnique
camille.demarre@polytechnique.edu

NAIT SAADA Thiziri
Télécom Paris
thiziri.naitsaada@telecom-paris.fr

Abstract

Lack of training data is a crucial challenge in fields such as medicine or genetics where experiments to generate them can have a high cost. To deal with this issue, Generative Adversarial Networks are powerful tools to generate samples following a given distribution. Since their introduction by Goodfellow et al. *[4], many improvements have been proposed in order to correct some instabilities in training and sample quality. Wasserstein GANs belong to these proposed advances and rely on the Optimal Transport theory to give some guarantees on the effectiveness of its proposed method. It is interesting to notice that this approach has popularized the use of Optimal Transport theoretical tools and applications to modern Machine Learning issues. In this report, we will study the theoretical Optimal transport background leading to the formulation of the WGAN objective and the comparisons that can be drawn between standard GANs and Wasserstein GANs. In particular, we will focus on the theoretical properties of the Wasserstein distance that enabled us to derive its second formuation based on the so called Kantorovich-Rubinstein duality theorem. We also study more specifically the limits of WGANs. In order to see how these models behave, we decided to re-implement the different algorithms - with our limited resources - in order to see if the results of the different papers can be replicated. We derived two versions of this approach: WGAN with weight clipping and with gradient penalty, the second one being slightly better than the one suggested in the paper, regarding stability and performances of the model.*

1. Introduction

Generative Adversarial Networks (GANs) are a famous class of generative models with a seemingly easy-to-explain structure. It can indeed be seen as a game between two players : the generator and the discriminator. The goal of the generator is to generate fake images resembling images from a dataset, while the goal of the discriminator is to distinguish fake images from true images.

The recent surge of interest in Generative Models and GANs has given rise to many applications, one of the most well-known being DeepFake images, which gives rise to ethical preoccupations. However, Generative Models cannot be summed up to a toy problem or a negative-impact instances: for instance, their power is now being increasingly leveraged in molecular biology, with applications such as the design of new molecules (as in [6]). They can also serve Deep learning models used to infer the energy free landscape of molecules for instance, by generating samples from any underlying true distribution and thus addressing the issue of lack of sample data [14].

1.1. Motivation

GANs have thus proven to be a promising way to generate images from a distribution close to the original dataset distribution, thanks to the training of the generator. After training, and given a latent variables Z, the generator is a parameterized function f_θ such that $f_\theta(Z)$ is close to the true underlying distribution. However, the training phase of GANS is known to be unstable and the range of generated samples are known to lose diversity (this diversity loss is referred to as mode collapse and its reason is studied in [1]).

The design of the Wasserstain GAN (WGAN) by Arjovsky *et al.* [2] aims to study those two limitations, and the lens of Optimal Transport theory can prove useful to alleviate unstability and reduce mode collapse.

1.2. Related work

The founding paper of GANs was written by Goodfellow *et al.* [4] in 2014. It introduces the main structure, demonstrates the effectiveness of GAN on the MNIST, TFD, and CIFAR-10 image datasets. Since then, many architectures and improvements have been proposed to apply this generative algorithm's idea to different datasets and applications. The paper on DCGAN by Radford *et al.* [12] proposes to build a deep convolutional network with batch normalization for the generator and discriminator, thus stabilizing training and allowing higher resolution images. Current state of the art GANs now include, among others, BigGAN for ImageNet ([3]), StyleGAN ([7]) and CycleGAN ([16]).

Optimal transport was shown recently renewed interest (see the works by Villani in 2009 and the works by Peyré and Cuturi in 2013 [15, 11]), yielding efficient algorithms for a wide variety of applications, ranging from brain decoding to image editing or image super-resolution, or - for our problem - in GANs. With the design of a new type of GAN, Arjovsky *et al.* [2] opened a new area of research that has proven fruitful: [5, 1] show some considerations on the training of such neural networks, [9] proposes to use a quadratic transport cost, and to regularize the training.

1.3. Problem definition

The aim of this presentation is to study what Optimal transport can bring to GANs by comparing the two proposed models, the original GAN by Goodfellow *et al.* [4], and the WGAN by Arjovsky *et al.* [2]. Since the paper from Arjovsky *et al.*, some improvements have been introduced, and amongst them the so-called WGAN with Gradient Penalty, which we will study as well, both in terms of theory and implementation-wise.

