CME 307/MS&E 311: Optimization
Instructor: Prof. Madeleine Udell
Final Exam: Spring 2023

HONOR CODE

In taking this examination, I acknowledge and accept Stanford University Honor Code.

NAME (Signed) : ______________________________

NAME (Printed): ______________________________

Problem	Full Points	Score
1	15	
2	15	
3	5 (Bonus)	
Total	30 + 5	

CME 307/MS&E 311 **Final**
Optimization **Spring 2022-2023**
Prof. Udell **3:30-5:00 pm, June 12th**

Note: You have 90 minutes to work on the exam, and you are allowed a single-sided one-page US-letter-size cheat sheet. In taking this exam, you acknowledge and accept the Stanford University Honor Code.

Question 1. *Quadratic models and smooth convex optimization.* Consider the optimization problem:

$$\text{minimize} \quad f(x) \tag{1}$$

where $f : \mathbf{R}^n \to \mathbf{R}$ is convex and L-smooth.

(a) Consider an algorithm that iteratively solves (1) by minimizing the quadratic model

$$x_{k+1} = \underset{x}{\operatorname{argmin}} \left\{ f(x_k) + \langle \nabla f(x_k), x - x_k \rangle + \frac{1}{2\eta} \|x - x_k\|_{A_k}^2 \right\},$$

given a sequence of symmetric positive-definite matrices $A_k \in \mathbf{S}_{++}^n$ and a scalar $\eta > 0$. Solve the preceding optimization problem to explicitly write x_{k+1} as a function of x_k.

(b) In the special case that $A_k = I$ for every iteration k, what is the name of this algorithm? Further, state a value of η that ensures descent, i.e.,

$$f(x_{k+1}) \leq f(x_k).$$

(c) The theoretical value for η that ensures descent can result in very slow convergence in practice. How would you recommend selecting the stepsize in practice for the algorithm in (b)? You may simply state the name of a method; no need to write any math. Justify your choice.

(d) You are consulting for a company designing a classification model to detect credit-card fraud, which requires solving a large logistic regression problem:

$$\text{minimize} \quad \tfrac{1}{m} \textstyle\sum_{i=1}^{m} \ell_{\text{logistic}}\left((Ax)_i;\, y_i\right),$$

where $A \in \mathbf{R}^{m \times n}$ is a *very sparse* data matrix with $m = 10^9$ samples and $n = 10^6$ features, and $y_i \in \{+1, -1\}$ the label of the ith datapoint. The client is currently running the algorithm in (b), but it takes too long to solve. Unfortunately, the number of features is so large that any algorithm that uses memory or compute quadratic in n is unacceptable. What algorithm would you recommend as replacement to achieve faster convergence? Note, there is more than one reasonable answer to this question, but you only need to provide one recommendation. Justify your answer.

Solution. *Quadratic models and smooth convex optimization.*

(a) To find x_{k+1} we apply the first-order optimality condition, and set the gradient of the defining objective equal to 0, from which we find

$$\nabla f(x_k) + \frac{1}{\eta} A_k(x_{k+1} - x_k) = 0.$$

Solving for x_{k+1}, we obtain

$$x_{k+1} = x_k - \eta A_k^{-1} \nabla f(x_k).$$

(b) When $A_k = I$ for all k, we recover the gradient descent algorithm. To ensure descent, one should select $\eta = \frac{1}{L}$. As with this choice, the L-smoothness bound guarantees

$$f(x_{k+1}) \leq f(x_k) - \frac{1}{L}\|\nabla f(x_k)\|^2.$$

(c) You should use Armijo Line Search. This procedure will often lead to a step that makes more progress then the theoretical bound in (b). For details on the procedure, see the course slides.

(d) There are 3 reasonable answers here. We are told A is extremely sparse, so it is feasible to compute full gradients (as the cost of computing ∇f is proportional to applying A to vectors for logistic regression). As full gradients are feasible to compute, L-BFGS and Accelerated gradient descent are both good recommendations. or the third possible recommendation, observe the objective has finite-sum structure, so we can easily form a stochastic gradient for the objective. Thus, stochastic gradient methods like SGD and SVRG are also appropriate choices. However, note SVRG is only a reasonable option as A is very sparse, which allows for computation of full gradients.

Question 2 [Duality and Resource Allocation] (15 points)

A multinational firm has an operating budget $M > 0$ that can be allocated across its divisions in each of n regions. Let x_j be the allocation to region j, $j = 1, \dots, n$. Each division j has estimated the 5-year profit they expect to achieve given allocation x; we will represent the negative profit as the function $f_j(x)$, $j = 1, \dots, n$. As the CFO of this firm, you plan to allocate the budget by solving the problem

$$\begin{array}{rl} \text{minimize} & \sum_{j=1}^{n} f_j(x_j) \\ \text{subject to} & \sum_{j=1}^{n} x_j = M, \\ \text{variables} & x_j \geq 0 \quad j = 1, ..., n. \end{array} \tag{2}$$

Assume the problem has a unique solution $x^\star \in \mathbf{R}^n$, and Slater's condition holds.

