Optimization for Machine Learning 50.579

Instructor: Ioannis Panageas Scribed by: ZHAO Yunqing(1004964) ; GONG Jia(1005044)

Lecture 7. Introduction to Min-max Optimization.

1 Introduction to GANs

The min-max ideas can be stemmed from zero-sum game, among which the most popular representative series are GANs(Generative Adversarial Nets)[2] In this week, we study the construction of GAN, as well as its core idea with mathematical proof. Generally, we also care about the min-max optimization and related analysis involved in previous lecture.

1.1 Definition

Generative Adversarial Nets is a generative model via adversarial training process, in which we can optimize two nets simultaneously: a generative models G with parameters θ that captures and simulates the data distribution; a discriminative models D with parameters w that estimates the probability of a sample from ground true distribution or from the discriminative model.

Precisely, one would like to solve the problem with objective function

$$\min_{\theta}\max_{w}\mathbb{E}_{x\sim p_{data}}[\log D_w(x)]+\mathbb{E}_{z\sim p_{noise}}[\log(1-D_w(G_\theta(z)))] \tag{1.1}$$

where p_{data} is the real data distribution, and p_{noise} is the noise distribution of Gaussian, which will be sent to generator G to create the fake sample. Maximizing D means D is trying to maximize the probability to assign correct label to both kinds of data, i.e., true label for sample from real data distribution, and false label for sample from generated data distribution.

One can also write the objective function in a simple form

$$\min_{\theta}\max_{w}\mathbb{E}_{x\sim Q}[D_w(x)]-\mathbb{E}_{z\sim F}[D_w(G_\theta(z))] \tag{1.2}$$

where the G_θ is the generator with parameters θ and D_w with parameters w is the discriminator. Q is the data distribution and F say Gaussian noise.

1.2 Optimal Discriminator of GAN

The final destination of GAN is to obtain a powerful generator G_θ that can produce the sample almost the same as the ground truth, fooling a good discriminator to remain unclear with true and fake samples.

In this case, we firstly explore the optimal discriminator as a prior condition for clarity.

Lemma 1.1 (Optimality) *For a fixed generator G, the optimal discriminator D has the density*

$$D_{w^*} = \frac{p_{data}(x)}{p_{data}(x) + p_G(x)} \tag{1.3}$$

where $p_G(x)$ is the implicit distribution of generated data from Gaussian noise, i.e., $x = G_\theta(z)$, as has been mentioned before, z is sampled from a Gaussian noise distribution.

Proof: For a fixed generator G, D is trying to maximize the following objective:

$$\begin{aligned}
\mathcal{J} &= \mathbb{E}_{x\sim p_{data}}[\log D(x)] + \mathbb{E}_{z\sim p_{noise}}[\log(1 - D(G(z)))] && (1.4)\\
&= \int_x p_{data}(x)\log D(x)dx + \int_z p_{noise}(z)\log(1 - D(G(z)))dz && (1.5)\\
&= \mathcal{J}_1 + \mathcal{J}_2 && (1.6)
\end{aligned}$$

Claim 1.2 *For $\mathcal{J}_2$, we can also re-write it as*

$$\int_x p_G(x)\log(1 - D(x))dx \tag{1.7}$$

To prove the Claim, generally we have two methods. For rigorous representation, we express both of them. First, we may gain the result by setting $x = G(z)$.

Proof of [: Claim 1.2, method 1]
We have a noise distribution of z

$$z \sim p_{noise}(z) \tag{1.8}$$

with $x = G(z)$ and **assuming that $G(z)$ is an inversible function** then $z = G^{-1}(x)$. Therefore

$$dz = (G^{-1})'(x)dx \tag{1.9}$$

and

$$\begin{aligned}
p_G(x) &= \left[\int_{-\infty}^{G^{-1}(x)} p_{noise}(t)dt\right]' && (1.10)\\
&= p_{noise}(G^{-1}(x))(G^{-1})'(x) && (1.11)
\end{aligned}$$

Hence, we get the transformed result of $\mathcal{J}_2$:

$$\begin{aligned}
\mathcal{J}_2 &= \int_z p_{noise}(z)\log(1 - D(G(z)))dz && (1.12)\\
&= \int_x p_{noise}(G^{-1}(x))\log(1 - D(x))(G^{-1})'(x)dx && (1.13)\\
&= \int_x p_{G(x)}\log(1 - D(x))dx && (1.14)
\end{aligned}$$

■

However, it is a very special case where $x = G(z)$ is an inversible function, since $G(z)$ is often a non-parametric model(with no assumption of the data distribution and always a neural network in GANs) and we cannot sure its inversibility. Here we provide another method to prove the transformation of Claim 1.2.

