IFT 6756 - Lecture 13
From GAN training to Optimization of Differentiable Games

This version of the notes has not yet been thoroughly checked. Please report any bugs to the scribes or instructor.

Scribes **Instructor:** Gauthier Gidel
Winter 2021: [Carl Perreault-Lafleur, Jonathan Tremblay, François Milot]

1 Summary

In the previous lecture we introduced Wasserstein GANs (WGANs) which have the following min-max objective function:

$$\min_{\theta} \max_{\phi} \mathrm{E}_{x\sim p_d}[F_\phi(x)] - \mathrm{E}_{z\sim p_z}[F_\phi(G_\theta(z)] \tag{1}$$

We understand that this payoff function depends on two parameters: ϕ and θ. The first one is used to define the discriminator whereas the second defines the generator function. Interestingly, this objective can be casted as a minimization of the Wasserstein distance as per the the WGAN paper [1]. The Wasserstein distance being:

$$W(p,q) = \inf_{\gamma\in\Pi(p,q)} \mathrm{E}_{(x,y)\sim\gamma}[\|x-y\|] \tag{2}$$

The motivation to minimize this distance is driven by the fact that it has interesting properties that the Jensen-Shannon or Kullback-Leibler divergences do not have (these divergences are commonly used to characterize the *classic* GANs). An attractive property of the WGAN is that it provides more stability during training.

In the this lecture, we will illustrate how to optimize a min-max objective function using gradient based optimization. More generally, GANs and WGANs objective functions can be described as:

$$\min_{\theta} \max_{\phi} \mathcal{L}(\theta,\phi) \tag{3}$$

This function can then be optimized using gradient descent-ascent and the goal will be to reach a point where the gradients are zero. We will later prove that in some cases this mean we have reached a Nash equilibrium.

2 GANs as differentiable games

As previously mentioned, the general objective function of a GAN can be defined as:

$$\min_{\theta} \max_{\phi} \mathcal{L}(\theta,\phi)$$

In order to solve the above objective, one could leverage a gradient descent-ascent approach to optimize the two parameters ϕ and θ:

$$\begin{cases} \theta_{t+1} = \theta_t - \eta\nabla_\theta\mathcal{L}(\theta_t,\phi_t) \\ \phi_{t+1} = \phi_t - \eta\nabla_\phi\mathcal{L}(\theta_t,\phi_t) \end{cases}$$

What the above suggest is that we are now looking for a set of points where the gradients are zero. It also means we are looking for a wider set of points than the min-max since the gradients can equal zero in more scenarios. In other words, a downside of this approach is that it is possible to get stuck in a situation where the gradients are zeros without

being sure we are at a global solution.

Interestingly, the above objective function can be formulated as a minimax two player zero-sum game. This concept was first defined within the original GAN paper [4]. All things considered, the objective of the lecture will be to demonstrate that it is possible to optimize such games (or objective) using a gradient based approach and that we can converge to a Nash equilibria given some specific settings.

2.1 Smooth games

Let's say we want to solve the following objective function:

$$\min_{\theta} \max_{\phi} \mathcal{L}(\theta, \phi)$$

This equation is not always easy to understand. Although, one can interpret the above as being the same as finding a Nash equilibrium for a two-player minimax game with a value function of $\mathcal{L}(\theta, \phi)$.

Definition 1 (Nash equilibrium for a two parameters minimax game)**.** *A Nash equilibrium is defined as a pair (θ^*, ϕ^*) where:*

$$\mathcal{L}(\theta^*, \phi) \leq \mathcal{L}(\theta^*, \phi^*) \leq \mathcal{L}(\theta, \phi^*) \quad \forall(\theta, \phi)$$

To find the Nash equilibrium of this game, we first need to do some assumptions with respect to the payoff function.

