DSC 211 Introduction to Optimization Winter 2024 Instructor: Jun-Kun Wang
Scribe: Cecilia Liu, Har Simrat Singh January 23, 2024
Editors/TAs: Merlin Chang, Marialena Sfyraki

Lecture 5: Projected Gradient Descent and Frank-Wolfe Method

1 Preliminary:

Constrained Convex Optimization: Given a convex function $f : C \to \mathbb{R}$, the task of constrained optimization entails solving the following problem:

$$\min_{x\in C} f(x), \quad \text{where } C \subset \mathbb{R}^d \text{ is a convex set.}$$

Similarly, the problem of unconstrained optimization can be defined as:

$$\min_{x\in\mathbb{R}^d} f(x)$$

In the unconstrained optimization case, if $f(\cdot)$ is convex, and x_* is the optimal point, i.e. $x_* = \arg\min_{x\in\mathbb{R}^d} f(x)$, then by the first-order optimality condition (for interior points) we have that $\nabla f(x_*) = 0$. In other words, if the gradient of a function vanishes at a point, the point is a candidate for a local minimizer of f. Observe that in the constrained optimization problem case, the optimizer x_* might be a boundary point, for which $\nabla f(x_*) \neq 0$. Therefore, we will introduce an optimiality condition using the subgradient.
Question. What are the optimality properties of the optimal point of a convex constrained optimization?

Definition 1. ***(Subgradient)*** *For a convex and not necessarily differentiable function $f(\cdot)$, defined over a set C, we say g_x is a subgradient of $f(\cdot)$ at $x \in C$, if for any $y \in C$ we have*

$$f(y) \geq f(x) + \langle g_x, y - x\rangle.$$

The set of g_x is called the sub-differential, denoted as $\partial f(x)$.

Remark: The set $\partial f(x)$ is a convex set.

Example: Let $f : \mathbb{R} \to \mathbb{R}$ be defined by $f(x) = |x|$. Observe that f is convex but not differentiable at $x = 0$. If g_0 is the subgradient at $x = 0$, then,

$$|y| \geq 0 + \langle g_0, y - 0\rangle \Rightarrow g_0 \in [-1, 1]$$

Using the subgradient, we have the following optimality condition for convex-constrained optimization problems:

Theorem 1. ***(Optimality condition for convex-constrained optimization:)*** *Assume f is a convex function, then x_* is a global optimal solution, where*

$$x_* = \arg\min_{x \in C} f(x)$$

if and only if there exists a subgradient $g_{x_*} \in \partial f(x_*)$ such that for any $y \in C$

$$\langle g_{x_*}, y - x_* \rangle \geq 0$$

Proof. We present the proof for one direction ("$\Leftarrow$").
Let $y \in C$. Then,

$$\begin{aligned} f(y) &\geq f(x_*) + \langle g_{x_*}, y - x_* \rangle && \text{(by definition of the subgradient)} \\ &\geq f(x_*) && \text{(by assumption } \langle g_{x_*}, y - x_* \rangle \geq 0\text{)} \end{aligned}$$

Since this holds for all $y \in C$ we have that,

$$x_* = \arg\min_{x \in C} f(x).$$

□

The proof for the other direction can be found in [1], under Theorem 2.4.11

2 Projected Gradient Descent

Projected gradient descent proceeds very similar to gradient descent, where we optimize following the descent direction. If at any iteration, we are outside of the constrained set C, then we will project x_{k+1} back into the set C. The projection with respect to the Euclidean norm of y onto set C is defined as

$$\mathbf{Proj}_C(\mathbf{y}) := \arg\min_{\mathbf{x} \in C} ||\mathbf{y} - \mathbf{x}||_2$$

In orther words, the projection of y onto C is equivalent to finding the point in C with the minimum Euclidean distance to x.
A formal statement of the PGD algorithm is given below.

Algorithm 1 Projected Gradient Descent

```
for $k = 1, 2, \dots$ do
    $\mathbf{x_{k+1}} = \mathbf{Proj}_C[\mathbf{x_k} - \eta \nabla f(\mathbf{x_k})]$
end for
```

or

Algorithm 2 Projected Gradient Descent (Rewrite)

```
for $k = 1, 2, \ldots$ do
    $\mathbf{y_{k+1}} = \mathbf{x_k} - \eta \nabla f(\mathbf{x_k})$    (Gradient Descent step)
    $\mathbf{x_{k+1}} = \mathbf{Proj}_C(\mathbf{y_{k+1}})$    (Projection)
end for
```

2.1 Convergence Rate of GD and PGD:

Let K be the number of iterations. Then, the convergence rates of

- GD for $\min_{x \in \mathbb{R}^d} f(x)$:

