535526 Fall 2022: Optimization Algorithms **(Due: 2023/1/5, 21:00)**

Homework 2: Frank-Wolfe, Accelerated GD, and Mirror Descent

Submission Guidelines: Your deliverables shall consist of 2 separate files – (i) A PDF file: Please compile all your write-ups and your report into one .pdf file (photos/scanned copies are acceptable; please make sure that the electronic files are of good quality and reader-friendly); (ii) A zip file: Please compress all your source code into one .zip file. Please submit your deliverables via E3.

Problem 1 (Nesterov's Accelerated Gradient) (8+8+8+8+8+8=48 points)

Recall from Lecture 9 that we have reformulated Nesterov's accelerated gradient as an ordinary differential equation and established the convergence. In this problem, let us formally prove the convergence rate of the original form of the Nesterov's method step by step. Recall that in each iteration, Nesterov's method with a step size $\eta_t = 1/L$ proceeds as follows: Given any initial points x_1, y_1 that satisfy $x_1 = y_1$,

$$x_{t+1} = y_t - \frac{1}{L}\nabla f(y_t) \tag{1}$$

$$y_{t+1} = x_{t+1} - \frac{1-\theta_t}{\theta_{t+1}}(x_{t+1} - x_t) \tag{2}$$

$$\theta_{t+1} = \frac{1+\sqrt{1+4\theta_t^2}}{2}, \quad \theta_0 = 0. \tag{3}$$

Note that (3) implies that $\theta_{t+1}^2 - \theta_{t+1} - \theta_t^2 = 0$.

(a) Show that for any $x, y \in \mathbb{R}^d$, we have

$$f\big(y - \frac{1}{L}\nabla f(y)\big) - f(x) \le -\frac{1}{2L}\|\nabla f(y)\|^2 - \nabla f(y)^\top (x-y). \tag{4}$$

(Hint: Use convexity and smoothness)

(b) By leveraging the result in (a), show that

$$f(x_{t+1}) - f(x_t) \le -\frac{L}{2}\|x_{t+1} - y_t\|^2 + L(x_{t+1} - y_t)^\top (x_t - y_t). \tag{5}$$

(c) Similarly, by leveraging the result in (a), show that

$$f(x_{t+1}) - f(x^*) \le -\frac{L}{2}\|x_{t+1} - y_t\|^2 + L(x_{t+1} - y_t)^\top (x^* - y_t). \tag{6}$$

(d) By adding $\theta_t(\theta_t - 1)$ times of (5) and θ_t times of (6), show that

$$\theta_t^2\Delta_{t+1} - \theta_{t-1}^2\Delta_t \le -\frac{L}{2}\big(\|\theta_t(x_{t+1} - y_t)\|^2 + 2\theta_t(x_{t+1} - y_t)^\top \underbrace{(\theta_t y_t - (\theta_t - 1)x_t - x^*)}_{=:\phi_t}\big) \tag{7}$$

(e) By completing the square for the RHS of (7) and using the update rule of Nesterov's method, show that

$$\theta_t^2\Delta_{t+1} - \theta_{t-1}^2\Delta_t \le \frac{L}{2}\Big(\|\phi_t\|^2 - \|\phi_{t+1}^2\|\Big) \tag{8}$$

(f) Finally, by using an induction argument, show that the convergence rate of Nesterov's method is

$$f(x_t) - f(x^*) \leq \frac{2L\|x_1 - x^*\|^2}{t^2}. \tag{9}$$

Problem 2 (Bregman Divergence) (12 points)

Recall from Lecture 11 that we learned the Bregman divergence, which is a key component in Mirror Descent. As mentioned in Page 26 of Lecture 11, show that Bregman divergence satisfies the Generalized Pythagorean Theorem, i.e.,

$$D_\phi(z\|x) \geq D_\phi(z\|\bar{x}) + D_\phi(\bar{x}\|x), \tag{10}$$

where $\bar{x}$ is the Bregman projection of x onto the feasible set C.

Problem 3 (Gurobi Optimization Solver for Frank-Wolfe) (40 points)

In this problem, you will have the opportunity to use a very useful tool – Gurobi Optimization Solver. Specifically, let us implement Frank-Wolfe method with the help of Gurobi optimization solver, which can be used through its Python API, and reproduce the performance of Frank-Wolfe in the top-left subfigure of Figure 2 of the paper (`https://arxiv.org/abs/2002.07003`) on a Portfolio Management dataset (provided to you on E3). Portfolio Management can be formulated as a constrained problem can be described as

$$\min_x \quad f(x) := -\sum_{i=1}^{n} \log(a_i^\top x) \tag{11}$$

$$\text{subject to} \quad \sum_{j=1}^{p} x_j = 1, x \geq 0, \tag{12}$$

where each a_i is a p-dimensional vector. To facilitate gradient computation, you may write your code in either PyTorch or TensorFlow. If you are a beginner in learning the deep learning framework, please refer to the following tutorials:

- PyTorch: `https://pytorch.org/tutorials/`
- Tensorflow: `https://www.tensorflow.org/tutorials`

For the introduction to the Python API for Gurobi Optimization Solver, please see:

- To use Gurobi, you need to install Gurobipy (`https://pypi.org/project/gurobipy/`)
- Examples can be found at `https://www.gurobi.com/documentation/`

For the deliverables, please submit the following:

- Technical report: Please summarize all your experimental results in 1 single report (and please be brief)
- All your source code