Assumption 2 (Differentiable convex-concave payoff)**.** *If the payoff function $\mathcal{L}(\theta, \phi)$ is convex in θ, concave in ϕ and differentiable for both parameters, it implies that we are at a Nash if and only if:*

$$\nabla \mathcal{L}(\theta^*, \phi^*) = 0$$

In other words, being at a Nash only happens when the gradients are zero. It is the case because of the nature of the Nash equilibrium's definition. For the above assumption to hold, we first need to be sure that a Nash actually exists in such settings. To do so, we will consider the following theorem:

Theorem 3 (Sion's Theorem [7])**.** *If the payoff function is convex-concave where set U and V are convex and compact, we have:*

$$\min_{\theta \in U} \max_{\phi \in V} \mathcal{L}(\theta, \phi) = \max_{\phi \in V} \min_{\theta \in U} \mathcal{L}(\theta, \phi)$$

With this theorem, we will be able to prove that at the point when the min-max equals the max-min, we are at a Nash:

Proof.

$$\text{Let} : \theta^* \in \arg\min_{\theta} \max_{\phi} \mathcal{L}(\theta, \phi) \; ; \; \phi^* \in \arg\max_{\phi} \min_{\theta} \mathcal{L}(\theta, \phi) \textbf{ and } \min_{\theta} \max_{\phi} \mathcal{L}(\theta, \phi) = \max_{\phi} \min_{\theta} \mathcal{L}(\theta, \phi)$$

$$\text{Where: } (\theta^*, \phi^*) \text{ is a Nash}$$

We know our function is convex with respect to θ and concave with respect to ϕ. It will mean that if we first minimize with respect to θ for any given ϕ and then maximize ϕ we will have a value that it is smaller or equal to the Nash:

$$\max_{\phi} \min_{\theta} \mathcal{L}(\theta, \phi) \leq \mathcal{L}(\theta^*, \phi^*)$$

And if we first maximize with respect to ϕ for any given θ and then minimize with respect to θ, we will have a value that is greater or equal to the Nash:

$$\mathcal{L}(\theta^*, \phi^*) \leq \min_{\theta} \max_{\phi} \mathcal{L}(\theta, \phi)$$

Although, as per Sion's theorem, we know that the min-max equals the max-min. Inserting the previous definition of our optimal θ and ϕ, we get:

$$\max_{\phi} \min_{\theta} \mathcal{L}(\theta, \phi) = \min_{\theta} \mathcal{L}(\theta, \phi^*) \leq \mathcal{L}(\theta^*, \phi^*) \leq \max_{\phi} \mathcal{L}(\theta^*, \phi) = \min_{\theta} \max_{\phi} \mathcal{L}(\theta, \phi)$$

Given the function is convex in θ:

$$\min_{\theta} \mathcal{L}(\theta, \phi^*) \leq \mathcal{L}(\theta^*, \phi^*) \implies \min_{\theta} \mathcal{L}(\theta, \phi^*) = \mathcal{L}(\theta^*, \phi^*)$$

Similarly for ϕ (concave):

$$\mathcal{L}(\theta^*, \phi^*) \leq \max_{\phi} \mathcal{L}(\theta^*, \phi) \implies \mathcal{L}(\theta^*, \phi^*) = \max_{\phi} \mathcal{L}(\theta^*, \phi)$$

Which proves that the point when min-max equals max-min is a Nash. □

One can also be interested in proving Sion's theorem:

Proof. We assume that u and v are compact and convex sets and $\mathcal{L}(\theta, \phi)$ is convex-concave.

We define two players: *Player 1* can play a strategy i from $[n]$ possible strategies and *Player 2* can play a strategy j from $[m]$ possible strategies. The payoff value of this game would be defined as $l_{i,j}$. We define a mixed strategies (with finite possible actions) approach where μ is the mixed strategies for *Player 1* and ν is the mixed strategies for *Player 2*. Combining the above assumptions gives us the following payoff function:

$$\mathcal{L}(\mu, \nu) = \mathbb{E}_{i \sim \mu, j \sim \nu}[l_{i,j}] = \sum_{1 \leq i \leq n, 1 \leq j \leq m} l_{i,j} \mu_i \nu_j$$

Where μ_i is the probability of *Player 1* of playing action i (ν_j for *Player 2*). The above can be re-written in matrix notation since it can be represented as a sum-product:

$$\mathcal{L}(\mu, \nu) = \mu^T L \nu$$

Where:

$$\begin{bmatrix} l_{1,1} & ... & l_{1,m} \\ ... & l_{i,j} & ... \\ l_{n,1} & ... & l_{n,m} \end{bmatrix}, \mu \in \Delta_n = \{\mu \in \mathbb{R}^n / \mu^T \mathbf{1} = 1, \mu > 0\}, \nu \in \Delta_m = \{\nu \in \mathbb{R}^m / \nu^T \mathbf{1} = 1, \nu > 0\}$$

We just need to prove that the above sets are compact and convex which will then prove that $\mathcal{L}(\mu, \nu)$ is convex-concave. Since $\mu^T L \nu$ is linear which can be defined as being convex and concave, we can say that $\mathcal{L}(\mu, \nu)$ is convex-concave as well since $\mathcal{L}(\mu, \nu) = \mu^T L \nu$.