In section 2, we will present the theoretical foundations leading to the introduction of the 1-Wasserstein distance in WGAN, and will discuss the properties ensuing. We will present our methodology and general implementation. Section 3 presents our numerical and visual results on two datasets: MNIST and CIFAR10.

2. Main body

2.1. Theoretical presentation

We will present the equation describing the original GAN and then introduce the equation describing the WGAN. To do so, we will state the Kantorovich-Rubinstein duality.

2.1.1 Equations from the GAN

Generative Adversarial Models (GAN) do not aim to explicitly model the density explaining the data: they rather aim to generate new instances that are close to real ones. They are based on an adversarial competition between a discriminator and a generator. During the training, each structure has its own objective. The generator tries to fool the discriminator by generating fake samples that get closer and closer to real data (starting from random noise) while the discriminator intends to distinguish fake from real data. In the end of the training, if everything is successful, the discriminator is not able to identify real from fake instances: this means that the generator can now generate samples that are close to the true distribution of the real data. As such, the generator can then be used to sample new unknown data.

Property 2.1 *GAN's objective*

$$\min_G \max_D \mathbb{E}_{x\sim p} \log D(x) - \mathbb{E}_{z\sim\mathcal{N}(0,\mathcal{I})} \log(1 - D(G(z))$$

Instead of this optimization problem - which computation requires a high computational cost - approximations with finite sums are preferred, and a GAN's approximate objective can thus be written:

Property 2.2 *GAN's approximate objective*

$$\min_G \max_D \frac{1}{|\tau|} \sum_{x\in\tau} \log D(x) + \frac{1}{N_z} \sum_{z\sim\mathcal{N}(0,\mathcal{I})} \log(1 - D(G(z)))$$

Figure 1 gives a schematic summary of a GAN's architecture.

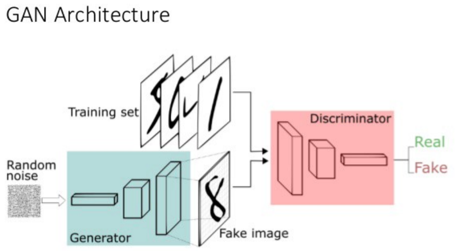


Figure 1: Stucture of GANs decomposed between a discriminator and a generator

2.1.2 Equations from the WGAN

Definition 2.1 *p-Wasserstein Distance between two positive measures with same mass*

$$\begin{aligned} \mathcal{W}_p^p(\alpha,\beta) := & \quad |\alpha| \min_{\pi\in\Pi(\alpha,\beta)} \int_{\mathbb{R}^d\times\mathbb{R}^d} |x-y|^p d\pi(x,y) \\ = & \quad |\alpha| \min_{\pi\in\Pi(\alpha,\beta)} \mathbb{E}_{(x,y)\sim\pi}(||x-y||^p) \end{aligned} \tag{1}$$

Recall that we seek to minimize the 1-Wasserstein distance between two measures of probability. The first one is the true distribution of the data p, while the second one is the approximate distribution p_θ inferred from the training data that is parametrized with a parameter $\theta \in \Theta$. Thus, the objective can be can be rewritten as :

$$\arg\min_{\theta\in\Theta} \mathcal{W}_1(p, p_\theta)$$

Kantorovich-Rubinstein duality plays a key role and enables us to rewrite the 1-Wasserstein distance as :

Property 2.3 *Kantorovich-Rubinstein duality*

$$\mathcal{W}_1(\alpha,\beta) = \sup_{||f||_L \leqslant 1} \mathbb{E}_{x\sim\alpha}[f(x)] - \mathbb{E}_{x\sim\beta}[f(x)] \qquad (2)$$

where the supremum is over all 1-Lipschitz functions $f : \mathcal{X} \mapsto \mathbb{R}$.

