EE364b Spring 2023 Homework 2

Due Sunday 4/23 at 11:59pm via Gradescope

2.1 *(10 points) Subgradient methods for Lasso.* Consider the optimization problem

$$\text{minimize} \quad f(x) := \tfrac{1}{2}\|Ax-b\|_2^2 + \lambda\|x\|_1 \,,$$

with variables $x \in \mathbf{R}^n$ and problem data $A \in \mathbf{R}^{m\times n}$, $b \in \mathbf{R}^m$ and $\lambda > 0$. This model is known as Lasso, or Least Squares with ℓ_1 regularization, which encourages sparsity in the solution via the non-smooth penalty $\|x\|_1 := \sum_{j=1}^n |x_j|$. In this problem, we will explore various subgradient methods for fitting this model.

(a) (1 points) Derive the subdifferential $\partial f(x)$ of the objective.

(b) (1 points) Find the update rule of the subgradient method and state the computational complexity of applying one update using big O notation in terms of the dimensions.

(c) (5 points) Let $n = 1000$, $m = 200$ and $\lambda = 0.01$. Generate a random matrix $A \in \mathbf{R}^{m\times n}$ with independent Gaussian entries with mean 0 and variance $1/m$, and a fixed vector $x^* = \big[\underbrace{1,\ldots,1}_{k\,\text{times}}, \underbrace{0,\ldots,0}_{n-k\,\text{times}}\big]^T \in \mathbf{R}^n$. Let $k = 5$ and then set $b = Ax^*$. Implement the subgradient method to minimize $f(x)$, initialized at the all-zeros vector. Try different step size rules, including constant step size, constant step length, $1/\sqrt{k}$, $1/k$, Polyak's step length with estimated objective value as shown in lecture slides. Plot objective value versus iteration curves of different step size rules on the same figure.

(d) (3 points) Repeat part (c) using a heavy ball term, $\beta_k(x^k - x^{k-1})$, added to the subgradient, as described on page 25 of lecture slides. Try different step size rules as in part (c) and tune the heavy ball parameter $\beta_k = \beta$ for faster convergence.

(e) (3 points) We can reformulate the optimization problem as follows:

$$\min_{x,y} \quad \tfrac{1}{2}\|Ax-b\|_2^2 + \lambda\|y\|_1 \ \text{ s.t. } \ x = y.$$

Derive the update rule of the primal-dual subgradient method for this problem.

(f) (3 points) Run the primal-dual subgradient method to solve the optimization problem in part (e) using the same values for A and b as in part (c), and an all-zeros initialization. Try constant step size, $1/\sqrt{k}$, and $1/k$ step size rules, and plot the objective values on the same figure. How does increasing the parameter ρ affect convergence?

2.2 (4 Points) *Recovering Discrete Signals via Convex Optimization.* Suppose that x is an n dimensional signal taking values only in $\{-1, +1\}$, i.e., $x \in \{-1, +1\}^n$, and we have observations $y = Ax$. Here, $A \in \mathbb{R}^{m\times n}$ is a matrix whose entries are known. This setting is frequently encountered in wireless communication systems. Typically, the signal x carries digital information and A models the propagation of the signal over a wireless channel. You will try recovering the signal by finding a point $\hat{x}$ that satisfies $\|\hat{x}\|_\infty \leq 1$ and $A\hat{x} = y$. Generate a random matrix A with independent standard Gaussian entries and random signal $x \in \{-1, +1\}^n$ with independent uniformly distributed values in $\{-1, +1\}$ and let $y = Ax$.

(a) Formulate an optimization problem and propose an algorithm to recover a signal from measurements $y = Ax$ obeying the constraint $\|x\|_\infty \leq 1$.

(b) Plot the convergence of the algorithm in part (a) in terms of the Euclidean distance $\|\hat{x} - x\|_2$ for $n = 100$ and $m \in 50, 80, 90$. Plot the original and recovered signals.