(a) (5 points) Write down the KKT conditions for (2). Prove that there exists a threshold $\lambda^\star \in \mathbf{R}$ such that

$$\begin{aligned} f_j'(x_j^\star) &= \lambda^\star \quad \text{if } x_j^\star > 0, \\ f_j'(x_j^\star) &\geq \lambda^\star \quad \text{if } x_j^\star = 0, \end{aligned}$$

for all $j = 1, \dots, n$ and any optimal solution $(x_1^\star, \dots, x_n^\star)$ of (2). (Hint: You can assume that all optimal solutions of (2) satisfy the KKT conditions).

In the following, suppose $f_j(x_j) = -w_j \log(x_j)$ where $w_j > 0$ for $j = 1, \dots, n$.

(b) (4 points) Write down the dual problem of (2). Find the dual optimal solution. (Hint: to find the dual optimal solution, you can first show that the dual optimal solution satisfies $\lambda^\star = -\frac{1}{M} \sum_{j=1}^{n} w_i$ using the KKT conditions.)

(c) (4 points) Write down an analytical formula for the optimal solution of (2).

(d) (2 points) Verify that for any $(x_1, \dots, x_n)$ in the feasible set of (2), the corresponding primal objective value $\sum_{j=1}^{n} f(x_j)$ is no less than the optimal dual objective value.

Solution.[Duality and Resource Allocation]

(a) The Lagrangian of the problem is given by

$$L(x, \lambda, \mu) = \sum f_j(x_j) + \lambda(M - \sum x_j) - \sum \mu_j x_j$$

and the first-order conditions will be

$$\frac{\partial L}{\partial x_j} = f_j'(x_j) - \lambda - \mu_j = 0$$

$$\lambda(M - \sum x_j) = 0$$

$$\mu_j x_j = 0$$

$$\mu_j \geq 0$$

From the first and last equation, we get

$$f_j'(x_j) = \lambda + \mu_j \geq \lambda.$$

λ^* represents the rate of optimal objective change over the change of right-hand-side resource M. Note that if $x_j > 0$ then $\mu_j = 0$ which implies

$$f_j'(x_j) = \lambda$$

In this case, λ^* corresponds to the marginal gain from increasing each one of the f_j when $x_j^* > 0$, that is, the optimal allocation of x_i is at the point such that every function f_j has the same marginal gain λ^*.

(b) To have the dual problem, we first compute the dual objective function by minimizing the Lagrangian function with respect to x. By the stationarity condition, we have $x_j = -\frac{w_j}{\lambda+\mu_j}$ for all j and the dual objective function is

$$\sum_{j=1}^{n} -w_j \log\left(-\frac{w_j}{\lambda + \mu_j}\right).$$

Thus, the dual problem is

$$\begin{aligned} \text{maximize} \quad & \sum_{j=1}^{n} -w_j \log\left(-\frac{w_j}{\lambda + \mu_j}\right) \\ \text{subject to} \quad & \mu_j \geq 0, \quad j = 1, 2, ... \end{aligned}$$

To find the optimal solution, we multiply the first FOC by x_j and get

$$f_j'(x_j)x_j - \lambda x_j - \mu_j x_j = 0$$

Replacing $\mu_j x_j = 0$ and summing over i

$$\sum f_j'(x_j)x_j = \lambda \sum x_j$$

And using the complementary condition $\lambda(M - \sum x_j) = 0$ we get

$$-\sum w_j = \sum f_j'(x_j)x_j = \lambda M$$

which gives the result.

(c) By complementary condition and the condition that the optimal primal solutions are strictly larger than 0, we have $\mu_j = 0$ for all j. Then, by the stationarity condition, we have $x_j = -\frac{w_j}{\lambda}$. Plugging in the value of λ from Part (b), we have the result.

(d) Weak convexity. We can also verify it by calculation.

Question 3 [Convergence Rate of SGD proved by GPT-4] (5 points, Bonus)

Recently, large language models like GPT-4 have garnered significant attention due to their impressive capabilities, including their ability to solve intricate scientific problems. In this examination problem, we'll examine GPT-4's performance in proving the convergence rate of the stochastic gradient descent (SGD) method.