Proof. To compute the 1-Wasserstein distance between two probability measures α, β ($|\alpha| = |\beta| = 1$) we need to compute an infinum with constraints on the marginal distributions. So let us derive its associated Lagrangian. We introduce two Lagrangian multipliers f, g that are continuous functions on $\mathcal{X}$ and $\mathcal{Y}$ respectively (dual of probability measures is the set of continuous functions):

$$\begin{aligned}\mathcal{L}(\pi, f, g) = & \int_{\mathcal{X}\times\mathcal{Y}} ||x-y||d\pi(x,y) + \int_{\mathcal{X}} f(x)d\alpha(x) \\ & - \int_{\mathcal{Y}} d\pi(x,y) + \int_{\mathcal{Y}} g(y)d\beta(y) - \int_{\mathcal{X}} d\pi(x,y) \\ = & \ \mathbb{E}_{x\sim\alpha}(f(x)) + \mathbb{E}_{y\sim\beta}(g(y)) \\ & + \int_{\mathcal{X}\times\mathcal{Y}} (||x-y|| - f(x) - g(y))d\pi(x,y)\end{aligned}$$

The primal problem is convex and there exists at least one strictly feasible point $\alpha \bigotimes \beta$ so Slater's conditions apply and strong duality holds :

$$\begin{aligned}\inf_{\pi} \sup_{f,g} \mathcal{L}(\pi, f, g) = & \sup_{f,g} \inf_{\pi} \mathcal{L}(\pi, f, g) \\ = & \sup_{f,g} \mathbb{E}_{x\sim\alpha}(f(x)) + \mathbb{E}_{y\sim\beta}(g(y)) \\ & + \inf_{\pi} \int_{\mathcal{X}\times\mathcal{Y}} (||x-y|| - f(x) \\ & - g(y))d\pi(x,y) \\ = & \sup_{\substack{f\in C(\mathcal{X}) \\ g\in C(\mathcal{Y}) \\ C - f \bigoplus g \geq 0}} \mathbb{E}_{x\sim\alpha}(f(x)) + \mathbb{E}_{y\sim\beta}(g(y)) \\ = & \ \mathcal{W}_1(\alpha,\beta)\end{aligned}$$

where we denoted $C : (x,y) \in \mathcal{X} \times \mathcal{Y} \mapsto ||x-y||$.

Now, let us consider $\mathcal{X} = \mathcal{Y}$, a 1-Lipschitz function h, and π^* the optimal transference plan then :

$$\begin{aligned}\mathbb{E}_{x\sim\beta}(h(x)) - \mathbb{E}_{y\sim\beta}(h(y)) = & \int_{\mathcal{X}} h(x)(\int_{\mathcal{Y}} d\pi^*(x,y)) \\ & - \int_{\mathcal{Y}} h(y)(\int_{\mathcal{X}} d\pi^*(x,y)) \\ = & \int_{\mathcal{X}\times\mathcal{Y}} (h(x) - h(y))d\pi^*(x,y) \\ \leqslant & \int_{\mathcal{X}\times\mathcal{Y}} ||x-y||d\pi^*(x,y) \\ = & \ \mathcal{W}_1(\alpha,\beta)\end{aligned}$$

Therefore,

$$\sup_{||h||_L \leqslant 1} \mathbb{E}_{x\sim\beta}(h(x)) - \mathbb{E}_{y\sim\beta}(h(y)) \leqslant \mathcal{W}_1(\alpha,\beta)$$

On the other hand, let us prove this upper bound is attained and thus this supremum is actually a maximum. To do so, let us consider $\kappa : x \mapsto \inf_u ||x-u|| - g(u)$.

Then, by triangular inequality :

$$\forall x, y \in \mathcal{X}, \kappa(x) \leqslant ||x-y|| + \inf_u ||y-u|| - g(u) = ||x-y|| + \kappa(y)$$

By interchanging the role of x, y, we also get $\kappa(y) - \kappa(x) \leqslant ||y-x||$, meaning that κ is 1-Lipschitz.

Now, for any (f, g) satisfying the constraint $\forall x, y \in \mathcal{X}, f(x) + g(y) \leqslant ||x-y||$, it follows :

$$\forall x \in \mathcal{X}, f(x) \leqslant \kappa(x) \leqslant ||x-x|| - g(x) = -g(x)$$

So, $\forall x \in \mathcal{X}, \begin{cases} f(x) \leqslant \kappa(x) \\ g(x) \leqslant -\kappa(x) \end{cases}$

Finally :

$$\mathbb{E}_{x\sim\beta}(f(x)) + \mathbb{E}_{y\sim\beta}(g(y)) \leqslant \mathbb{E}_{x\sim\beta}(\kappa(x)) - \mathbb{E}_{y\sim\beta}(\kappa(y))$$