Proof of [: Claim 1.2, method 2]
First, we define a deterministic mapping $G(\cdot) : \mathcal{Z} \to \mathcal{X}$ because of the fixed G. Then the transformed result of $\mathcal{J}_2$ can be:

$$\int_x p_G(x) \log(1 - D(x))dx = \int_x \left[\int_z p_G(x, z) \log(1 - D(x))dz \right] dx \tag{1.15}$$

$$= \int_x \left[\int_z p_z(z) p_G(x|z)dz \right] \log(1 - D(x))dx \tag{1.16}$$

Since $\mathcal{Z} \to \mathcal{X}$ is a deterministic mapping with generator G, thus

$$p_G(x|z) = \delta(x - G(z)) \tag{1.17}$$

where $\delta(\cdot)$ is a Dirac delta function. Then cont from Eq.1.16

$$\int_x \left[\int_z p_z(z) p_G(x|z)dz \right] \log(1 - D(x))dx = \int_x \left[\int_z p_z(z)\delta(x - G(z))dz \right] \log(1 - D(x))dx \tag{1.18}$$

$$= \int_z \left[\int_x \log(1 - D(x)) \cdot \delta(x - G(z))dx \right] p_z(z)dz \tag{1.19}$$

$$= \int_z \left[\log(1 - D(x)) * \delta(x - G(z)) \right] p_z(z)dz \tag{1.20}$$

$$= \int_z \log(1 - D(G(z)))p_z(z)dz \tag{1.21}$$

$$= \mathcal{J}_2 \tag{1.22}$$

where " $*$ " denotes the convolution operation. This is a more rigorous proof of Claim 1.2.

■

Hence, the whole objective function can be

$$\mathcal{J} = \mathcal{J}_1 + \mathcal{J}_2 \tag{1.23}$$

$$= \int_x p_{data}(x) \log D(x)dx + \int_x p_G(x) \log(1 - D(x))dx \tag{1.24}$$

$$= \int_x p_{data}(x) \log D(x) + p_G(x) \log(1 - D(x))dx \tag{1.25}$$

which is a form like

$$f(y) = a \log y + b \log(1 - y) \tag{1.26}$$

Set $f'(y) = 0$ we get

$$f'(y) = \frac{a}{y} - \frac{b}{1 - y} = 0 \tag{1.27}$$

holds for $y = \frac{a}{a+b}$ and $f(y)$ achieves its **maximum**. Therefore, the optimal discriminator is

$$D_{w^*} = \frac{p_{data}(x)}{p_{data}(x) + p_G(x)} \tag{1.28}$$

■

Given the optimal discriminator, we aim to train the generator model to achieve the global solution, in which case the discriminator cannot make correct classification anymore like mentioned before. The objective function then boils down to **minimizing** a cost function with variable G, parameterized by θ:

$$C(G) := \mathbb{E}_{x\sim p_{data}}[\log D_{w^*}] + \mathbb{E}_{x\sim p_G}[\log(1 - D_{w^*})] \tag{1.29}$$

$$= \mathbb{E}_{x\sim p_{data}}\left[\log \frac{p_{data}}{p_{data} + p_G}\right] + \mathbb{E}_{x\sim p_G}\left[\log \frac{p_G}{p_{data} + p_G}\right] \tag{1.30}$$

1.3 Global Solution of *C(G)*

Given the cost function $C(G)$ under condition of optimal discriminator D_{w^*}, we aim to minimize this objective function to reach a global solution, i.e., a powerful generator G_θ.

Firstly let's recap the ***K**ullback–**L**eibler Divergence*[5], which will be used for the later proof. The *KL-divergence*, also called *relative entropy*, is a measure of how one probability distribution is different from a second, reference probability distribution. The larger value of KL-divergence, the bigger difference between two distributions. The KL-divergence can be expressed as

$$KL(p(x)\|q(x)) = \mathbb{E}_{x\sim p}\left[\log \frac{p(x)}{q(x)}\right] \tag{1.31}$$

which is an asymmetrical measurement, that $KL(p(x)\|q(x)) \neq KL(q(x)\|p(x))$, as a remedy, this can also be replaced by *Jensen–Shannon divergence*[4]

$$JS(p(x)\|q(x)) = \frac{1}{2}KL\left(p(x)\|\frac{p(x)+q(x)}{2}\right) + \frac{1}{2}KL\left(q(x)\|\frac{p(x)+q(x)}{2}\right) \tag{1.32}$$

JS-divergence is apparently a symmetric formula.

