1 Outline

In this lecture, we study

- matrix completion via convex relaxation.

2 Low-Rank Approximation via Convex Relaxation

Note that

$$f(X) = \frac{1}{2}\|D - X\|_F^2 = \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{p}(D_{ij} - X_{ij})^2$$

is convex in X. However, the constraint $\text{rank}(X) \le k$ induces a nonconvex feasible set. Then we may take the following relaxation using the **nuclear norm**.

$$\min_{X\in\mathbb{R}^{n\times p}} \quad \frac{1}{2}\|D - X\|_F^2 \quad \text{subject to} \quad \|X\|_* \le k \tag{8.1}$$

where

$$\|X\|_* = \text{Trace}(\sqrt{X^\top X}) = \sum_{i=1}^{\min\{n,p\}} \sigma_i(X) \le k$$

and $\sigma_1(X), \dots, \sigma_{\min\{n,p\}}(X)$ are the singular values of X. It is known that the nuclear norm is a convex function in X.

In problem (8.1), the constraint set

$$B = \left\{X \in \mathbb{R}^{n\times p} : \ \|X\|_* \le k\right\}$$

is what is obtained from scaling up the unit nuclear norm ball $\{X \in \mathbb{R}^{n\times p} : \ \|X\|_* \le 1\}$. To solve (8.1), we may apply **projected gradient descent** over the constraint set B. It is known that projection onto the constraint set B amounts to computing the singular value decomposition of matrix D [DSSSC08].

3 General Formulation

Given a matrix $D \in \mathbb{R}^{n\times p}$, let us consider

$$\min_{X\in\mathbb{R}^{n\times p}} \quad \|D - O \odot X\|_F \quad \text{subject to} \quad \text{rank}(X) \le k$$

where

- O is the binary matrix with

$$O_{ij} = \begin{cases} 1, & \text{if } D_{ij} \neq 0, \\ 0, & \text{if } D_{ij} = 0, \end{cases}$$

- $O \odot X$ is the **Hadamard product** of O and X given by

$$(O \odot X)_{ij} = O_{ij} X_{ij}.$$

Basically, $O \odot X$ is what is obtained from X after keeping only the entries that correspond to the observable entries of D. For this version of the matrix completion problem, we need a different method. As before, we take a relaxation using the nuclear norm as follows.

$$\min_{X \in \mathbb{R}^{n \times p}} \quad \frac{1}{2}\|D - O \odot X\|_F^2 \quad \text{subject to} \quad \|X\|_* \leq k.$$

Moreover, note that

$$f(X) = \frac{1}{2}\|D - O \odot X\|_F^2 = \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{p}(D_{ij} - O_{ij}X_{ij})^2$$

is convex in X as D and O are fixed matrices. Note that

$$\nabla f(X) = O \odot X - D.$$

In general, we may consider any convex function $f(X)$ on X and

$$\min_{X \in \mathbb{R}^{n \times p}} \quad f(X) \quad \text{subject to} \quad \|X\|_* \leq k \tag{8.2}$$

with the gradient $\nabla f(X)$ of the function $f(X)$. To solve (8.2), we apply another interative algorithm, the **Frank-Wolfe method**.

3.1 Frank-Wolfe Method

In this section, we introduce the **conditional gradient method**, introduced by Frank and Wolfe in 1956 [FW56]. Named after the author, the conditional gradient method is often referred to as the **Frank-Wolfe algorithm**. We consider the following convex optimization problem

$$\min_{x \in \mathbb{R}^d} \quad f(x) \quad \text{subject to} \quad x \in C$$

where f is β-smooth in a norm $\|\cdot\|$ for some $\beta > 0$ and C is a convex set. A pseudo-code of the method is given in Algorithm 1.

Algorithm 1 Frank-Wolfe Algorithm

Initialize $x_1 \in C$.
for $t = 1, \ldots, T-1$ **do**
  Take $v_t \in \operatorname{argmin}_{v \in C} \nabla f(x_t)^\top v$.
  Update $x_{t+1} = (1 - \lambda_t)x_t + \lambda_t v_t$ for some $0 < \lambda_t < 1$.
end for
Return x_T.

The main component of the conditional gradient method is to compute the direction v_t by solving

$$\min_{v \in C} \nabla f(x_t)^\top v$$

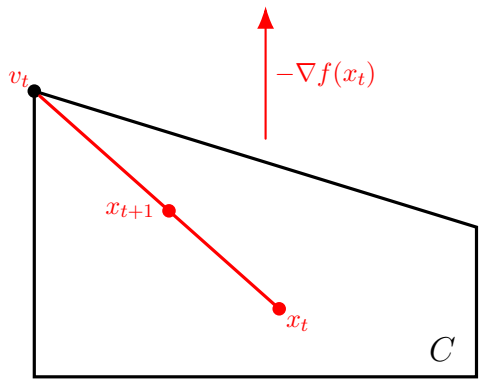

Figure 8.1: Illustration of an update from conditional gradient descent

whose objective is a linear function. In particular, when C is a polyhedron, it is just a linear program. Figure 8.1 provides a pictorial description of the update rule of the Frank-Wolfe algorithm. v_t is a point up to which we can move as far as we can in the direction of $-\nabla f(x_t)$ within C. Then we take a convex combination of the current point x_t and v_t to obtain the new iterate v_{t+1}.