Meaning that :

$$\begin{aligned}\mathcal{W}_1(\alpha,\beta) = & \sup_{\substack{f\in C(\mathcal{X}) \\ g\in C(\mathcal{Y}) \\ C - f \bigoplus g \geq 0}} \mathbb{E}_{x\sim\alpha}(f(x)) + \mathbb{E}_{y\sim\beta}(g(y)) \\ \leqslant & \ \mathbb{E}_{x\sim\alpha}(\kappa(x)) - \mathbb{E}_{y\sim\beta}(\kappa(y)) \\ \leqslant & \sup_{||h||_L \leqslant 1} \mathbb{E}_{x\sim\alpha}(h(x)) - \mathbb{E}_{y\sim\beta}(h(y)) \\ \leqslant & \ \mathcal{W}_1(\alpha,\beta)\end{aligned}$$

Thus, equality holds and the supremum is actually a maximum.

Remark 2.1 *The function κ used in the proof is well defined as the Lagrangian multiplier function g is bounded. Indeed, it is a continuous function on $\mathcal{X}$, which is assumed to be compact.*

Recall one wants to derive a parametrized approximated distribution p_θ as close as possible to the true distribution that has induced the data, the objective is to minimize over the parameter space Θ the 1-Wasserstein distance :

Property 2.4 *WGAN's objective*

$$\min_\theta \mathcal{W}_1(p, p_\theta) = \min_\theta \max_{||h||_L \leqslant 1} \mathbb{E}_{x\sim p}(h(x)) - \mathbb{E}_{y\sim p_\theta}(h(y))$$

If we try to encode the function h with a parametrized family of functions induced by parameters $w \in \mathcal{W}$, the objective becomes :
$\min_\theta \max_{w s.t. ||f_w||_L \leqslant 1} \mathbb{E}_{x\sim p}(f_w(x)) - \mathbb{E}_{y\sim p_\theta}(f_w(y))$

Additionally, these expectations can be approximated using finite sums of samples. To generate a sample from the approximate distribution p_θ, we generate a latent variable from a gaussian noise, and then decode it into a vector of $\mathcal{X}$ with a generator function g_θ that is learnt during the training process. Therefore, the final objective can be rewritten as :

Property 2.5 *WGAN's approximate objective*

$$\min_{\theta} \max_{ws.t.||f_w||_L \leqslant 1} \frac{1}{|\tau|} \sum_{x \in \tau} f_w(x) - \frac{1}{N_z} \sum_{z \sim \mathcal{N}(0,\mathcal{I})} f_w(g_\theta(z))$$

2.1.3 Differences between the standard GAN and the WGAN

Difference in the metric used It is observed that compared to other metrics such as the Total Variation distance, the Kullback-Leibler (KL) divergence and the Jensen-Shannon divergence (JS), the 1-Wasserstein distance is strongly related to the notion of weak convergence.

Property 2.6 *On a compact space, the 1-Wasserstein distance does metrize the weak convergence, meaning :* $\alpha_n \longrightarrow^*_{weakly} \alpha \Longleftrightarrow \mathcal{W}_1(\alpha_n, \alpha) \longrightarrow 0$

Remark 2.2 *A topology induced by a metric is said to be weaker than another one if all the sequences converging with respect to the first metric are necessarily converging with respect to the second one.*

The 1-Wasserstein distance has desirable properties in the sense that :

- It is a smooth function, which makes the back propagation possible while training
- It has an intuitive behaviour as we will see on vary simple examples in Section 3.1.

It can be shown that the original GAN's loss function is equal (up to affine coefficients) to the Jensen-Shannon divergence, which is a symmetrized version of the Kullback-Leibler divergence. As such, it may lead to discontinuity in the loss function and using the 1-Wasserstein distance is a way to correct this. By using another metric, WGAN also addresses the problem of vanishing gradients.

Training instability It is shown in [2] that training instability may be tempered with WGAN. The original GAN is known to be hard to train, necessitating specific architectures and hyperparameter tuning. These choices are critical: if the discriminator learns too fast, the generator will not have the possibility to effectively learn. On the contrary, in the case of WGAN, the discriminator - called the critic - can be trained till optimality, so as to have the best possible expression of the 1-Wasserstein distance, on which the generator will derive its gradient. That is why the discriminator is trained more than the generator in the case of GAN. In practice, it is not trained till optimality.