Claim 1.3 *An important property of* KL-divergence *that it is non-negative, i.e.,*

$$KL(p(x)\|q(x)) \geq 0 \tag{1.33}$$

Proof: Assume that f is a convex function, from *Jensen's inequality* we have

$$\mathbb{E}[f(x)] \geq f[\mathbb{E}(x)] \tag{1.34}$$

Since $f(x) = -\log x$ is a convex function, set $\mathcal{T}(x) = \frac{q(x)}{p(x)}$ we have

$$\mathbb{E}_x\left[-\log(\mathcal{T}(x))\right] \geq -\log \mathbb{E}_x\left[\mathcal{T}(x)\right] \tag{1.35}$$

Substituting in the $f(x)$ with $\mathcal{T}(x)$

$$KL(p(x)\|q(x)) = \mathbb{E}_{x\sim p(x)}\left[-\log\left(\frac{q(x)}{p(x)}\right)\right] \tag{1.36}$$

$$\geq -\log \mathbb{E}_{x\sim p(x)}\left(\frac{q(x)}{p(x)}\right) \tag{1.37}$$

$$= -\log \int q(x)dx \tag{1.38}$$

$$= -\log(1) \tag{1.39}$$

$$= 0 \tag{1.40}$$

■

Then we can get closer to the global solution of generator G, given the optimal discriminator D_{w^*} obtained before.

Theorem 1.4 (Global Solution) *The global minimum of $C(G)$ is achieved if and only if*

$$p_G = p_{data} \tag{1.41}$$

which means the distribution of generated data from G is equal to that of the true data.

Proof:
Recall that $C(G)$ 1.30 equals to

$$C(G) = \mathbb{E}_{x\sim p_{data}}\left[\log\frac{p_{data}}{p_{data}+p_G}\right] + \mathbb{E}_{x\sim p_G}\left[\log\frac{p_G}{p_{data}+p_G}\right] \tag{1.42}$$

$$= \mathbb{E}_{x\sim p_{data}}\left[\log\frac{\frac{p_{data}}{2}}{\frac{p_{data}+p_G}{2}}\right] + \mathbb{E}_{x\sim p_G}\left[\log\frac{\frac{p_G}{2}}{\frac{p_{data}+p_G}{2}}\right] \tag{1.43}$$

$$= -\log 4 + \mathbb{E}_{x\sim p_{data}}\left[\log\frac{p_{data}}{\frac{p_{data}+p_G}{2}}\right] + \mathbb{E}_{x\sim p_G}\left[\log\frac{p_G}{\frac{p_{data}+p_G}{2}}\right] \tag{1.44}$$

$$= -\log 4 + KL\left(p_{data}\|\frac{p_{data}+p_G}{2}\right) + KL\left(p_G\|\frac{p_{data}+p_G}{2}\right) \tag{1.45}$$

$$= -\log 4 + 2JS(p_{data}\|p_G) \tag{1.46}$$

Therefore, when $p_G = p_{data}$, the minimum of $C(G)$, i.e., $-\log 4$, is achieved.

■

GAN is currently a very eye-catching research topic and can be applied in various work, such as image super-resolution, neural style or image in-painting. It is also a representative for min-max game in machine learning algorithm.

For better understanding of the training mechanism of GANs, we would like to introduce the figure from [2] as follows:

