1 Outline

In this lecture, we cover

- algorithms for minimax optimization,
- variational inequality.

2 More Algorithms for Minimax Optimization

We consider minimax optimization

$$\min_{x\in X}\max_{y\in Y}\phi(x,y)$$

where $X\subseteq\mathbb{R}^d$, $Y\subseteq\mathbb{R}^p$, and $\phi: X\times Y\to\mathbb{R}$. Recall that a point $(\bar{x},\bar{y})\in X\times Y$ is an ϵ-saddle point if

$$0\leq\max_{y\in Y}\phi(\bar{x},y)-\min_{x\in X}\phi(x,\bar{y})\leq\epsilon.$$

In the last lecture, we discussed Gradient Descent Ascent (GDA) for minimax optimization. In this lecture, we will cover more algorithms for minimax optimization.

We say that $\phi: X\times Y\to\mathbb{R}$ is **convex-concave** if $\phi: X\times Y\to\mathbb{R}$ is convex in x and concave in y. We say that $\phi: X\times Y\to\mathbb{R}$ is **L-Lipschitz continuous** if

$$|\phi(x_1,y_1)-\phi(x_2,y_2)|\leq L\|(x_1,y_1)-(x_2,y_2)\|_2\quad\forall (x_1,y_1),(x_2,y_2)\in X\times Y.$$

We say that ϕ is **α-strongly-convex-strongly-concave** if ϕ is α-strongly convex in x and α-strongly concave in y. The α-strong convexity means that for any fixed $y\in Y$, we have

$$\phi(x_1,y)\geq\phi(x_2,y)+\nabla_x\phi(x_2,y)^\top(x_1-x_2)+\frac{\alpha}{2}\|x_1-x_2\|_2^2,\quad\forall x_1,x_2\in X.$$

The α-strong concavity means that for any fixed $x\in X$, we have

$$-\phi(x,y_1)\geq-\phi(x,y_2)-\nabla_y\phi(x,y_2)^\top(y_1-y_2)+\frac{\alpha}{2}\|y_1-y_2\|_2^2,\quad\forall y_1,y_2\in Y.$$

Moreover, we say that ϕ is **β-smooth** if

$$|\nabla\phi(x_1,y_1)-\nabla\phi(x_2,y_2)|\leq\beta\|(x_1,y_1)-(x_2,y_2)\|_2\quad\forall (x_1,y_1),(x_2,y_2)\in X\times Y$$

where

$$\nabla\phi(x,y)=\left(\nabla_x\phi(x,y)^\top,\nabla_y\phi(x,y)^\top\right)^\top.$$

Algorithm 1 Gradient Descent Ascent

Initialize $x_1 \in X$ and $y_1 \in Y$.
for $t = 1, \ldots, T$ **do**
  Take a step size $\eta_t > 0$
  Update $x_{t+1} = \text{proj}_X(x_t - \eta_t \nabla_x \phi(x_t, y_t))$
  Update $y_{t+1} = \text{proj}_Y(y_t + \eta_t \nabla_y \phi(x_t, y_t))$
end for

2.1 Gradient Descent Ascent Revisited

Assuming that ϕ is differentiable, GDA works as in Algorithm 1. The following is a convergence guarantee for Algorithm 1 for Lipschitz continuous functions.

Theorem 14.1. *Let $\phi : X \times Y \to \mathbb{R}$ be a L-Lipschitz continuous convex-concave function. Assume that $\|x_1 - x_2\|_2 \leq R$ for any $x_1, x_2 \in X$ and $\|y_1 - y_2\|_2 \leq R$ for any $y_1, y_2 \in Y$. Then Algorithm 1 with step size $\eta = R/L\sqrt{T}$ guarantees that for any $(x, y) \in X \times Y$,*

$$\phi\left(\frac{1}{T}\sum_{t=1}^{t} x_t, y\right) - \phi\left(x, \frac{1}{T}\sum_{t=1}^{t} y_t\right) \leq \frac{2LR}{\sqrt{T}}.$$

Theorem 14.1 implies that GDA finds an ϵ-saddle point after $O(1/\epsilon^2)$ iterations for a Lipschitz continuous convex-concave function.

Recall that gradient descent can have a faster convergence for smooth functions than the case of Lipschitz continuous functions. In fact, that is not the case for GDA. The following provides a convergence guarantee for the case of smooth functions.