However, [10] argues that the fact that the discriminator is not trained till optimality prevents convergence in certain cases for the WGAN.

2.1.4 Lispchitz constraint

Clipping weights One of the main drawback of this problem formulation is that it needs the function that is optimized to be 1-Lispchitz. The paper introduced a way to enforce the lipschitzness by clipping the weights w with a certain parameter `clip_value`. This way, the gradient is ensured to be bounded as the Lipschitz constraint indicates. However, the authors admit that clipping weights may not be the greatest idea to enforce the Lipschitz constraints.

Gradient penalty An improved method was introduced a few years later that consists in adding a gradient penalty in the objective function so as to constraint the gradient of the function f to be close to 1, as can be done for instance in the optimization field to encode a constraint. This added term is weighted by a parameter λ_{gp}, which results in a trade-off between the quality of the generation and the feasibility of our problem. We will see later on that this proposed method has a high impact on the quality of the generated samples, compared to the clipping method.

2.2. Methodology

Our first aim was to try and reproduce the results from the paper with our own implementation, so as to be able to see how adding levels of complexity to the GAN structure modified the outputs. Due to lack of time, we were not able to test our implementations on the LSUN dataset as we had initially planned, and decided to test on lighter datasets such as MNIST and CIFAR. Since all the implementation is of our making, it cannot of course compete with state-of-the-art papers: our aim was above all to understand the different assertions made for GANs, WGANs, and WGAN-GP.

2.3. Algorithm

We use the DCGAN architecture [12] for all our experiments. We designed the architecture so as to be able to add extra layers for more elaborate datasets. Our optimizer is RMSProp as in the original GAN implementation. For the MNIST dataset, we use a 4-layered architecture, and for the CIFAR dataset, we use a 6-layered architecture. In the WGAN and WGAN-GP implementations, we train the critic 3 times more than the generator, so `n_disc=3`.

Algorithm 1 WGAN Pseudo Code Algorithm

```
for $iter$ to $N_{epochs}$ do
    for $k_{critic}$ steps do
        Sample minibatch of $m$ samples $\{z^1, ..., z^m\}$;
        Sample minibatch of $m$ samples $\{x^1, ..., x^m\}$;
        Update parameters $w$ for the critic $f_w$ by stochastic gradient ascent
        $\nabla_\omega = \frac{\partial}{\partial w}[\frac{1}{m}\sum_i (f_w(x^i) - f_w(g_\theta(z^i))];$
        $w \leftarrow \text{clip}(w, -c, c);$
    Sample minibatch of $m$ samples $\{z^1, ..., z^m\}$;
    Update parameters $\theta$ for the generator $g_\theta$ by stochastic gradient descent
    $\nabla_\theta = \frac{\partial}{\partial \theta}[\frac{1}{m}\sum_i f_w(x^i) - \frac{1}{m}\sum_i f_w(g_\theta(z^i))];$
```

3. Evaluation / Results

3.1. Comparison distances between distributions

Then main argument of [2] to design a new GAN is that using the 1-Wasserstein distance to compute the difference between two probability distributions is better than using the Jensen-Shannon divergence, at the basis of the original GAN.

Dirac measures The example taken in the paper on WGANs [2] uses this example to show the interest of using the 1-Wasserstein distance compared to other measures: the distance measured between two Dirac measures, one in 0 and one in θ is equal to :

- $W_1(\mathbb{P}_0, \mathbb{P}_\theta) = |\theta|$

  for the 1-Wasserstein distance

- $JS(\mathbb{P}_\theta, \mathbb{P}_0) = \begin{cases} \log 2 & \text{if } \theta \neq 0 \\ 0 & \text{if } \theta = 0 \end{cases}$

  for the Jensen-Shannon divergence

- $KL(\mathbb{P}_\theta, \mathbb{P}_0) = \begin{cases} +\infty & \text{if } \theta \neq 0 \\ 0 & \text{if } \theta = 0 \end{cases}$