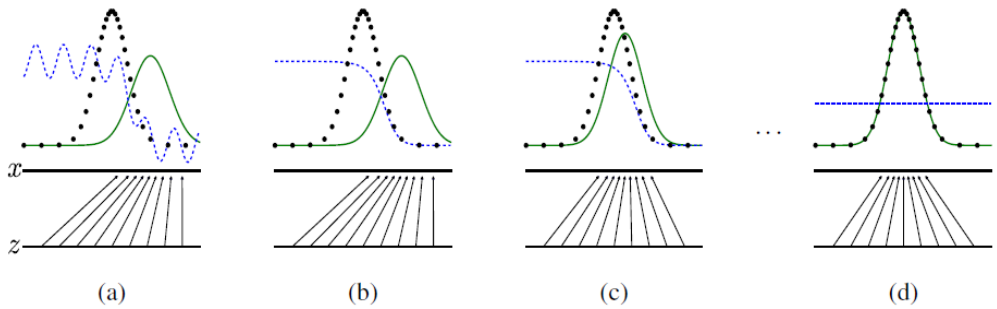

Figure 1: Generative adversarial nets are trained by simultaneously updating the discriminative distribution (D, blue, dashed line) so that it discriminates between samples from the data generating distribution (black, dotted line) p_{data} from those of the generative distribution p_G (green, solid line). The lower horizontal line is the noise domain from which z is sampled, in this case uniformly. The horizontal line above is part of the domain of x. The upward arrows show how the mapping $x = G(z)$ imposes the non-uniform distribution p_G on transformed samples. G contracts in regions of high density and expands in regions of low density of p_G.

The above four sub-figure can be interpreted as follows: (a) Consider an adversarial pair near convergence: p_G is similar to p_{data} and D is a partially accurate classifier. (b) In the inner loop of the algorithm D is trained to discriminate samples from data, converging to $D^*(x) = \frac{p_{data}(x)}{p_{data}(x)+p_G(x)}$. (c) After an update to G, gradient of D has guided $G(z)$ to flow to regions that are more likely to be classified as true data. (d) After several steps of training, if G and D have enough capacity, they will reach a point at which both cannot improve because $p_G = p_{data}$. The discriminator is unable to differentiate between the two distributions, i.e., $D(x) = \frac{1}{2}$.

2 Min-Max Optimization

The research and exploration of GAN motivate the study of min-max optimization, which is intrinsically harder than minimization. Min-max game optimization can be expressed by

Definition 2.1 (Min-Max Optimization) *For some continuous function f we want to solve*

$$\min_{x\in\mathcal{X}} \max_{y\in\mathcal{Y}} f(x, y) \tag{2.1}$$

Remarks

- *Domains are typically compact.*
- *In general the above problem may not have a solution or it is hard to train the models.*
- *There are guarantees when domains are compact and f is convex-concave, mentioned later.*

Claim 2.1 (Min-Max Inequality) *For min-max optimization, it is always true(no-necessary for $\boldsymbol{f}$ being convex-concave) that*

$$\inf_{x\in\mathcal{X}} \sup_{y\in\mathcal{Y}} f(x, y) \geq \sup_{y\in\mathcal{Y}} \inf_{x\in\mathcal{X}} f(x, y) \tag{2.2}$$

Proof: Define

$$g(z) \triangleq \inf_{w \in W} f(z, w) \tag{2.3}$$

easy to find

$$\forall w, \forall z, g(z) \leq f(z, w) \tag{2.4}$$

$$\Rightarrow \forall w, \sup_z g(z) \leq \sup_z f(z, w) \tag{2.5}$$

$$\Rightarrow \sup_z g(z) \leq \inf_w \sup_z f(z, w) \tag{2.6}$$

substituting in $g(z) = \inf_w f(z, w)$ we get

$$\sup_z \inf_w f(z, w) \leq \inf_w \sup_z f(z, w) \tag{2.7}$$

■

Theorem 2.2 (Min-Max by *John von Neumann*) *Let $\mathcal{X} \subset \mathbb{R}^n$ and $\mathcal{Y} \subset \mathbb{R}^n$ be compact convex sets. If* f *is a continuous function that is convex-concave it holds*

$$\min_{x \in \mathcal{X}} \max_{y \in \mathcal{Y}} f(x, y) = \max_{y \in \mathcal{Y}} \min_{x \in \mathcal{X}} f(x, y) \tag{2.8}$$

Remarks

- Many applications, especially in Game Theory.
- If $f = x^T A y$, and the domains are ∇_n, ∇_m, it captures classic zero sum games.
- $f(x, y)$ is the value of the game.

Proof of [: Theorem 2.2]
Firstly Let's recall the definition of *Online Gradient Descent* and use it as an example for the proof.

Definition 2.2 (Online Gradient Descent) . *Let $f : \mathbb{R}^n \to \mathbb{R}$ be convex function, differentiable and $L-$lipschitz in some compact convex set $\mathcal{X}$ of diameter D. Online GD is defined:*

Algorithm 1 Online Gradient Descent

Initialize at some x_0.
For t:= 1 to T do

- Choose x_t and observe $l_t(x_t)$
- $y_t = x_t - \alpha_t \nabla l_t(x_t)$
- $x_{t+1} = \Pi_{\mathcal{X}}(y_t)$

Regret: $\frac{1}{T}\left(\sum_{t=1}^T l_t(x_t) - \min_x \sum_{t=1}^T l_t(x_t)\right)$

The goal is to minimize the regret with error ϵ>0 such that

$$\frac{1}{T}\left(\sum_{t=1}^{T} l_t(x_t) - \min_x \sum_{t=1}^{T} l_t(x)\right) < \epsilon \tag{2.9}$$

which can then be called "No-regret".