ϵ-Optimality Gap: $f(x_K) - \min_{x \in \mathbb{R}^d} f(x) \leq \epsilon$	
L-smooth convex	$O\left(\frac{L}{K}\right)$
L-smooth and μ-strongly convex	$O(\exp(-\frac{\mu}{L}K))$

- PGD for $\min_{x \in C} f(x)$:

ϵ-Optimality Gap: $f(x_K) - \min_{x \in C} f(x) \leq \epsilon$	
L-smooth convex	$O\left(\frac{L}{K}\right)$
L-smooth and μ-strongly convex	$O(\exp(-\frac{\mu}{L}K))$

2.2 When to Choose PGD?

The projection step needed in PGD is in and of itself an optimization problem. Thus, when the $\mathbf{Proj}_C(\mathbf{y_{k+1}})$ has a closed-form solution or there exists an efficient algorithm to solve it, we can choose PGD to solve constrained problems.

2.3 How to Implement the Projection

Suppose we want to solve the following problem

$$\hat{\mathbf{x}} = \mathbf{Proj}_C(\mathbf{y}) := \arg\min_{\mathbf{x} \in C} \|\mathbf{y} - \mathbf{x}\|_2$$

Example 1: (With closed-form solution)

Let $C := \{\mathbf{x} \in \mathbb{R}^d : ||\mathbf{x}||_2 \leq 1\}$. Then,

$$\hat{\mathbf{x}} = \begin{cases} \frac{\mathbf{y}}{||\mathbf{y}||_2} & \text{, if } \mathbf{y} \notin C \\ y & \text{, otherwise} \end{cases}$$

Example 2: (With closed-form solution)

Let $C := \{\mathbf{x} \in \mathbb{R}^d : ||\mathbf{x}||_\infty \leq 1\}$, with $||\mathbf{x}||_\infty := \max_i |\mathbf{x}[i]|$. This implies that $\forall x \in C$ we have that $\forall i \in [d], -1 \leq \mathbf{x}[i] \leq 1$. Now, we have the following cases

- If $\mathbf{y} \notin C$:

$$\hat{\mathbf{x}}[i] = \begin{cases} -1 & \text{, if } \mathbf{y}[i] \leq -1 \\ 1 & \text{, if } \mathbf{y}[i] \geq 1 \\ \mathbf{y}[i] & \text{, otherwise} \end{cases}$$

- If $\mathbf{y} \in C$:

$$\hat{\mathbf{x}} = \mathbf{y}$$

Example 3: (Without closed-form solution)

Let $C := \{\mathbf{x} \in \mathbb{R}^d : ||\mathbf{x}||_1 \leq 1\}$. By definition, $(\mathbf{z})_+ \triangleq \max\{0, \mathbf{z}\}$.

If $\mathbf{y} \notin C$,

$$\hat{\mathbf{x}}[i] = \text{sign}(\mathbf{y}[i]) \, (|\mathbf{y}[i]| - \lambda)_+,$$

where λ is the solution to $\sum_{i=1}^d (|\mathbf{y}[i]| - \lambda)_+ = 1$.

Later, we will know how to derive the solution using duality theory and KKT conditions.

Remark: Depending on the application, it may or may not be necessary to have constraints when solving optimization problems. For example, in machine learning, it's common to add a regularization to the original objective function; however, for domains like economics or operational research, it's important to take constraints into consideration.

3 Frank-Wolfe Method

Frank-Wolfe(FW) method also known as conditional gradient method, is an alternative to projected gradient descent. It could be applied in scenarios where the projection is computationally inefficient to calculate. Below is a formal statement of the Frank-Wolfe method algorithm.

Algorithm 3 Frank-Wolfe method

```
initialize a starting point $\mathbf{x_1} \in C$.
for $k = 1, 2, \dots$ do
    $\mathbf{v_k} = \arg\min_{\mathbf{v} \in C} \langle \mathbf{v}, \nabla f(\mathbf{x_k}) \rangle$  (Linear optimization)
    $\mathbf{x_{k+1}} = (1 - \eta_k)\mathbf{x_k} + \eta_k \mathbf{v_k}$ where $\eta_k \in [0, 1]$
end for
```

Compared to projected gradient descent rather than taking a gradient step and then projecting onto the convex constraint set, the Frank-Wolfe method optimizes an objective defined by the gradient inside the convex set. Since $\mathbf{x_{k+1}}$ is a convex combination of $\mathbf{x_k}$ and $\mathbf{v_k}$ in the convex set C, we know $\mathbf{x_{k+1}} \in C$. (This statement can be proved by induction.)