Finally, we can state that the above sets are convex since μ and ν are probability vectors of dimension n and m and therefore taking a convex combination of those probability vector will also give a probability vector.

These sets are also compact since the values are bounded within $0 \leq \mu_i \leq 1$ and $0 \leq \nu_i \leq 1$ which will also mean that $\|\mu\| \leq n$ and $\|\nu\| \leq m$. It also confirms these are compact sets. □

We can conclude that when we are in the space of mixed strategies where the payoff is convex-concave and the sets of strategies are convex, Sion's theorem can be seen as a generalization of the Nash's equilibrium within specific settings. Therefore, we want to work in such a space (that complies with Sion's theorem) since it will always have a Nash equilibria which is exactly what we want. This opens the door to questioning how min-max theorem applies to a GAN environment where functions are not convex-concave.

2.1.1 Nash equilibria for linear WGAN min-max game

The previous section was a motivation to apply such concept to GANs. As an example, in the current section, we will define a WGAN within the settings we defined in the previous section.

Let's first start with the definition of the WGAN:

$$\mathcal{L}(\theta, \phi) = \min_{\theta} \max_{\phi} \mathrm{E}_{x \sim p_d}[F_\phi(x)] - \mathrm{E}_{z \sim p_z}[F_\phi(G_\theta(z))]$$

We first define the linear discriminator and generator as:

$$F_\phi(x) = \phi \cdot x \ , \ G_\theta(z) = \theta_1 z + \theta_2 \text{ where } z \sim \mathcal{N}(0,1)$$

Which means:

$$\mathcal{L}(\phi, \theta) = \mathrm{E}_{x \sim p_d}[\phi \cdot x] - \mathrm{E}_{z \sim p_z}[\phi(\theta_1 z + \theta_2)]$$

By linearity and by applying the expectation on z, we get:

$$\min_\theta \max_\phi \phi \cdot (\mathrm{E}_{x \sim p_d}[x] - \theta_2) \tag{4}$$

Therefore we understand that the goal of this game is to match the means (since we can see that θ_2 is the mean of the generated variable) and that the Nash equilibria is achieved when θ_2 equals the mean of x. Finally, we can prove that this is a Nash:

$$\mathcal{L}(\theta, \phi) = \phi \cdot (\mu - \theta) \text{ where } \mu = \mathbb{E}_{x \sim p_d}[x]$$

Taking the gradients of the above function with respect to our parameters, we get:

$$\nabla_\theta \mathcal{L}(\phi, \theta) = -\phi \ , \ \nabla_\theta \mathcal{L}(\phi, \theta) = \mu - \theta$$

Equalling these two values to zero implies:

$$\phi^* = 0 \ , \ \theta^* = \mu$$

Finally, we can find the Nash equilibria definition by plugging the found value within the Nash equilibria definition:

$$0 = \mathcal{L}(\theta^*, \phi) \leq \mathcal{L}(\theta^*, \phi^*) = 0 \leq \mathcal{L}(\theta, \phi^*) = 0 \ , \forall (\theta, \phi)$$

Which confirms we found a Nash.

2.2 Solving the game

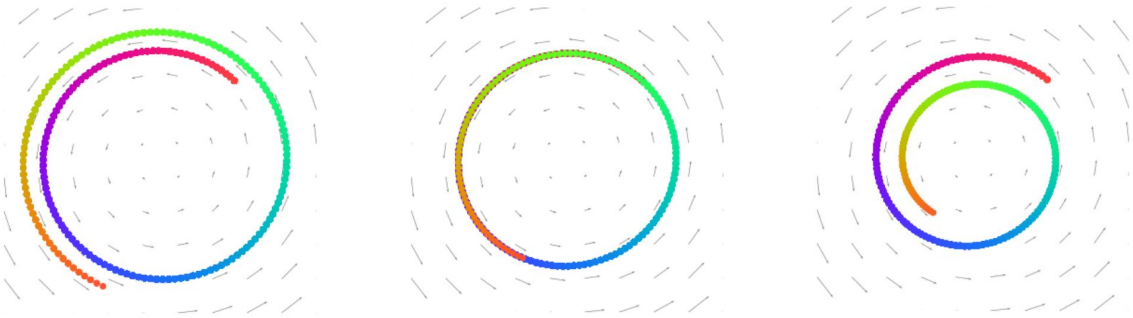
By translation of θ, we can reformulate Equation 4 as:

$$\min_\theta \max_\phi \theta \cdot \phi$$

Result 4 (The Nash equilibrium of this game is (0,0))**.**

Proof. This is easy to see. If θ chooses 0, then ϕ has no strictly better solution than playing 0. The same argument works the other way around. □