2.3 (4 Points) *Line-search for Non-smooth Functions.* In this question, we will examine the feasibility of line-search for choosing the step-size in subgradient descent. Let $f : \mathbf{R}^n \to \mathbf{R}$ be a convex function. At iteration k of subgradient descent, the Armijo line-search selects the *largest* step-size $\alpha_k > 0$ which satisfies

$$f(x_k - \alpha_k g_k) \leq f(x_k) - c\alpha_k \|g_k\|_2^2, \tag{1}$$

where $g_k \in \partial f(x_k)$ and $c \in (0, 1)$ is a relaxation parameter. In practice, this can be achieved by reducing the step-size as $\alpha_k \leftarrow \beta\alpha_k$ for some $\beta \in (0, 1)$ until (1) is satisfied. This is called backtracking.

We will analyze the performance of this backtracking line-search procedure for the following piece-wise linear function.

$$f(x) = \begin{cases} -2x & \text{if } x \leq 0 \\ -\frac{1}{2}x & \text{if } x \in (0, 4) \\ x - 6 & \text{if } x \geq 4. \end{cases}$$

(a) Plot f over the domain $[-2, 6]$ in your favorite plotting software and report the figure. Is f a convex function? Report the minimizer(s) of f.

(b) Since f is piece-wise linear with a finite number of pieces, its subdifferential takes only a finite number of distinct set values. Report each unique subdifferential set of f and the interval over which it is valid.

(c) Suppose we attempt to minimize f using subgradient descent with the Armijo line-search. In particular, suppose that we choose a random subgradient at each iteration and backtrack on α_k until (1) holds.

Suppose $c > 0.25$ and show that there exists an initial point $x_0 \in [-2, 6]$, $x_0 \notin \operatorname{argmin}_x f(x)$ and subgradient $g_0 \in \partial f(x_0)$ such that no step-size $\alpha_0 > 0$ exists for which the Armijo condition holds.

(d) Now let $c \in (0,1)$. Modify f using knowledge of c to show that there exists a function for which the line-search fails analogously to part (c). As in part (c), enforce $x_0 \notin \operatorname{argmin}_x f(x)$.

2.4 (4 points) *Finding a point in the intersection of convex sets.* Let $A \in \mathbf{R}^{n\times n}$ be a positive definite matrix and let Σ be an $n \times n$ diagonal matrix with diagonal entries $\sigma_1, \ldots, \sigma_n > 0$, and y a given vector in $\mathbf{R}^n$. Consider the compact convex sets $\mathcal{E} = \{z \in \mathbf{R}^n \mid \|A^{1/2}(z-y)\|_2 \leq 1\}$ and $B = \{z \in \mathbf{R}^n \mid \|\Sigma z\|_\infty \leq 1\}$.

(a) (2 points) Formulate an optimization problem and propose an algorithm in order to find a point $x \in \mathcal{E} \cap B$. *You can assume that $\mathcal{E} \cap B$ is not empty.* Your algorithm must be provably converging (although you do not need to prove it and you can simply refer to the lecture slides).

(b) (2 points) Implement your algorithm with the following data: $n = 2$, $y = (3,2)$, $\sigma_1 = 0.5$, $\sigma_2 = 1$,

$$A = \begin{bmatrix} 1 & 0 \\ -1 & 1 \end{bmatrix},$$

and $x = (2,1)$. Plot the objective value of your optimization problem versus the number of iterations.

2.5 *Optional (extra credit, 4 points). Non-convex non-differentiable functions, Clarke subdifferentials and Neural Networks.* Let $f : \mathbf{R}^n \to \mathbf{R}$ be a given function that we do not assume to be convex nor to be differentiable (e.g., a deep neural network with ReLU activation functions), so that the subdifferential $\partial f(x) = \{g \in \mathbf{R}^n \mid f(y) \geq f(x) + g^\top (y-x)\ \forall y\}$ is possibly an empty set. In this question, we explore generalized subdifferentials, or Clarke subdifferentials, as we have seen on page 11 of the lecture notes.