3.1 Convergence analysis of FW Method:

Theorem 2. *Assume $f(\cdot)$ is a L-smooth convex function. Denote D as the diameter of the set C. Let $\eta_K = \min\{1, \frac{2}{K}\} \in [0, 1]$, then FW achieves:*

$$f(x_K) - f(x_*) \leq \frac{2LD^2}{K} \approx O\left(\frac{1}{K}\right).$$

Remark. We have that $||x - y|| \leq D = \text{diam } C, \ \forall x, y \in C$.

Proof. By L-Smoothness we have,

$$f(x_{K+1}) \leq f(x_K) + \langle \nabla f(x_K), x_{K+1} - x_K \rangle + \frac{L}{2} ||x_{K+1} - x_K||^2.$$

Additionally, by the update rule of FW, we have

$$x_{K+1} - x_K = \eta_k (v_K - x_K).$$

Substituting in the above equation, we get

$$f(x_{K+1}) \leq f(x_K) + \eta_K \langle \nabla f(x_K), v_K - x_K \rangle + \frac{L\eta_K^2}{2} ||v_K - x_K||^2.$$

Since

$$||v_K - x_K||^2 \leq D^2$$

we have

$$f(x_{K+1}) \leq f(x_K) + \eta_K \langle \nabla f(x_K), v_K - x_K \rangle + \frac{L\eta_K^2}{2} D^2.$$

Now, since

$$v_K = \arg\min_{v \in C} \langle \nabla f(x_K), v \rangle,$$

we have

$$\langle \nabla f(x_K), v_K \rangle \leq \langle \nabla f(x_K), x_* \rangle.$$

Thus,

$$f(x_{K+1}) \leq f(x_K) + \eta_K \langle \nabla f(x_K), x_* - x_K \rangle + \frac{L\eta_K^2}{2} D^2.$$

By the definition of convexity,

$$f(x_*) - f(x_K) \geq \langle \nabla f(x_K), x_* - x_K \rangle,$$

we have,

$$f(x_{K+1}) \leq f(x_K) + \eta_k (f(x_*) - f(x_K)) + \frac{L\eta_K^2}{2} D^2.$$

Denoting the minimum value and the optimality gap respectively as,

$$f_* := \min_{x \in C} f(x) \quad \text{and } \delta_{K+1} := f(x_{K+1}) - f_*,$$

and substituting in the equation we get,

$$\delta_{K+1} \leq (1 - \eta_K)\delta_K + \frac{L\eta_K^2 D^2}{2}. \tag{1}$$

Lemma 1. *Let $\{\delta_K\}$ be a sequence that satisfies the recurrence*

$$\delta_{K+1} \leq \delta_K (1 - \eta_K) + \eta_K^2 c.$$

Then taking $\eta = \min\{1, \frac{2}{K}\}$, we get

$$\delta_K \leq \frac{4c}{K}.$$

For equation 1, taking

$$c = \frac{LD^2}{2},$$

we get

$$\delta_K \leq \frac{4c}{K} = \frac{2LD^2}{K}.$$

Hence, proved. □

The proof of the lemma 1 can be found in the book [2] in Chapter 9.

3.2 Applications of Frank-Wolfe Method - Matrix Completion

Definition 2. *(**Nuclear Norm:**) The nuclear norm of a matrix A denoted as $||A||_*$ is defined as the sum of all singular values of the the matrix.*

$$||A||_* = \sum_i \sigma_i(A)$$

By the singular value decomposition, if $A = U\Sigma V^T$, then

$$\Sigma = \begin{bmatrix} \sigma_1(A) & & \\ & \sigma_2(A) & \\ & & \ddots \end{bmatrix}$$

The matrix completion problem is defined as - given a rating matrix $M \in \mathbb{R}^{m\times n}$ which has some observed value for each (i, j). We define another matrix X which is partially filled with 0s. We need to find an optimum X over an $r-$nuclear norm ball such that it closely approximates M. This is defined by the objective:

$$\min_{X \in R^{m\times n} : ||X||_* \leq r} f(X)$$
$$\text{where } f(X) := \frac{1}{2}||X - P_O(M)||_2^2$$

Here $P_O(M) = O \odot M$ where O is a matrix of binary entries. The operation $P_O(M)$ gives us a matrix of observable entries as defined by O while the rest being 0, that is

$$P_O(M)_{i,j} = \begin{cases} M_{i,j} & \text{if } (i,j) \text{ is an observed entry} \\ 0 & \text{otherwise.} \end{cases}$$

Calculating the gradient of f

$$\nabla f(X) = X - P_O(M) \in \mathbb{R}^{m\times n}.$$

Applying the FW method to this objective, the linear optimization step is

$$\mathbf{v_k} = \arg\min_{||\mathbf{v}||_* \leq r} \langle \nabla f(X_k), \mathbf{v} \rangle.$$

The minimum of this objective can be obtained as:

$$\mathbf{v_k} = -r\mathbf{u_1}\mathbf{w_1}^T,$$

where $\mathbf{u_1}$ and $\mathbf{w_1}$ are the top left singular vector and the top right singular vector of $\nabla f(X)$, respectively, which can be obtained easily via techniques such as power method.