We will now explore different methods for solving the game, and observe if they converge to the Nash equilibrium.



(a) Sim-GDA (b) Alt-GDA (c) PPM

Figure 1: Exploration of different update rules: (a) Simultaneous Gradient Descent-Ascent (b) Alternated Gradient Descent-Ascent (c) Proximal Point Method

2.2.1 Simultaneous Gradient Descent-Ascent

We can now solve this simple bi-linear game with gradient-based method. Since $\nabla_\theta(\theta \cdot \phi) = \phi$ and $\nabla_\phi(\theta \cdot \phi) = \theta$, we obtain that the gradient descent steps are:

$$\begin{cases} \theta_{t+1} = \theta_t - \eta\phi_t \\ \phi_{t+1} = \phi_t + \eta\theta_t \end{cases} \tag{5}$$

By observing the dynamic of these updates presented in Figure 1(a), we can see that the parameters diverge instead of converging to the solution $(0, 0)$. We can prove this formally.

Result 5 (Sim-GDA diverges)**.** *We have*

$$\theta_t^2 + \phi_t^2 = \rho^t(\theta_0^2 + \phi_0^2) > (\theta_0^2 + \phi_0^2)$$

where $\rho = 1 + \eta^2 > 1$. Note that for $\eta = 0$, there are no updates.

Proof. We observe the squared L_2-norm for (θ, ϕ) at time t, that we call $L_t = \theta_t^2 + \phi_t^2$. We have:

$$\begin{aligned} L_{t+1} &= \theta_{t+1}^2 + \phi_{t+1}^2 \\ &= (\theta_t - \eta\phi_t)^2 + (\phi_t - \eta\theta_t)^2 && \text{using Equation 5} \\ &= \theta_t^2 + \phi_t^2 - 2\eta\theta_t\phi_t + \eta^2\phi_t^2 + 2\eta\phi_t\theta_t + \eta^2\theta_t^2 && \text{by developing} \\ &= L_t + \eta^2 L_t && \text{after canceling and regrouping} \\ &= (1 + \eta^2)L_t \\ &= (1 + \eta^2)^{t+1} L_0 \end{aligned}$$

□

So far, we have only looked at $\eta = \eta_1 = \eta_2$. However, the usage of different η is useful for different loss functions that may have different scales. We can show the analog result for different step-size.

Result 6 (Sim-GDA diverges even with different step sizes)**.** *We still have*

$$\frac{\theta_t^2}{\eta_1} + \frac{\phi_t^2}{\eta_2} = \frac{\theta_0^2}{\eta_1} + \frac{\phi_0^2}{\eta_2} + \eta_1 \sum_{t'=0}^{t-1} \phi_{t'-1}^2 + \eta_2 \sum_{t'=0}^{t-1} \theta_{t'-1}^2 > \frac{\theta_0^2}{\eta_1} + \frac{\phi_0^2}{\eta_2}$$

for the update rule

$$\begin{cases} \theta_{t+1} = \theta_t - \eta_1\phi_t \\ \phi_{t+1} = \phi_t + \eta_2\theta_t \end{cases}$$

Note that for $\eta_1 = \eta_2 = 0$, there are no updates.