In this proof, we use "No-regret" learning for both "players", i.e., function w.r.t. both variables.

Let $x_1, \ldots, x_T$ and $y_1, \ldots, y_T$ be the parameters that iterates as advised by some no-regret algorithm. Define $\hat{x} = \sum_{i=1}^{T} x_i$ and $\hat{y} = \frac{1}{T}\sum_{i=1}^{T} y_i$ and $T = \Theta\left(\frac{1}{\epsilon^2}\right)$. Since we want to **minimize** the objective function w.r.t. variable x, from "No-regret" property we choose any x and get that

$$\frac{1}{T}\sum_t f(x_t, y_t) \leq \frac{1}{T}\sum_t f(x, y_t) + \epsilon \tag{2.10}$$

$$\leq f(x, \hat{y}) + \epsilon \;\; \textit{by concavity} \tag{2.11}$$

Similarly, we want to **maximize** the objective function w.r.t. variable y, from "No-regret" property we choose any y and get that

$$\frac{1}{T}\sum_t f(x_t, y_t) \geq \frac{1}{T}\sum_t f(x_t, y) - \epsilon \tag{2.12}$$

$$\geq f(\hat{x}, y) - \epsilon \;\; \textit{by convexity} \tag{2.13}$$

Combine the inequalities 2.11 and 2.13 we have

$$f(\hat{x}, y) - 2\epsilon \leq f(x, \hat{y}) \tag{2.14}$$

Conclude for all x and y we have

$$\max_y f(\hat{x}, y) - 2\epsilon \leq \min_x f(x, \hat{y}) \tag{2.15}$$

Observe the right-hand side

$$\max_y \min_x f(x, y) \geq \min_x f(x, \hat{y}) \tag{2.16}$$

$$\geq \max_y f(\hat{x}, y) - 2\epsilon \tag{2.17}$$

$$\geq \min_x \max_y f(x, y) - 2\epsilon \tag{2.18}$$

then the theorem can be proved with $\epsilon \to 0$. ■

3 Last Iterate Convergence

Generally, the *convex-concave* settings with compact domains of function f are easy. However it is tough to optimize with GANs, in which

- Functions are not necessarily convex-concave
- Time averaging does not help, and *Jensen's inequality* is not applicable

Therefore we fail to care about the time-averaging case, this motivates us to notice *last iterate convergence*[3]. We focus on

$$\min_{x\in\mathcal{X}}\max_{y\in\mathcal{Y}} x^T A y \tag{3.1}$$

for the rest part.

3.1 Gradient Descent/Ascent(GDA)

We consider gradient descent for minimizing the loss function w.r.t. x and maximizing that with gradient ascent w.r.t. y, expressed as

$$x_{t+1} = x_t - \eta\nabla_x f(x_t, y_t) \tag{3.2}$$
$$y_{t+1} = y_t + \eta\nabla_y f(x_t, y_t) \tag{3.3}$$

With simplest case where $f(x, y) = xy$, and η is the step size. In this case GDA boils down to

$$x_{t+1} = x_t - \eta y_t \tag{3.4}$$
$$y_{t+1} = y_t + \eta x_t \tag{3.5}$$

Since when we consider the evolution trend of the x and y, we may imagine their changing of distance from the origin, i.e. $(0, 0)$. To better illustrate the convergence situation, we substitute the update formulas above in the example below.

Claim 3.1 (Divergence) *It holds that $x_t^2 + y_t^2$ is increasing in t.*

Proof: From GDA algorithm we have

$$x_{t+1} = x_t - \eta y_t \tag{3.6}$$
$$y_{t+1} = y_t + \eta x_t \tag{3.7}$$

then

$$x_{t+1}^2 + y_{t+1}^2 = (\eta^2 + 1)(x_t^2 + y_t^2) > x_t^2 + y_t^2 \tag{3.8}$$

which proves the divergence in this situation.