This means that the run time of the FW method for the matrix completion problem only involves computing the first left and right singular vectors of $\nabla f(X)$ and thus follows - $\tilde{\mathrm{O}}(m \times n)$ (quadratic) time complexity per iteration. Comparing this with PGD. In PGD, the projection step will be defined as

$$\mathbf{x_{k+1}} = Proj_{||\mathbf{v}||_* \leq r}(\mathbf{y_{k+1}}).$$

This objective would, however, require us to calculate the SVD of $\nabla f(X)$. Thus the time complexity of each iteration would be of the order $\tilde{\mathrm{O}}(m \times n \times \min(m, n))$ (cubic). The actual running cost of an algorithm can be computed as the total number of iterations required to reach an ϵ optimality gap (called the iteration complexity) times the iteration cost. Because of the cheap iteration cost compared to PGD, for the task of matrix completion with a large matrix size, FW would be preferred over PGD, even though the convergence rate of FW is not better than PGD.

3.3 Convergence Analysis when $f(\cdot)$ is smooth and strongly convex

Section 3.1 describes the behaviour of the FW method when $f(\cdot)$ is L-smooth and convex. We observed a convergence rate of $O(1/K)$. Now, if the function is smooth and strongly convex how does this convergence rate change, do we observe a faster rate than $O(1/K)$? The answer is that it depends (on the constraint set C).

Negative Example: If C is a probability simplex defined as

$$C := \{x \in \mathbb{R}^d : \sum_{i=1}^{d} x[i] = 1, x[i] \geq 0\}$$

then there exists a strongly convex smooth function $f(\cdot)$ such that FW method converges to a ϵ optimality gap in at least K iterations, where

$$K = \Omega\Bigg(\max\left(\frac{L}{\epsilon}, \frac{d}{2}\right)\Bigg).$$

This implies $\epsilon = O(1/K)$. For this C, the FW method cannot achieve a better complexity than $O(1/K)$. Detailed proof of the statement can be found in the work by Lan(2014)[3].

Positive Example:

Definition 3. ***(Strongly Convex Set)*** *When a set C is a μ-strongly convex set w.r.t a norm $||\cdot||$, i.e. $x, z \in C$ implies that a ball centered at $\alpha x + (1-\alpha)z$ with a radius in $\alpha(1-\alpha)\frac{\mu}{2}||x-z||^2$ is in C, where $\alpha \in [0,1]$.*

A function $f(\cdot)$ exists, which is L-smooth and μ-strongly convex such that FW converges in K iterations to a ϵ optimality gap where:

$$K = O\left(\sqrt{\frac{L}{\mu}} \log \frac{1}{\epsilon}\right)$$

The proof can be found in the work by Wang et al.(2023) [4].
These examples show that the behaviour of the FW method depends on the choice of the constraint set C.

References

[1] John Duchi. Introductory Lectures on Stochastic Optimization https://urldefense.com/v3/__https://stanford.edu/*jduchi/PCMIConvex/Duchi16.pdf__;fg!!Mih3wA!EQww7byd9EiPF5s_NWwXUBVH4S99BNt5YUyCrqiYx9AmTI8sMKBAPhD0D04_6Hp1uJgzN0pxyQM2QbatKg$

[2] Elad Hazan. Introduction to Online Convex Optimization https://urldefense.com/v3/__https://arxiv.org/abs/1909.05207__;!!Mih3wA!EQww7byd9EiPF5s_NWwXUBVH4S99BNt5YUyCrqiYx9AmTI8sMKBAPhD0D04_6Hp1uJgzN0pxyQMQJMB2lw$

[3] Guanhui Lan. The Complexity of Large-scale Convex Programming under a Linear Optimization Oracle https://urldefense.com/v3/__https://arxiv.org/abs/1309.5550__;!!Mih3wA!EQww7byd9EiPF5s_NWwXUBVH4S99BNt5YUyCrqiYx9AmTI8sMKBAPhD0D04_6Hp1uJgzN0pxyQMI-jhcqg$

[4] Jun-Kun Wang, Jacob Abernethy, Kfir Y. Levy No-Regret Dynamics in the Fenchel Game: A Unified Framework for Algorithmic Convex Optimization https://urldefense.com/v3/__https://arxiv.org/abs/2111.11309__;!!Mih3wA!EQww7byd9EiPF5s_NWwXUBVH4S99BNt5YUyCrqiYx9AmTI8sMKBAPhD0D04_6Hp1uJgzN0pxyQMUb1mHmg$