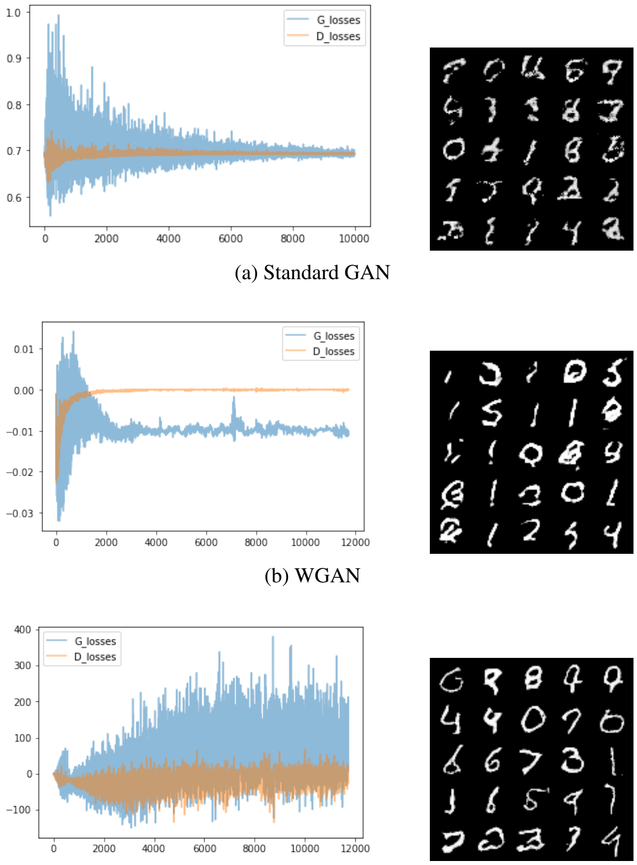
  for the Kullback-Leibler divergence

- $\delta(\mathbb{P}_\theta, \mathbb{P}_0) = \begin{cases} 1 & \text{if } \theta \neq 0 \\ 0 & \text{if } \theta = 0 \end{cases}$

  for the Total Variation distance

We can see that in this case the 1-Wasserstein distance seems a much more appropriate metric to quantify the behaviour of the measure $\mathbb{P}_\theta$ when θ goes to 0. Moreover, the two probability distributions have a disjoint support: this illustrates the possible lack of continuity for the JS divergence which is the principal point of introducing the WGAN.

Figure 2: MNIST dataset



(a) Standard GAN

(b) WGAN

(c) WGAN-GP

3.2. MNIST

Our first application is training the three networks to generate digits by training them with the MNIST dataset [8]. We present our results in figure 2 and 3.

We can see a difference in sample quality: the WGAN-GP indeed seems to retrieve more real-like outputs than the WGAN with clipping gradients and the standard GAN. For the losses, we can see that they globally diminish in the case of the standard GAN and WGAN with clipping gradients. As the WGAN-GP introduces a penalizing term is the loss, it is not necessarily bounded and we see that we get a lot of oscillations, with a totally different scale in the y-axis. The unboundedness of the loss could explain this fact.

In figure 3, we plotted the WGAN-GP results from an interpolation between two points in the latent space: we can see a digit 4 turning into a 1 and then into a 7.