■

3.2 Multiplicative Weights Update Algorithm(MWUA)

The similar result can be obtained via MWUA:

$$x_i^{t+1} = \frac{x_i^t e^{-\eta(Ay^t)_i}}{Z_x} \tag{3.9}$$
$$y_j^{t+1} = \frac{y_j^t e^{\eta(A^\top x^t)_j}}{Z_y} \tag{3.10}$$

with a theorem pointing its divergence:

Theorem 3.2 (Divergence) *Assume there exists a unique fully mixed Nash (x^*, y^*) equilibrium (full support). It holds that the KL-divergence between player strategies the fully mixed Nash goes to infinity, i.e.,*

$$\lim_t KL(x^* \| x^t) = \infty \tag{3.11}$$

$$\lim_t KL(y^* \| y^t) = \infty \tag{3.12}$$

To solve this issue, we may consider some other algorithms and ideas to restrict the divergence. Next, we would like to consider more about min-max optimization and *Optimism Gradient Descent*!

3.3 Negative Momentum(Optimism)

Previously we discussed about the last iterate convergence but found that Gradient Descent Ascent(GDA) diverges even for the simplest form $x^\top Ay$. In this part, we start from the continuous GDA for the bi-linear form, then the objective function should be

$$\min_{x \in \mathbb{R}^n} \max_{y \in \mathbb{R}^m} x^\top Ay \tag{3.13}$$

Firstly let's recall the GDA formula of bilinear system:

$$x_{t+1} = x_t - \eta \nabla_x f(x_t, y_t) \tag{3.14}$$

$$y_{t+1} = y_t + \eta \nabla_y f(x_t, y_t) \tag{3.15}$$

Consider the continuous GDA that is the system of odes:

$$\frac{dx}{dt} = -\eta Ay \tag{3.16}$$

$$\frac{dy}{dt} = \eta A^\top x \tag{3.17}$$

then we also consider the cycles.

Lemma 3.3 (Cycles) *It holds that $\|x\|_2^2 + \|y\|_2^2$ is constant w.r.t* t.

Proof: Observe that

$$\frac{dx_i^2}{dt} = 2x_i \frac{dx_i}{dt} = -2\eta x_i (Ay)_i \tag{3.18}$$

$$\frac{dy_j^2}{dt} = 2y_j \frac{dy_j}{dt} = 2\eta y_j (A^\top x)_j \tag{3.19}$$

Hence, we get

$$\frac{d}{dt}\{\|x\|_2^2 + \|y\|_2^2\} = \sum_i \frac{\partial}{\partial t} x_i^2 + \sum_j \frac{\partial}{\partial t} y_j^2 \tag{3.20}$$

$$= -2\eta \sum_i x_i (Ay)_i + 2\eta \sum_j y_j (A^\top x)_j \tag{3.21}$$

$$= 0 \tag{3.22}$$

therefore it remains constant and fails to convergence.

■

We here to use optimism(or negative momentum since we step back) to try to fix this behavior. The update function can be expressed like this:

$$x_{t+1} = x_t - 2\eta\nabla_x f(x_t, y_t) + \eta\nabla_x f(x_{t-1}, y_{t-1}) \tag{3.23}$$

$$y_{t+1} = y_t + 2\eta\nabla_y f(x_t, y_t) - \eta\nabla_y f(x_{t-1}, y_{t-1}) \tag{3.24}$$

Intuitively, the optimism gradient descent can be interpreted as follows:

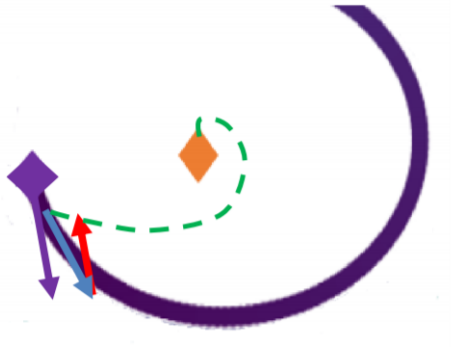

Figure 2: Interpretation of Optimism Gradient Descent

As we have showed before, if we do not constrain here, the objective function may remain constant or either divergence, as the purple circle shows. Then we use optimism to step back, trying to force the objective to go inside as the green dashed line. The orange small square is the theoretical optimal destination.

Formally, we can describe the theorem of OGDA(Optimism Gradient Descent Ascent) as follows.