Two proofs are possible to show the previous result:

Proof. (#1). After rescaling θ and ϕ, we have:

$$\begin{aligned} L_{t+1} &= \frac{\theta_{t+1}^2}{\eta_1} + \frac{\phi_{t+1}^2}{\eta_2} \\ &= \frac{(\theta_t - \eta\phi_t)^2}{\eta_1} + \frac{(\phi_t - \eta\theta_t)^2}{\eta_2} && \text{using Equation 5} \\ &= \frac{\theta_t^2}{\eta_1} + \eta_1\phi_t^2 - 2\theta_t\phi_t + \frac{\phi_t^2}{\eta_2} + \eta_2\theta_t^2 + 2\theta_t\phi_t && \text{by developing} \\ &= L_t + \eta_1\phi_t^2 + \eta_2\theta_t^2 && \text{after canceling and regrouping} \end{aligned}$$

□

Proof. (#2). By analyzing the eigenvalues of this problem we can also see the divergence as well. First, we got the matrix representation of the update rule as being:

$$w_{t+1} = \begin{pmatrix} \theta_{t+1} \\ \phi_{t+1} \end{pmatrix} = \begin{pmatrix} 1 & -\eta_1 \\ \eta_2 & 1 \end{pmatrix} \begin{pmatrix} \theta_t \\ \phi_t \end{pmatrix} = Aw_t$$

Let assume that $\exists \lambda, s.t. |\lambda| > 1, \lambda + Sp(A)$, then $\exists x, s.t. Ax = \lambda x$. From this, we can see that:

$$||w_{t+1}|| = ||Aw_t|| = |\lambda|^{t+1}||w_0||$$

Since $|\lambda| > 1$, we can easily see that this is diverging as $t \to \infty$.

In regards to the relationship between η_1 and η_2, we can see that:

$$Trace(A) = \lambda_1 + \lambda_2 = 2$$
$$det(a) = \lambda_1 \lambda_2 = 1 + \eta_1 \eta_2 > 1$$

□

We notice that increasing η implies a faster divergence. As we can see in Figure 2, it makes sense because the vector field we are following in green is tangent to the circle. We cannot get inside the circle which is a necessary condition for converge (as shown with the purple arrow). Moreover, since we are doing discrete steps, we will overshoot (red arrow) and thus increase the norm, which will eventually go to infinity.

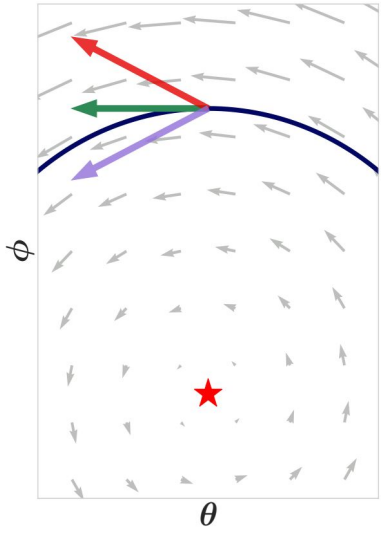


Figure 2: The gradient may point to a direction (red arrow) that pushes the iterate away from the Nash Equilibrium (red star). Figure from [2].

This diverging behaviour happens not only for the studied example. It happens in particular for all games that can be formulated with payoff $\mathcal{L}(\theta, \phi) = \theta^{\mathrm{T}} A \phi$, where $A \in \mathrm{R}^{n \times m}$. But since many games from the WGAN setting have an objective function of that form around the optimum by doing Taylor expansion, this problematic behaviour arises for several games by using gradient methods. This is proven formally in [6], and we give an overview of the proof in the following discussion.

Assuming we are in the WGAN setting, where

$$\mathcal{L}(\theta, \phi) = \mathrm{E}_{p_d}\left[f_\phi(x)\right] - \mathrm{E}_{p_z}\left[f_\phi(G_\theta(x))\right]$$

And $\phi^\star$ is such that $f_{\phi^\star}(x) = 0 \; \forall x$, and $\theta^\star$ is such that G generates the real data distribution. If the pair (θ, ϕ) is close enough from the optimum $(\theta^\star, \phi^\star)$, we can use Taylor expansion to show that:

$$\begin{aligned}\mathcal{L}(\theta,\phi) &\sim \mathcal{L}(\theta^\star,\phi^\star) + \nabla\mathcal{L}(\theta^\star,\phi^\star)^{\mathrm{T}}\begin{pmatrix}\theta-\theta^\star\\ \phi-\phi^\star\end{pmatrix} + \nabla^2\mathcal{L}(\theta^\star,\phi^\star)^{\mathrm{T}}\begin{pmatrix}\theta-\theta^\star\\ \phi-\phi^\star\end{pmatrix}\\ &= \mathcal{L}(\theta^\star,\phi^\star) + J\begin{pmatrix}\theta-\theta^\star\\ \phi-\phi^\star\end{pmatrix} \qquad \text{since } \nabla\mathcal{L}(\theta^\star,\phi^\star)=0\end{aligned}$$