Consider a μ-strongly convex and L-smooth function $f(x)$ defined on $\mathbb{R}^n$, where n is a positive integer, and $\mu, L > 0$ are two positive constants. Denote the solution to the unconstrained minimization problem as $x^\star = \arg\min f(x)$ as the minimizer and the initial iterate as x_0. Suppose $f(x)$ is twice differentiable. SGD updates the iterate as

$$x_{t+1} = x_t - \eta g_t,$$

where $\eta > 0$ is the step size, g_t is a stochastic gradient at x_t satisfying $\mathbb{E}[g_t] = \nabla f(x_t)$ for all $t = 1, 2, \ldots$

Suppose that $\|g_t\| \leq G$ for a positive constant $G > 0$, for all $t = 1, 2, \ldots$. In homework 1, we showed that, with some specified step size η, SGD finds an ϵ-optimal solution x_t such that

$$\mathbb{E}\left[f(x_t) - f(x^\star)\right] \leq \epsilon$$

within $O\left(\frac{1}{\epsilon}\log\left(\frac{2(f(x_0)-f(x^\star))}{\epsilon}\right)\right)$ iterations for any x_0.

Can GPT-4 match the mathematical prowess of students in CME 307? To show this statement, GPT-4 proposes the following proof. Is this proof correct? If not, identify the first wrong step, and correct the proof to show the statement.

Proof of the convergence rate of SGD provided by GPT-4.
Given a twice-differentiable function $f(x)$ defined on $\mathbb{R}^n$ that is μ-strongly convex and L-smooth. The minimizer of the function is given by $x^\star = \arg\min_{x\in\mathbb{R}^n} f(x)$.

Let's apply Stochastic Gradient Descent (SGD) to find the minimizer as follows:

$$x_{t+1} = x_t - \eta g_t, \tag{3}$$

where η is the step size, and g_t is a stochastic gradient at x_t with $\mathbb{E}[g_t] = \nabla f(x_t)$ for all $t = 1, 2, ...$ We suppose that $\mathbb{E}[|g_t|_2^2] \leq G^2$ for some positive constant $G > 0$ for all t.

Our goal is to specify the step size η and prove that after at most $O(\frac{1}{\epsilon}\log(2(f(x_0) - f(x^\star))/\epsilon))$ steps,

$$\mathbb{E}[f(x_{t+1}) - f(x^\star)] \leq \epsilon. \tag{4}$$

Let's denote $f_t = f(x_t)$ and $D_t = f_t - f^\star$ where $f^\star = f(x^\star)$. Also denote G^2 as the upper bound on the expected squared norm of the stochastic gradient, i.e., $\mathbb{E}[|g_t|_2^2] \leq G^2$.

Let's denote the SGD step size as $\eta = 1/L$.

We know that the function $f(x)$ is L-smooth, hence we have the following inequality:

$$f_{t+1} \leq f_t - \eta \nabla f_t^\top g_t + \frac{L\eta^2}{2} |g_t|_2^2. \tag{5}$$

Taking the expectation on both sides, we get:

$$\mathbb{E}[f_{t+1}] \leq \mathbb{E}[f_t] - \eta \mathbb{E}[\nabla f_t^\top g_t] + \frac{L\eta^2}{2} G^2. \tag{6}$$

Because $\mathbb{E}[g_t] = \nabla f_t$, and we know the function is μ-strongly convex, we have $\mathbb{E}[\nabla f_t^\top g_t] = |\nabla f_t|_2^2 \geq \mu D_t$. Thus,

$$\mathbb{E}[D_{t+1}] \leq (1 - \eta\mu) D_t + \frac{L\eta^2}{2} G^2. \tag{7}$$

After substituting $\eta = 1/L$, we have$f(x)$ being μ-strongly convex, we find that

$$\mathbb{E}[D_{t+1}] \leq \left(1 - \frac{\mu}{L}\right) D_t + \frac{1}{2L} G^2. \tag{8}$$

Applying this inequality recursively for t steps and noticing that $D_0 = f_0 - f^\star$, we get

$$\mathbb{E}[D_t] \leq \left(1 - \frac{\mu}{L}\right)^t (f_0 - f^\star) + \frac{G^2}{2\mu} \left(1 - \left(1 - \frac{\mu}{L}\right)^t\right). \tag{9}$$

To make $\mathbb{E}[D_t] \leq \epsilon$, we need

$$\left(1 - \frac{\mu}{L}\right)^t (f_0 - f^\star) \leq \epsilon. \tag{10}$$

Solving for t, we get

$$t \geq \frac{L}{\mu} \log\left(\frac{(f_0 - f^\star)}{\epsilon}\right) = O\left(\frac{1}{\epsilon} \log\left(\frac{2(f_0 - f^\star)}{\epsilon}\right)\right), \tag{11}$$

where the O notation hides the constant factor $\frac{L}{\mu}$.

Therefore, after $O\left(\frac{1}{\epsilon} \log\left(\frac{2(f_0 - f^\star)}{\epsilon}\right)\right)$ iterations, the expected gap between the function value and its minimum is less than ϵ, i.e., $\mathbb{E}\left[f(x_{t+1}) - f(x^\star)\right] \leq \epsilon$.

This completes the proof.

$\square$