Let $D \subset \mathbf{R}^n$ be the set of points at which f is differentiable. We assume that D has (Lebesgue) measure 1, meaning that f is differentiable *almost everywhere*. The Clarke subdifferential of f at x is then defined as

$$\partial_C f(x) = \mathbf{Co}\left\{\lim_{k\to\infty} \nabla f(x_k) \mid x_k \to x,\, x_k \in D\right\}.$$

The goal of this exercise is to characterize some basic properties of Clarke subdifferentials, relate $\partial_C f(x)$ to $\partial f(x)$ and study some implications of the condition $0 \in \partial_C f(x)$, which is necessary and sufficient for global optimality in the convex case.

We make the following technical assumption: we assume that f is locally Lipschitz, i.e., for any $x \in \mathbf{R}^n$, there exists $\eta > 0$ and $L_x > 0$ such that $|f(y) - f(z)| \leq L_x \|y - z\|_2$ for any y, z such that $\|x-y\|_2, \|x-z\|_2 \leq \eta$. Then, it follows that the function f is differentiable almost everywhere with respect to the Lebesgue measure (this result is sometimes referred to as Rademacher's theorem [BL10]).

Prove the following:

(a) If f is a continuously differentiable function then $\partial_C f(x) = \{\nabla f(x)\}$.

(b) If f is convex then $\partial_C f(x) \subseteq \partial f(x)$. Show that equality actually holds, i.e., $\partial_C f(x) = \partial f(x)$. *Hint: Suppose by contradiction that there exists $g \in \partial f(x)$ such that $g \notin \partial_C f(x)$. Set $h(x) = f(x) - g^T x$. Show that $0 \in \partial h(x)$ and $0 \notin \partial_C h(x)$. Use the hyperplane separation theorem to conclude.*

We say that x is *Clarke stationary* if $0 \in \partial_C f(x)$. If f is convex, then, from (b), we know that x is a global minimizer of f. For a non-convex function f, this property does not extend in general as we explore next.

(c) Suppose that x is a local minimum (resp. maximum) of f, i.e., there exists a radius $\eta > 0$ such that $f(y) \geq f(x)$ (resp. $f(y) \leq f(x)$) for any y such that $\|y - x\|_2 \leq \eta$. Show that x is Clarke stationary. *Hint: suppose by contradiction that $0 \notin \partial_C f(x)$ and conclude by using the hyperplane separating theorem with the convex sets $\partial_C f(x)$ and $\{0\}$.*

(d) Suppose that $\inf_x f(x) > -\infty$ and that $\inf_x f(x)$ is attained. Show that if x is the *unique* Clarke stationary point of f, then x is the unique global minimizer of f.

Finally, we study two examples of non-convex non-differentiable functions: a two-dimensional input function which has a unique Clarke stationary point that is the global minimizer, and, a neural network training loss which has a spurious Clarke stationary point at $(0, \ldots, 0)$.

(e) Consider the function with two-dimensional inputs $f(x_1, x_2) = 10\,|x_2 - x_1^2| + (1 - x_1)^2$. Show that the unique Clarke stationary point of f is $(x_1, x_2) = (1, 1)$ and that it is the unique global minimizer of f.

(f) Consider a supervised learning setting with a neural network parameterization: let $X \in \mathbf{R}^{n \times d}$ be a given data matrix and $y \in \mathbf{R}^n$ be a vector of real-valued observations. For the neural network parameters $u_1, \ldots, u_m \in \mathbf{R}^d$ and $\alpha_1, \ldots, \alpha_m \in \mathbf{R}$, consider the loss function

$$f(u_1, \ldots, u_m, \alpha_1, \ldots, \alpha_m) = \|y - \sum_{i=1}^{m} \sigma(X u_i) \alpha_i\|_2^2 \,,$$

where we have introduced the component-wise ReLU activation function σ defined as $\sigma(z) = (\max\{z_1, 0\}, \ldots, \max\{z_n, 0\}) \in \mathbf{R}^n$ for $z = (z_1, \ldots, z_n) \in \mathbf{R}^n$. Show that $0 \in \partial f_C(0, \ldots, 0, 0, \ldots, 0)$.

References

[BL10] Jonathan Borwein and Adrian S Lewis. *Convex analysis and nonlinear optimization: theory and examples*. Springer Science & Business Media, 2010.