Columbia University in the City of New York
Optimization Methods for Machine Learning
Instructors: Satyen Kale
Authors: Chao Qin
Email: cq2199@columbia.edu

SCRIBE

17

Analysis of the Frank-Wolfe Algorithm

In this lecture, we analyze the Frank-Wolfe algorithm, which is also called conditional gradient method.

Algorithm 1: Frank-Wolfe Algorithm/Conditional Gradient Method

1. Start with arbitrary $\mathbf{x}_0 \in K$
2. For $t = 0, 1, \ldots, T-1 \ldots$
   (a) Compute $\mathbf{y}_{t+1} = \text{argmax}_{\mathbf{x} \in K} - \nabla f(\mathbf{x}_t)^\top \mathbf{x}$
   (b) $\mathbf{x}_{t+1} = \lambda_t \mathbf{y}_{t+1} + (1 - \lambda_t)\, \mathbf{x}_t$
3. Output $\mathbf{x}_T$

One great property of this algorithm is that $\{x_t\}$ is always inside K. Now we analyze this algorithm.

Theorem 17.1 Let f be a convex and β-smooth function, and $\text{diam}(K) \leq D$. If $\lambda_t = \frac{2}{t+2}, \forall t$, then

$$f(\mathbf{x}_t) - f(\mathbf{x}^*) \leq \frac{2\beta D^2}{t+1}, \quad \forall t \geq 1.$$

Proof. First we have

$$\begin{aligned} f(\mathbf{x}_{t+1}) &\leq f(\mathbf{x}_t) + \nabla f(\mathbf{x}_t)^\top (\mathbf{x}_{t+1} - \mathbf{x}_t) + \frac{\beta}{2} \| \mathbf{x}_{t+1} - \mathbf{x}_t \|^2 \\ &= f(\mathbf{x}_t) + \lambda_t \nabla f(\mathbf{x}_t)^\top (\mathbf{y}_{t+1} - \mathbf{x}_t) + \frac{\beta}{2} \lambda_t^2 \| \mathbf{y}_{t+1} - \mathbf{x}_t \|^2 \\ &\leq f(\mathbf{x}_t) + \lambda_t \nabla f(\mathbf{x}_t)^\top (\mathbf{x}^* - \mathbf{x}_t) + \frac{\beta}{2} \lambda_t^2 D^2 \\ &\leq f(\mathbf{x}_t) + \lambda_t (f(\mathbf{x}^*) - f(\mathbf{x}_t)) + \frac{\beta}{2} \lambda_t^2 D^2 \end{aligned}$$

where the second to last inequality uses the optimality of $\mathbf{y}_{t+1}$ and the last inequality just follows from the convexity of f. This leads to

$$f(\mathbf{x}_{t+1}) - f(\mathbf{x}^*) \leq (1 - \lambda_t)(f(\mathbf{x}_t) - f(\mathbf{x}^*)) + \frac{\beta}{2} \lambda_t^2 D^2. \quad (1)$$

Now we are ready to prove the main claim by induction. When $t = 1$, we have $\lambda_0 = \frac{2}{0+2} = 1$, and then

$$f(\mathbf{x}_1) - f(\mathbf{x}^*) \le (1 - \lambda_0)(f(\mathbf{x}_0) - f(\mathbf{x}^*)) + \frac{\beta}{2}\lambda_0^2 D^2 = \frac{\beta}{2} D^2 \le \beta D^2.$$

Suppose the claim holds for t. Then for $t + 1$, we have

$$\begin{aligned} f(\mathbf{x}_{t+1}) - f(\mathbf{x}^*) &\le (1 - \lambda_t)(f(\mathbf{x}_t) - f(\mathbf{x}^*)) + \frac{\beta}{2}\lambda_t^2 D^2 \\ &\le \frac{t}{t+2}\frac{2\beta D^2}{t+1} + \frac{\beta}{2}\left(\frac{2}{t+2}\right)^2 D^2 \\ &= \frac{2\beta D^2}{t+2}\left(\frac{t}{t+1} + \frac{1}{t+2}\right) \\ &\le \frac{2\beta D^2}{t+2}. \end{aligned}$$

This completes the proof. □

This result shows that the Frank-Wolfe algorithm achieves the same convergence rate as projected gradient descent. The output of the Frank-Wolfe algorithm has some kind of sparsity structure. For example, let K be the simplex of distributions, i.e,

$$K = \left\{ \mathbf{x} : \sum_{i=1}^{d} x_i = 1, \quad x_i \ge 0, \forall i \in [d] \right\}.$$

Given $\mathbf{v}$,

$$\text{argmax}_{\mathbf{x} \in K} \mathbf{v}^\top \mathbf{x} = \mathbf{e}_i$$

where $\mathbf{e}_i$ is the standard basis vector where $i = \text{argmax}_{j \in [d]} v_j$. Then the Frank-Wolfe algorithm over K performs similarly as the coordinate descent algorithm.