Figure 3: Interpolations with WGAN-GP

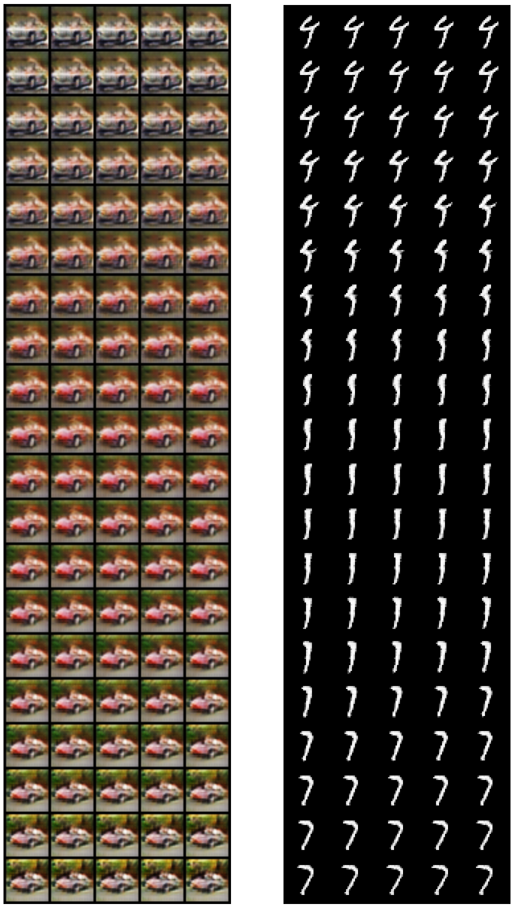
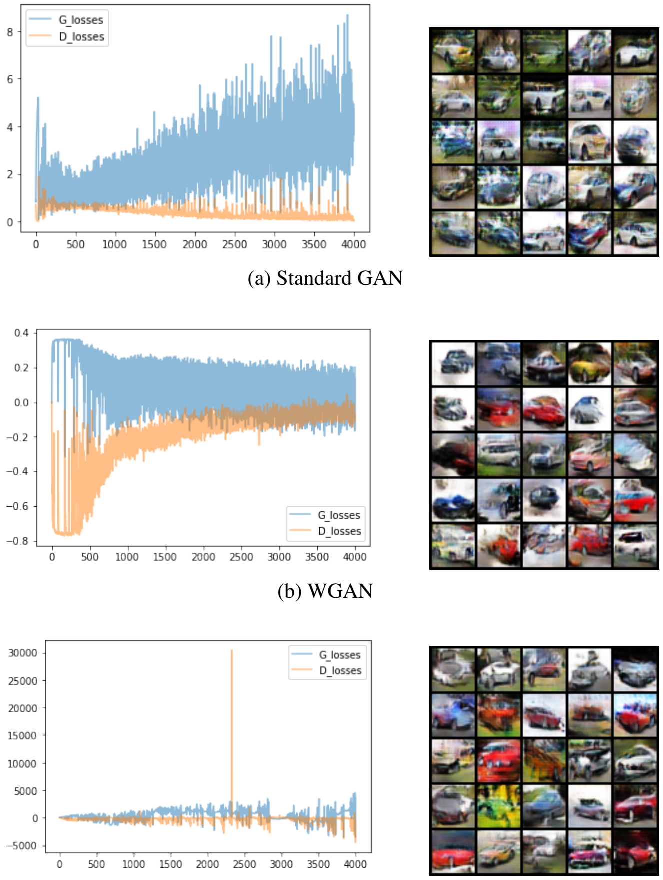


Figure 4: CIFAR dataset



(a) Standard GAN

(b) WGAN

(c) WGAN-GP

3.3. CIFAR

We present our results in figure 3, 4 and . In order to alleviate training execution time, we only trained our networks on a single class of CIFAR10: the class of cars. CIFAR10 is a dataset with 32 × 32 image size dataset representing various classes with low quality images but a wide variance in the data: we here have cars in various environments, with various colors and camera angles.

Again, we must take the results on the loss for WGAN-GP with caution. In the case of the standard GAN, we can see that the generator's loss goes up after some iterations: this can be explained because the discriminator is very well-trained and the generator finds it hard to fool it. Visually speaking, we can guess that the Standard GAN outputs samples that have low variety. The WGAN gives more variety in color, but we have many images that do not really have the shape of a car. The WGAN-GP results generally recover finer details.

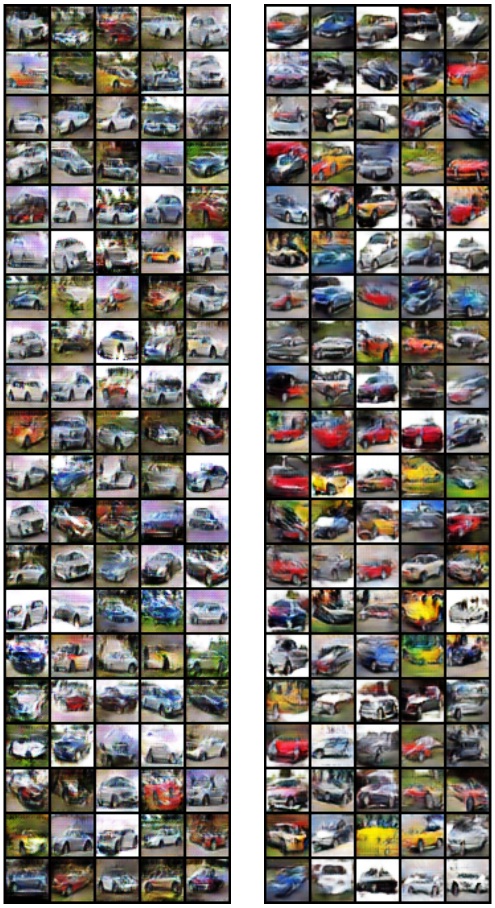
In figure 3, we can see a white car turning gradually red and then becoming smaller and white again.