The next theorem shows that conditional gradient descent converges with rate $O(1/T)$ for any smooth function with repsect to an arbitrary norm.

Theorem 8.1. *Let $f : \mathbb{R}^d \to \mathbb{R}$ be a convex function that is β-smooth in a norm $\|\cdot\|$ for some $\beta > 0$. Let $\{x_t : t = 1, \ldots, T\}$ be the sequence of iterates generated by the Frank-Wolfe algorithm with*

$$\lambda_t = \frac{2}{t+1}$$

for each t. Then for any $t \geq 2$,

$$f(x_t) - f(x^*) \leq \frac{2\beta R^2}{t+1}$$

where x^* is an optimal solution to $\min_{x \in C} f(x)$ and $R = \sup_{x,y \in C} \|x - y\|$.

Proof. Note that

$$\begin{aligned}
f(x_{t+1}) - f(x_t) &\leq \nabla f(x_t)^\top (x_{t+1} - x_t) + \frac{\beta}{2}\|x_{t+1} - x_t\|^2 \\
&= \lambda_t \nabla f(x_t)^\top (v_t - x_t) + \frac{\beta}{2}\|x_{t+1} - x_t\|^2 \\
&\leq \lambda_t \nabla f(x_t)^\top (x^* - x_t) + \frac{\beta}{2}\|x_{t+1} - x_t\|^2 \\
&\leq \lambda_t (f(x^*) - f(x_t)) + \frac{\beta}{2}\|x_{t+1} - x_t\|^2
\end{aligned}$$

where the first inequality is from the β-smoothness of f, the first equality follows from $x_{t+1} = (1 - \lambda_t)x_t + \lambda_t v_t$, the second inequality is due to the definition of $v_t = \operatorname{argmin}_{v \in C} \nabla f(x_t)^\top v$, and the last inequality is by the convexity of f. Since

$$\|x_{t+1} - x_t\| = \lambda_t \|v_t - x_t\| \leq \lambda_t R,$$

it follows that

$$f(x_{t+1}) - f(x^*) \leq (1-\lambda_t)(f(x_t) - f(x^*)) + \frac{\beta\lambda_t^2 R^2}{2}$$
$$= \frac{t-1}{t+1}(f(x_t) - f(x^*)) + \frac{2\beta R^2}{(t+1)^2}.$$

By this inequality, it follows that

$$f(x_2) - f(x^*) \leq \frac{\beta R^2}{2} \leq \frac{2\beta R^2}{3}.$$

Then by the induction hypothesis,

$$f(x_{t+1}) - f(x^*) \leq \frac{2(t-1)+2}{(t+1)^2}\beta R^2 = \frac{t}{(t+1)^2}2\beta R^2 \leq \frac{1}{t+2}\beta R^2,$$

as required. □

3.2 Applying the Frank-Wolfe method

The Frank-Wolfe algorithm applied to (8.2) has the following step of computing the direction V_t. Given $X_t \in \mathbb{R}^{n\times p}$,

$$V_t \in \operatorname{argmin}_{V\in B} \nabla f(X_t)^\top V$$

where

$$B = \left\{X \in \mathbb{R}^{n\times p} : \ \|X\|_* \leq k\right\}.$$

We will show that V_t can be computed by the power method! To argue this, we need the following lemmas.

Lemma 8.2. *The unit nuclear norm ball is equivalent to the convex hull of rank 1 matrices, i.e.,*

$$\left\{X \in \mathbb{R}^{n\times p} : \ \|X\|_* \leq 1\right\} = \operatorname{conv}\left\{uv^\top : \ \|u\|_2 = \|v\|_2 = 1, \ u \in \mathbb{R}^n, \ v \in \mathbb{R}^p\right\}.$$

Based on Lemma 8.2, we prove the second lemma.

Lemma 8.3. *Let $A \in \mathbb{R}^{n\times p}$ be an $n \times p$ matrix. Let u and v be the top left and right singular vectors of $-A$, respectively. Then*

$$k \cdot uv^\top \in \operatorname{argmin}\left\{A^\top X : \ \|X\|_* \leq k\right\}.$$

By Lemma 8.3, it follows that we can set V_t as

$$V_t = k \cdot u_t v_t^\top$$

where u_t and v_t are the top left and right singular vectors of

$$-\nabla f(X_t) = D - O \odot X_t,$$

respectively. Here, u_t and v_t can be computed by the power method. To summarize, we get the following pseudo-code.

Algorithm 2 Matrix Completion by the Frank-Wolfe Algorithm

Initialize $X_1 \in B$.
for $t = 1, \ldots, T-1$ **do**
  Compute the top left and right singular vectors u_t and v_t of $D - O \odot X_t$ by the power method
  Update $X_{t+1} = (1 - \lambda_t)X_t + \lambda_t V_t$ for some $0 < \lambda_t < 1$.
end for
Return X_T.

References

[DSSSC08] John Duchi, Shai Shalev-Shwartz, Yoram Singer, and Tushar Chandra. Efficient projections onto the l1-ball for learning in high dimensions. In *Proceedings of the 25th International Conference on Machine Learning*, ICML '08, page 272–279, New York, NY, USA, 2008. Association for Computing Machinery. 2

[FW56] Marguerite Frank and Philip Wolfe. An algorithm for quadratic programming. *Naval Research Logistics Quarterly*, 3(1-2):95–110, 1956. 3.1