Theorem 3.4 (Convergence) *Consider the bilinear game* $x^\top Ay$ *where* A *is full rank. Optimistic GDA converges point-wise and reaches an* ϵ *neighborhood in*

$$T := \Theta\left(\frac{\lambda_{max}(AA^\top)}{\lambda_{min}(AA^\top)}\log\frac{1}{\epsilon}\right) \tag{3.25}$$

by choosing learning rate $\eta = \frac{1}{4\sqrt{\lambda_{max}(AA^\top)}}$.

An intuitive demonstration of proof can be showed as follows.

Proof: We can treat the above optimism gradient descent procedure as a linear system, i.e.:

$$\begin{bmatrix} x_{t+1} \\ y_{t+1} \\ x_t \\ y_t \end{bmatrix} = \begin{bmatrix} I & -2\eta A & 0 & \eta A \\ 2\eta A^\top & I & -\eta A^\top & 0 \\ I & 0 & 0 & 0 \\ 0 & I & 0 & 0 \end{bmatrix} \begin{bmatrix} x_t \\ y_t \\ x_{t-1} \\ y_{t-1} \end{bmatrix} \tag{3.26}$$

by setting

$$M_{t+1} = \begin{bmatrix} x_{t+1} \\ y_{t+1} \\ x_t \\ y_t \end{bmatrix} \tag{3.27}$$

and

$$Q = \begin{bmatrix} I & -2\eta A & 0 & \eta A \\ 2\eta A^\top & I & -\eta A^\top & 0 \\ I & 0 & 0 & 0 \\ 0 & I & 0 & 0 \end{bmatrix} \tag{3.28}$$

we get the simplified form

$$M_{t+1} = QM_t \tag{3.29}$$

then

$$M_{t+n} = Q^n M_t \tag{3.30}$$

Obviously that coefficient matrix Q is full rank like that of A. From matrix decomposition we have

$$Q = S\Lambda S^{-1} \tag{3.31}$$

where matrix S is composed of eigenvectors placed by column of Q, and Λ is a diagonal matrix with eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_n$. Therefore

$$Q^n = S\Lambda^n S^{-1} \tag{3.32}$$

since S is a inversible matrix due to full rank of Q. The state equation can be described as

$$M_{t+n} = Q^n M_t \tag{3.33}$$

Then we can propose the following lemma:

Lemma 3.5 (Eigenvalues) *The coefficient matrix Q has eigenvalues with magnitude being all less than 1 by appropriate choice of η, which has been proposed in Theorem 3.4.*

With the above lemma we are sure that the state(objective function) M is going to convergence. ■

4 Min-max in bilinear constrained case

Consider the problem

$$\min_{x \in \nabla_n} \max_{y \in \nabla_m} x^\top A y \tag{4.1}$$

Let's do Optimistic Multiplicative Weights Update, i.e.,

$$x_i^{t+1} = x_i^t \frac{1 - 2\eta(Ay^t)_i + \eta(Ay^{t-1})_i}{\sum_j x_j^t (1 - 2\eta(Ay^t)_j + \eta(Ay^{t-1})_j)} \tag{4.2}$$

$$y_i^{t+1} = y_i^t \frac{1 + 2\eta(A^\top x^t)_i - \eta(A^\top x^{t-1})_i}{\sum_j y_j^t (1 + 2\eta(A^\top x^t)_j - \eta(A^\top x^{t-1})_j)} \tag{4.3}$$

We can then have the following theorem.

Theorem 4.1 (Convergence) *Let A be the payoff matrix of a zero sum game and the game has a unique Nash equilibrium. It holds that fot η sufficiently small(depends on* n, m, A*, η can be exponentially small in* n, m*), starting from uniform distribution* $\lim_{t\to\infty}(x^t, y^t) = (x^*, y^*)$ *under OMWU dynamics.*

5 Min-max in general settings

In this case, the above mentioned Min-max theorem is not applicable. We can instead solve the problem by relaxing the solution concept:

Definition 5.1 (Local Nash) *A critical point (x^*, y^*) is a local Nash if there exists a neighborhood $\mathcal{U}$ around (x^*, y^*) so that for all $(x, y) \in \mathcal{U}$ we have that*

$$f(x^*, y) \leq f(x^*, y^*) \leq f(x, y^*) \tag{5.1}$$

However, it is natural to ask: Does there always exist a local Nash? The answer is no. The following theorem points out its nature:

Theorem 5.1 (Local Convergence) *Under mild assumptions on $f(x, y)$ and step-size η we have*

$$Local\ Nash\ \subset\ GDA-stable\ \subset OGDA-stable \tag{5.2}$$

Remarks

- This is a local result.
- The inclusion relationship can be strict.