In figure 5, we plotted random samples to show the diversity of outputs for the GAN and WGAN-GP in order to see if we could see the phenomenon of mode collapse in the standard GAN compared to the WGAN-GP. Mode collapse is the fact that the Standard GAN often gets restricted during training to a specific part of the distribution and thus loses diversity in its generated outputs. We can see that the Standard GAN outputs in our example a significant amount of white cars with purple background, and has less color diversity than the WGAN-GP.

3.4. Discussion

We recall that our first objective was to reproduce the results of the initial paper from Arjovsky *et al.* [2]. Given our limiter resources and time, we decided to train on lighter datasets, such as MNIST and CIFAR10. We know that our results are far from the optimal output that could be produced: for instance, it might be worth trying to have a better interpretation of the generator and discriminator's losses behaviours. However, by building by ourselves the architectures and training processes, we were able to under-

Figure 5: Illustration of mode collapse


(a) Standard GAN

(b) WGAN-GP

stand fully the complexity behind training different types of GANs, and thus comprehend why it is today an active area of research.

4. Conclusion

The introduction of GANs by Goodfellow *et al.* [4] has opened a series of possibilities and axes of research that WGANs try to give an answer to. The introduction of WGAN has interesting theoretical properties that however does not always give satisfying implementation results, which is mainly due to approximations that have to be made to have a tractable algorithm (weight clipping and sub-optimality of the discriminator's training at each iteration). Designing instead gradient penalty tackles the problems raised by weight clipping but does not solve all questions: the behaviour of the loss is less easy to interpret for instance. All those problems of training instability and convergence, of mode collapse, and of the right hyperparameter tuning, although alleviated by the introduction of WGAN and WGAN-GP, remain open questions: on that subject, [13] gives a broad view of the current axes of research, with other questions such as the right way to quantify the performances of the proposed GANs, the introduction of conditional GANs and alternative optimization strategy.

References

[1] M. Arjovsky and L. Bottou. Towards principled methods for training generative adversarial networks, 2017. 1, 2

[2] M. Arjovsky, S. Chintala, and L. Bottou. Wasserstein gan. 1, 2, 4, 5, 6

[3] A. Brock, J. Donahue, and K. Simonyan. Large scale gan training for high fidelity natural image synthesis. 2

[4] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial networks. 1, 2, 7

[5] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. Courville. Improved training of wasserstein gans. 2

[6] A. Kadurin, S. Nikolenko, K. Khrabrov, A. Aliper, and A. Zhavoronkov. Drugan: An advanced generative adversarial autoencoder model for de novo generation of new molecules with desired molecular properties in silico. *Molecular Pharmaceutics*, 14, 07 2017. 1

[7] T. Karras, S. Laine, and T. Aila. A style-based generator architecture for generative adversarial networks. 2

[8] Y. LeCun and C. Cortes. MNIST handwritten digit database. 2010. 5

[9] H. Liu, X. Gu, and D. Samaras. Wasserstein gan with quadratic transport cost. pages 4831–4840, 10 2019. 2

[10] L. Mescheder, A. Geiger, and S. Nowozin. Which training methods for gans do actually converge? 4

[11] G. Peyré and M. Cuturi. Computational optimal transport. 2

[12] A. Radford, L. Metz, and S. Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. 1, 4

[13] D. Saxena and J. Cao. Generative adversarial networks (gans): Challenges, solutions, and future directions. *ArXiv*, abs/2005.00065, 2020. 7

[14] A. Senior, R. Evans, J. Jumper, J. Kirkpatrick, L. Sifre, T. Green, C. Qin, A. Žídek, A. Nelson, A. Bridgland, H. Penedones, S. Petersen, K. Simonyan, S. Crossan, P. Kohli, D. Jones, D. Silver, K. Kavukcuoglu, and D. Hassabis. Improved protein structure prediction using potentials from deep learning. *Nature*, 577:1–5, 01 2020. 1

[15] C. Villani. *Optimal Transport: Old and New*. Grundlehren der mathema- tischen Wissenschaften, 2009. 2

[16] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. 2