Where

$$J = \nabla^2\mathcal{L}(\theta^\star,\phi^\star)^{\mathrm{T}} = \begin{pmatrix}\nabla^2_\theta\mathcal{L}(\theta^\star,\phi^\star) & \nabla_\phi\nabla_\theta\mathcal{L}(\theta^\star,\phi^\star)\\ \nabla_\theta\nabla_\phi\mathcal{L}(\theta^\star,\phi^\star) & \nabla^2_\phi\mathcal{L}(\theta^\star,\phi^\star)\end{pmatrix} = \begin{pmatrix}0 & \nabla_\phi\nabla_\theta\mathcal{L}(\theta^\star,\phi^\star)\\ \nabla_\theta\nabla_\phi\mathcal{L}(\theta^\star,\phi^\star) & 0\end{pmatrix}$$

and the top left 0 comes from the fact that $\mathcal{L}(\theta,\phi^\star)=0\ \forall\theta$, while the bottom right 0 arises under certain assumptions detailed in [6].

We can now approximate the update from the gradient method as follows:

$$\begin{pmatrix}\nabla_\theta\mathcal{L}(\theta,\phi)\\ -\nabla_\phi\mathcal{L}(\theta,\phi)\end{pmatrix} \sim \tilde{J}\begin{pmatrix}\theta-\theta^\star\\ \phi-\phi^\star\end{pmatrix} = \begin{pmatrix}A^{\mathrm{T}}(\phi-\phi^\star)\\ -A(\theta-\theta^\star)\end{pmatrix} \tag{6}$$

Where

$$\tilde{J} = \begin{pmatrix}0 & \nabla_\theta\nabla_\phi\mathcal{L}(\theta^\star,\phi^\star)\\ -\nabla_\phi\nabla_\theta\mathcal{L}(\theta^\star,\phi^\star) & 0\end{pmatrix} = \begin{pmatrix}0 & A^{\mathrm{T}}\\ -A & 0\end{pmatrix} \text{ under certain assumptions detailed in [6].}$$

We can find back in Equation 6 the form $\theta^{\mathrm{T}}A\phi$ discussed previously and for which the gradient method diverges. [5] discusses the assumptions made above, and general convergence for different training methods in GANs.

2.2.2 Alternating Gradient Descent-Ascent

Now, when implementing the previous algorithm in practice (Simultaneous Gradient Descent-Ascent), the update of the parameters, θ,ϕ, are done sequentially. That means that if we first update θ_t and then we update ϕ_t, the second parameters being updated (ϕ) is using the update of the other parameters (θ_{t+1}) in its calculation of the gradient. This behavior introduce well the second approach which is the alternating gradient descent-ascent.

Formally, we can define the parameters' update as:

$$\begin{cases}\theta_{t+1} = \theta_t - \eta\phi_t\\ \phi_{t+1} = \phi_t + \eta\theta_{t+1}\end{cases} \tag{7}$$

An illustration of the convergence results of this approach is shown in 1(b).

Result 7 (Alt-GDA is not converging and show stationnary behavior)**.**

Proof. The proof is similar to the one from the SDA algorithm.

$$\begin{pmatrix}\theta_{t+1}\\ \phi_{t+1}\end{pmatrix} = \begin{pmatrix}\theta_t - \eta\phi_t\\ \phi_t + \eta\theta_{t+1}\end{pmatrix} \iff \begin{pmatrix}\theta_{t+1}\\ \phi_{t+1} - \eta\theta_{t+1}\end{pmatrix} = \begin{pmatrix}\theta_t - \eta\phi_t\\ \phi_t\end{pmatrix}$$

We can represent the above equation by another matrix representation:

$$\begin{pmatrix}1 & 0\\ -\eta & 1\end{pmatrix}\begin{pmatrix}\theta_{t+1}\\ \phi_{t+1}\end{pmatrix} = \begin{pmatrix}1 & -\eta\\ 0 & 1\end{pmatrix}\begin{pmatrix}\theta_t\\ \phi_t\end{pmatrix}$$

Now, we can inverse the left hand side to get only the $t+1$ expressions.