The second term of remarks directs to the following lemma:

Lemma 5.2 (Inclusion strict) *There are functions with critical points that are* GDA-stable *but not local Nash [1]. An example is*

$$f(x, y) = -\frac{1}{8}x^2 - \frac{1}{2}y^2 + \frac{6}{10}xy \tag{5.3}$$

Proof: Let

$$f(x, y) = -\frac{1}{8}x^2 - \frac{1}{2}y^2 + \frac{6}{10}xy \tag{5.4}$$

with α being the step size. The GDA update rule for this min-max game is:

$$x_{t+1} = x_t - \alpha\left(-\frac{1}{4}x_t + \frac{6}{10}y_t\right) \tag{5.5}$$

$$y_{t+1} = y_t + \alpha\left(-y_t + \frac{6}{10}x_t\right) \tag{5.6}$$

Computing the Jacobian Matrix of above GDA rule at coordinate $(0, 0)$

$$\mathcal{J}_{GDA} = \begin{bmatrix} 1 + \frac{1}{4}\alpha & -\frac{6}{10}\alpha \\ \frac{6}{10}\alpha & 1 - \alpha \end{bmatrix} \tag{5.7}$$

Both eigenvalues of $\mathcal{J}_{GDA}$ have magnitude less than 1 (for any $0<\alpha<\frac{1}{L}$ where $L \leq 1.34$). Hence, there exists a neighborhood $\mathcal{U}$ of $(0,0)$ so that for all $(x_0, y_0) \in \mathcal{U}$ we get that $\lim_t (x_t, y_t) = (0,0)$ for GDA update rules. However it is clear that (0, 0) is not a local min-max. See also the following figure for a pictorial illustration of the conclusion.

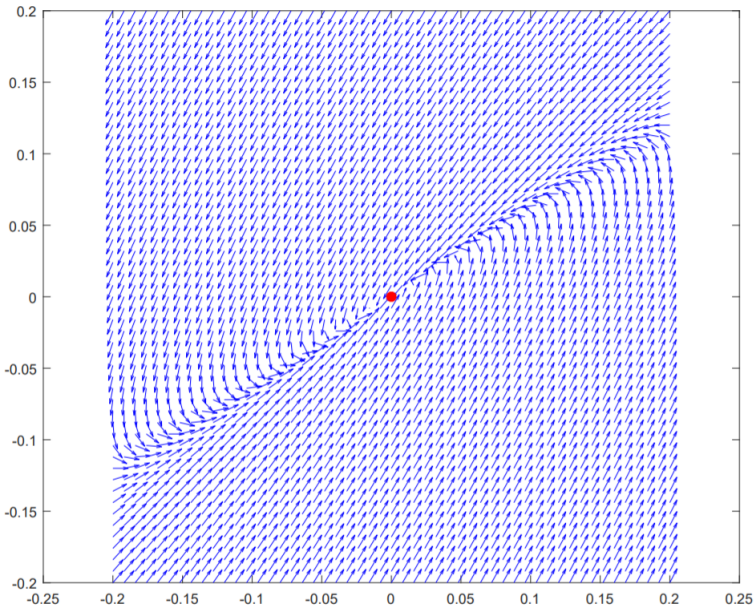

Figure 3: Function $f(x,y) = -\frac{1}{8}x^2 - \frac{1}{2}y^2 + \frac{6}{10}xy$ with $\alpha = 0.001$

The arrows point towards the next step of the Gradient Descent/Ascent dynamics. We can see that the system converges to $(0,0)$ point (GDA-stable), which is not a local min-max critical point(refer to the above definition) .

■

References

[1] Constantinos Daskalakis and Ioannis Panageas. The limit points of (optimistic) gradient descent in min-max optimization. In *Advances in Neural Information Processing Systems*, pages 9236–9246, 2018.

[2] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[3] Qi Lei, Sai Ganesh Nagarajan, Ioannis Panageas, and Xiao Wang. Last iterate convergence in no-regret learning: constrained min-max optimization for convex-concave landscapes. *arXiv preprint arXiv:2002.06768*, 2020.

[4] Wikipedia. Definition of Jensen–Shannon divergence. `https://en.wikipedia.org/wiki/Jensen%E2%80%93Shannon_divergence`, 2020.

[5] Wikipedia. Definition of Kullback–Leibler divergence. `https://en.wikipedia.org/wiki/Kullback%E2%80%93Leibler_divergence`, 2020.