Theorem 14.2. *Let $\phi : X \times Y \to \mathbb{R}$ be a β-smooth convex-concave function. Assume that $\|x_1 - x_2\|_2 \leq R$ for any $x_1, x_2 \in X$ and $\|y_1 - y_2\|_2 \leq R$ for any $y_1, y_2 \in Y$. Then Algorithm 1 with step size $\eta = R/L\sqrt{T}$ where*

$$L = 2\beta R + \|\nabla \phi(x_1, y_1)\|_2$$

guarantees that for any $(x, y) \in X \times Y$,

$$\phi\left(\frac{1}{T}\sum_{t=1}^{t} x_t, y\right) - \phi\left(x, \frac{1}{T}\sum_{t=1}^{t} y_t\right) \leq \frac{2LR}{\sqrt{T}}.$$

Proof. As ϕ is β-smooth, we have that

$$\begin{aligned}
\|\nabla \phi(x, y)\|_2 &\leq \|\nabla \phi(x_1, y_1)\|_2 + \beta\|(x, y) - (x_1, y_1)\|_2 \\
&\leq \|\nabla \phi(x_1, y_1)\|_2 + \beta\|x - x_1\|_2 + \beta\|y - y_1\|_2 \\
&\leq L.
\end{aligned}$$

This implies that ϕ is L-Lipschitz continuous. Then the result follows from Theorem 14.1. $\square$

What is perhaps more surprising is that the convergence rate for smooth functions given by Theorem 14.2 is tight. In contrast, gradient descent applied to smooth convex minimization guarantees a rate of $O(1/T)$.

Moreover, if we further assume that ϕ is α-strongly-convex-strongly-concave, then GDA can converge at a exponentially fast rate.

Theorem 14.3. *Let $\phi : X \times Y \to \mathbb{R}$ be a β-smooth α-strongly-convex-strongly-concave function. Let κ denote the condition number $\kappa = \beta/\alpha$. Then Algorithm 1 with step size $\eta = \alpha/\beta^2$ guarantees that for any $(x, y) \in X \times Y$,*

$$\phi(x_t, y) - \phi(x, y_t) \leq \beta\kappa \left(1 - \frac{1}{\kappa^2}\right)^t \|(x_1, y_1) - (x, y)\|_2^2 \quad \forall t \geq 1.$$

Theorem 14.3 implies that GDA provides a convergence rate of $O(\kappa^2 \log(1/\epsilon))$.

2.2 Extra Gradient Method

In the previous subsection, we considered GDA and its performance for structured functions. A simple modification to GDA is shown to provide improved performances. The **extra gradient (EG)** method works as follows. Given $(x_t, y_t) \in X \times Y$, we apply

$$\begin{aligned}
x_{t+\frac{1}{2}} &= \mathrm{proj}_X\left(x_t - \eta \nabla_x \phi(x_t, y_t)\right), \\
y_{t+\frac{1}{2}} &= \mathrm{proj}_Y\left(y_t + \eta \nabla_y \phi(x_t, y_t)\right), \\
x_{t+1} &= \mathrm{proj}_X\left(x_t - \eta \nabla_x \phi\left(x_{t+\frac{1}{2}}, y_{t+\frac{1}{2}}\right)\right), \\
y_{t+1} &= \mathrm{proj}_Y\left(y_t + \eta \nabla_y \phi\left(x_{t+\frac{1}{2}}, y_{t+\frac{1}{2}}\right)\right).
\end{aligned}$$

Basically, to obtain (x_{t+1}, y_{t+1}) from (x_t, y_t), we compute the gradient $\nabla\phi(x_t, y_t)$ and the gradient $\nabla\phi\left(x_{t+\frac{1}{2}}, y_{t+\frac{1}{2}}\right)$.

Theorem 14.4. *Let $\phi : X \times Y \to \mathbb{R}$ be a β-smooth convex-concave function. Assume that $\|x_1 - x_2\|_2 \leq R$ for any $x_1, x_2 \in X$ and $\|y_1 - y_2\|_2 \leq R$ for any $y_1, y_2 \in Y$. Then the extra gradient method with step size $\eta = 1/\beta$ guarantees that for any $(x, y) \in X \times Y$,*

$$\phi\left(\frac{1}{T}\sum_{t=1}^{t} x_{t+\frac{1}{2}}, y\right) - \phi\left(x, \frac{1}{T}\sum_{t=1}^{t} y_{t+\frac{1}{2}}\right) \leq \frac{\beta R^2}{2T}.$$

Here, we use a constant step size $\eta = 1/\beta$ for EG, which provides a convergence rate of $O(1/T)$. It is proved that the rate $O(1/T)$ is optimal [OX21]. Furthermore, if we further assume that ϕ is strongly-convex-strongly-concave, then we get the following improved guarantee.