$$w_{t+1} = \begin{pmatrix}\theta_{t+1}\\ \phi_{t+1}\end{pmatrix} = \begin{pmatrix}1 & 0\\ \eta & 1\end{pmatrix}\begin{pmatrix}1 & -\eta\\ 0 & 1\end{pmatrix}\begin{pmatrix}\theta_t\\ \phi_t\end{pmatrix} = \begin{pmatrix}1 & -\eta\\ \eta & 1-\eta^2\end{pmatrix}\begin{pmatrix}\theta_t\\ \phi_t\end{pmatrix} = Aw_t$$

When looking at the determinant of A, we get that

$$det(A) = \lambda_1 \lambda_2 = 1 - \eta^2 + \eta^2 = 1$$

Finally, we can write that:

$$||w_t|| = ||A^t w_0|| \leq ||A^t|| ||w_0|| = ||PD^t P^{-1}|| ||w_0|| \leq max_i(|\lambda_i|)^t ||w_0|| \leq c||w_0||$$

It is now bounded by a constant independent from t which proves the result. For a more formal proof, you can look at Gidel et al. [3]. □

2.2.3 An improved Gradient Descent: the Proximal Point Method

The last method has the following form:

$$\begin{cases} \theta_{t+1} = \theta_t - \eta \phi_{t+1} \\ \phi_{t+1} = \phi_t + \eta \theta_{t+1} \end{cases} \tag{8}$$

An illustration of the convergence results of this approach is shown in 1(c).

Result 8 (Proximal-Point Method converges)**.** *We have*

$$\theta_t^2 + \phi_t^2 = \rho^t(\theta_0^2 + \phi_0^2) < (\theta_0^2 + \phi_0^2)$$

where $\rho = \frac{1}{1+\eta^2} < 1$*. Note that for* $\eta = 0$*, there are no updates.*

Proof. Let us start by finding an explicit form for the updates:

$$\begin{cases} \theta_{t+1} = \theta_t - \eta \phi_{t+1} = \theta_t - \eta(\phi_t + \eta \theta_{t+1}) \\ \phi_{t+1} = \phi_t + \eta \theta_{t+1} = \phi_t + \eta(\theta_t - \eta \phi_{t+1}) \end{cases} \tag{9}$$

which leads to

$$\begin{cases} (1+\eta^2)\theta_{t+1} = \theta_t - \eta \phi_t \\ (1+\eta^2)\phi_{t+1} = \phi_t + \eta \theta_t \end{cases} \tag{10}$$

We observe the squared L_2-norm for (θ, ϕ) at time t, that we call $L_t = \theta_t^2 + \phi_t^2$. We have:

$$\begin{aligned} L_{t+1} &= \theta_{t+1}^2 + \phi_{t+1}^2 \\ &= (\frac{\theta_t - \eta \phi_t}{1+\eta^2})^2 + (\frac{\phi_t + \eta \theta_t}{1+\eta^2})^2 \\ &= \frac{1}{(1+\eta^2)^2}\left(\theta_t^2 + \phi_t^2 - 2\eta\theta_t\phi_t + \eta^2\phi_t^2 + 2\eta\phi_t\theta_t + \eta^2\theta_t^2\right) \\ &= \frac{L_t}{(1+\eta^2)} \\ &= \frac{L_0}{(1+\eta^2)^{t+1}} \end{aligned}$$

□

In practice, this problem is harder than what we have introduced because the gradients are not necessarily linear and the behavior is not trivial. In conclusion, we have experience on very simple games to evaluate the different methods and we have shown that the promising one was the proximal point method but this method is not deemed practical. Next week, we will see an explicit method (extra-gradient methods) that is similar to the proximal method.

References

[1] M. Arjovsky, S. Chintala, and L. Bottou. Wasserstein gan, 2017.

[2] T. Chavdarova, G. Gidel, F. Fleuret, and S. Lacoste-Julien. Reducing noise in gan training with variance reduced extragradient, 2020.

[3] G. Gidel, H. Berard, G. Vignoud, P. Vincent, and S. Lacoste-Julien. A variational inequality perspective on generative adversarial networks, 2019.

[4] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial networks, 2014.

[5] L. Mescheder, A. Geiger, and S. Nowozin. Which training methods for gans do actually converge?, 2018.

[6] V. Nagarajan and J. Z. Kolter. Gradient descent gan optimization is locally stable, 2018.

[7] M. Sion. On general minimax theorems. *Pacific Jouronal of Mathematics*, 8(1):171–176, 1958.