COMS 4995-004: Optimization for Machine Learning
Homework 4 **(Corrected version)**

HW4 is due Wednesday, Nov 27 by 1:00 pm. No late assignments will be accepted[1]. Please refer to `https://www.satyenkale.com/optml-f19/` for instructions on how to submit homework assignments.

As discussed in class, projected gradient descent makes use of a projection oracle, whereas the Frank-Wolfe method uses a linear optimization oracle. A linear optimization oracle is easier to implement than a projection oracle, and in this exercise we will formalize this qualitative statement. Suppose that K is a convex set in $\mathbb{R}^d$ in the unit ℓ_2 ball, i.e. for all $x \in K$, we have $\|x\|_2 \leq 1$. A *linear optimization oracle* for K is an algorithm that, given any vector $v \in \mathbb{R}^d$, computes $\arg\max_{x \in K} v \cdot x$. A *projection oracle* for K is an algorithm that, given any point $y \in \mathbb{R}^d$, computes $\Pi_K(y) := \arg\min_{x \in K} \|y - x\|_2^2$.

1. **(9 points)** Suppose we are given a projection oracle for K. We now want to implement an ϵ-*approximate* linear optimization oracle for K using the projection oracle, i.e. given a vector $v \in \mathbb{R}^d$, we want to find a point $x \in K$ such that $v \cdot x \geq \max_{x' \in K} v \cdot x' - \epsilon$. Show that we can find such a point x by making *one* call to the projection oracle applied to a carefully chosen point y (i.e. by computing $\Pi_K(y)$ for some point y). Give a precise formula for y in terms of v and ϵ.
   Hint: consider applying the projection oracle to a point $y = \alpha v$ for some scalar α.

2. **(16 points)** Suppose we are given a linear optimization oracle for K. We now want to implement an ϵ-*approximate* projection oracle for K using the linear optimization oracle, i.e. given a point $y \in \mathbb{R}^d$, we want to find a point $x \in K$ such that $\|y - x\|_2^2 \leq \min_{x' \in K} \|y - x'\|^2 + \epsilon$. Show that we can find such a point x by making $O(\frac{1}{\epsilon})$ calls to the linear optimization oracle. Describe your implementation of the projection oracle via pseudo-code.
   Hint: consider computing the projection via the Frank-Wolfe method.

Solution: Part 1. Suppose we apply the projection oracle to the point $y = \alpha v$ for some scalar α. Let $x = \Pi_K(y)$, and let $x' = \arg\max_{x'' \in K} v \cdot x''$. Then, since x is the projection of y on K and $x' \in K$, we have $\|x - y\|_2^2 \leq \|x' - y\|_2^2$. This implies that

$$0 \leq \|x' - y\|_2^2 - \|x - y\|_2^2 = \|x'\|^2 - \|x\|^2 - 2\alpha v \cdot (x' - x) \leq 1 - 2\alpha v \cdot (x' - x).$$

The last inequality follows because $\|x'\| \leq 1$. Thus, we have

$$v \cdot x \geq v \cdot x' - \frac{1}{2\alpha}.$$

[1] Unless you have an emergency; in that case please write to Satyen as soon as possible.

So if we choose $\alpha = \frac{1}{2\epsilon}$, then we have $v \cdot x \geq \max_{x' \in K} v \cdot x' - \epsilon$ as required.

Solution: Part 2. Finding a point $x \in K$ such that $\|y - x\|_2^2 \leq \min_{x' \in K} \|y - x'\|^2 + \epsilon$ is equivalent to solving the optimization problem $\min_{x' \in K} \|y - x'\|^2$ up to an optimization error of ϵ. In order to solve this optimization problem using a linear optimization oracle, we employ the Frank-Wolfe method. The pseudo-code for the implementation is given below:

1. Start with an arbitrary point $x_0 \in K$.
2. For $t = 0, 1, 2, \ldots, T-1$:
   (a) Compute $\nabla[\|y - x\|^2]_{x = x_t} = 2(x_t - y)$.
   (b) Compute $v_t = \arg\max_{v \in K} -2(x_t - y) \cdot v$ via a call to the linear optimization oracle.
   (c) Update $x_{t+1} = \frac{2}{t+2} v_t + \left(1 - \frac{2}{t+2}\right) x_t$.
3. Output x_T.

In order to compute a bound on T to obtain an ϵ-suboptimal solution, we need to compute the smoothness constant of the objective function $\|y - x\|^2$. The Hessian of this function is $2I$, so the function is $\beta = 2$ smooth. The diameter of K is at most $D = 2$ since for all $x \in K$ we have $\|x\| \leq 1$. Via the analysis of the Frank-Wolfe method, the optimization error after T iterations is bounded by $\frac{2\beta D^2}{T+1} = \frac{16}{T+1}$. Thus, in order to get optimization error at most ϵ, taking $T = \frac{16}{\epsilon} - 1 = O(\frac{1}{\epsilon})$ suffices.