Theorem 14.5. *Let $\phi : X \times Y \to \mathbb{R}$ be a β-smooth α-strongly-convex-strongly-concave function. Then the extra gradient method with step size $\eta = 1/4\beta$ guarantees that for any $(x, y) \in X \times Y$,*

$$\phi(x_t, y) - \phi(x, y_t) \leq \beta\kappa \left(1 - \frac{1}{4\kappa}\right)^t \|(x_1, y_1) - (x, y)\|_2^2 \quad \forall t \geq 1.$$

Theorem 14.5 implies that EG provides a convergence rate of $O(\kappa \log(1/\epsilon))$. Here, the rate of order $O(\kappa \log(1/\epsilon))$ is optimal [ZHZ22].

2.3 Optimistic GDA

The next algorithm is referred to as **Optimistic Gradient Descent Ascent (OGDA)**. The algorithm proceeds with the following update rule.

$$
\begin{aligned}
x_{t+\frac{1}{2}} &= \mathrm{proj}_X\left(x_t - \eta \nabla_x \phi\left(x_{t-\frac{1}{2}}, y_{t-\frac{1}{2}}\right)\right), \\
y_{t+\frac{1}{2}} &= \mathrm{proj}_Y\left(y_t + \eta \nabla_y \phi\left(x_{t-\frac{1}{2}}, y_{t-\frac{1}{2}}\right)\right), \\
x_{t+1} &= \mathrm{proj}_X\left(x_t - \eta \nabla_x \phi\left(x_{t+\frac{1}{2}}, y_{t+\frac{1}{2}}\right)\right), \\
y_{t+1} &= \mathrm{proj}_Y\left(y_t + \eta \nabla_y \phi\left(x_{t+\frac{1}{2}}, y_{t+\frac{1}{2}}\right)\right).
\end{aligned}
$$

The algorithm can be equivalently expressed in terms of the following update rule.

$$
\begin{aligned}
x_{t+1} &= x_t - 2\eta \nabla_x \phi(x_t, y_t) + \eta \nabla_x \phi(x_{t-1}, y_{t-1}), \\
y_{t+1} &= y_t - 2\eta \nabla_y \phi(x_t, y_t) + \eta \nabla_y \phi(x_{t-1}, y_{t-1}).
\end{aligned}
$$

OGDA provides a convergence rate of $O(1/\epsilon)$ for smooth functions and a rate of $O(\kappa \log(1/\epsilon))$ for smooth and strongly-convex-strongly-concave functions.

2.4 Proximal Point Algorithm

Proximal Point Algorithm (PPA) computes (x_{t+1}, y_{t+1}) as the unique solution to the following minimax optimization problem,

$$
\min_{x \in X} \max_{y \in Y} \quad \phi(x, y) + \frac{1}{2\eta}\|x - x_t\|_2^2 - \frac{1}{2}\|y - y_t\|_2^2.
$$

PPA also guarantees a rate of $O(1/\epsilon)$ for the smooth case [MOP20]. We may argue that the update rule of PPA is equivalent to

$$
\begin{aligned}
x_{t+1} &= \mathrm{proj}_X(x_t - \eta \nabla_x \phi(x_{t+1}, y_{t+1})), \\
y_{t+1} &= \mathrm{proj}_Y(y_t + \eta \nabla_y \phi(x_{t+1}, y_{t+1})).
\end{aligned}
$$

Here, to follow directly this update rule, we need to compute $\nabla \phi(x_{t+1}, y_{t+1})$. In this regard, EG and OGDA can be interpreted as some approximate versions of PPA:

$$
\begin{aligned}
x_{t+1} &= \mathrm{proj}_X(x_t - \eta \nabla_x \phi(x_{t+1}, y_{t+1}) + \epsilon_t^x), \\
y_{t+1} &= \mathrm{proj}_Y(y_t + \eta \nabla_y \phi(x_{t+1}, y_{t+1}) + \epsilon_t^y)
\end{aligned}
$$

where

- for EG, we have

$$
\begin{aligned}
\epsilon_t^x &= \eta\left(\nabla_x \phi(x_{t+1}, y_{t+1}) - \nabla_x \phi\left(x_{t+\frac{1}{2}}, y_{t+\frac{1}{2}}\right)\right), \\
\epsilon_t^y &= \eta\left(\nabla_y \phi(x_{t+1}, y_{t+1}) - \nabla_y \phi\left(x_{t+\frac{1}{2}}, y_{t+\frac{1}{2}}\right)\right),
\end{aligned}
$$

- for OGDA, we have

$$
\begin{aligned}
\epsilon_t^x &= \eta\left(\nabla_x \phi(x_{t+1}, y_{t+1}) - 2\nabla_x \phi\left(x_t, y_t\right) + \nabla_x \phi\left(x_{t-1}, y_{t-1}\right)\right), \\
\epsilon_t^y &= \eta\left(\nabla_y \phi(x_{t+1}, y_{t+1}) - 2\nabla_y \phi\left(x_t, y_t\right) + \nabla_y \phi\left(x_{t-1}, y_{t-1}\right)\right).
\end{aligned}
$$

3 Variational Inequalities

Consider the problem of minimizing a convex function f over a domain X. Recall that x^* is an optimal solution if and only if

$$\nabla f(x^*)^\top (x - x^*) \geq 0 \quad \forall x \in X.$$

We can generalize this to minimax optimization. For the minimiax optimization of ϕ over $X \times Y$, (x^*, y^*) is a saddle point if and only if

$$\nabla_x \phi(x^*, y^*)^\top (x - x^*) - \nabla_y \phi(x^*, y^*)^\top (y - y^*) \geq 0 \quad \forall (x, y) \in X \times Y.$$

For convex minimization, we can define the gradient operator $F = \nabla f$. Then the optimality condition becomes

$$F(x^*)^\top (x - x^*) \geq 0 \quad \forall x \in X.$$

For minimax optimization, we define the operator

$$F(x, y) = \begin{bmatrix} \nabla_x \phi(x, y) \\ -\nabla_y \phi(x, y) \end{bmatrix}.$$

Then the condition for a saddle point can be equivalently written as

$$F(x^*, y^*)^\top \left((x, y) - (x^*, y^*)\right) \geq 0 \quad \forall (x, y) \in X \times Y.$$

In general, given a domain $Z \subseteq \mathbb{R}^d$ and an operator $F : Z \to \mathbb{R}^d$, the **variational inequality problem** is to find a solution $z^* \in Z$ such that

$$F(z^*)^\top (z - z^*) \geq 0 \quad \forall z \in Z.$$

We say that operator F is **monotone** if

$$(F(u) - F(v))^\top (u - v) \geq 0 \quad \forall u, v \in Z.$$

Note that for a convex function f, the gradient operator $F = \nabla f$ is monotone. We may also prove that for a convex-concave function ϕ, the associated operator $F = [\nabla_x \phi^\top, -\nabla_y \phi^\top]^\top$ is monotone as well. Furthermore, we say that operator F is **β-Lipschitz continuous** if

$$\|F(u) - F(v)\|_2 \leq \beta \|u - v\|_2 \quad \forall u, v \in Z.$$

For convex minimization and minimax optimization, the β-Lipschitz continuity of the associated operators is equivalent to β-smoothness.

EG converges to a solution of the variational inequality problem with a rate of $O(1/T)$ [Nem04]. The algorithm is also referred to as **Mirror-Prox**. One may come up with an accelerated version of Mirror-Prox [CLO17].

References

[CLO17] Yunmei Chen, Guanghui Lan, and Yuyuan Ouyang. Accelerated schemes for a class of variational inequalities. *Mathematical Programming*, 165(1):113–149, 2017. 3

[MOP20] Aryan Mokhtari, Asuman Ozdaglar, and Sarath Pattathil. A unified analysis of extra-gradient and optimistic gradient methods for saddle point problems: Proximal point approach. In Silvia Chiappa and Roberto Calandra, editors, *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, volume 108 of *Proceedings of Machine Learning Research*, pages 1497–1507. PMLR, 26–28 Aug 2020. 2.4

[Nem04] Arkadi Nemirovski. Prox-method with rate of convergence o(1/t) for variational inequalities with lipschitz continuous monotone operators and smooth convex-concave saddle point problems. *SIAM Journal on Optimization*, 15(1):229–251, 2004. 3

[OX21] Yuyuan Ouyang and Yangyang Xu. Lower complexity bounds of first-order methods for convex-concave bilinear saddle-point problems. *Mathematical Programming*, 185(1):1–35, 2021. 2.2

[ZHZ22] Junyu Zhang, Mingyi Hong, and Shuzhong Zhang. On lower iteration complexity bounds for the convex concave saddle point problems. *Mathematical Programming*, 194(1):901–935, 2022. 2.2