Exercise sheet 3: Exam 2023-2024 (adapted)

Optimization for Machine Learning, M2 MIAGE ID Apprentissage

December 20, 2024

Exercise 1: A nonconvex problem

Let $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^n$ be a dataset with $y_i \in (0,1)$ for every i. Given the following loss function:

$$\ell(h, y) := \left(y - \frac{1}{1+\exp(-h)}\right)^2, \tag{1}$$

we consider the optimization problem corresponding to fitting a linear model to the data, given by

$$\operatorname*{minimize}_{\boldsymbol{w}\in\mathbb{R}^d} \phi(\boldsymbol{w}) := \frac{1}{n}\sum_{i=1}^n \ell\left(\boldsymbol{x}_i^{\mathrm{T}}\boldsymbol{w}, y_i\right) = \frac{1}{n}\sum_{i=1}^n \left(y_i - \frac{1}{1+\exp(-\boldsymbol{x}_i^{\mathrm{T}}\boldsymbol{w})}\right)^2. \tag{2}$$

The function ϕ is $\mathcal{C}^2$ and it is nonconvex.

a) Justify that 0 is a lower bound on the function ϕ. Is it necessarily its optimal value?

b) We wish to apply the gradient descent algorithm to (2).

   i) Write the iteration of this algorithm with an arbitrary stepsize.

   ii) Give two possible choices for the stepsize.

   iii) Under appropriate assumptions, what is the complexity of the algorithm on a problem such as (2)? What quantity does this result apply to?

c) Suppose that gradient descent returns a point with a zero gradient. Is it necessarily a minimum?

d) State the second-order necessary optimality conditions for problem (2). Is a point satisfying these conditions a minimum?

e) Suppose that we start gradient descent from a random initial point, and that the method converges towards a point satisfying the second-order necessary optimality conditions. How can you explain this phenomenon?

Exercise 2: Convex matrix recovery

We consider a data matrix $\boldsymbol{X} \in \mathbb{R}^{d_1 \times d_2}$, and a subset $\mathcal{S} \subset \{1, \ldots, d_1\} \times \{1, \ldots, d_2\}$. The *matrix recovery* problem consists in finding the best approximation of $\boldsymbol{X}$ given some observed entries $\{\boldsymbol{X}_{ij} \mid (i,j) \in \mathcal{S}\}$. This amounts to solving the following optimization problem:

$$\underset{\boldsymbol{W} \in \mathbb{R}^{d_1 \times d_2}}{\text{minimize}} \ \frac{1}{2} \sum_{(i,j) \in \mathcal{S}} (\boldsymbol{W}_{ij} - \boldsymbol{X}_{ij})^2 \tag{3}$$

For any value of $\mathcal{S}$, the problem (3) can be reformulated as a vector optimization problem. Indeed, if we denote by $\boldsymbol{w} \in \mathbb{R}^d$ the concatenation of all columns of $\boldsymbol{W} \in \mathbb{R}^{d_1 \times d_2}$ (with $d = d_1 d_2$), problem (3) can be rewritten as

$$\underset{\boldsymbol{w} \in \mathbb{R}^d}{\text{minimize}} \ f(\boldsymbol{w}) := \frac{1}{2} \sum_{(i,j) \in \mathcal{S}} \left([\boldsymbol{w}]_{i+(j-1)d_1} - \boldsymbol{X}_{ij} \right)^2. \tag{4}$$

The objective function of problem (4) is convex and $\mathcal{C}^1$

a) The objective function of problem (4) is convex and $\mathcal{C}^1$.
   i) How can we characterize a solution of problem (4) using the derivative of f?
   ii) Give an example of a $\mathcal{C}^1$, convex function that does not possess a minimum.

b) The standard convergence rate of gradient descent on a convex problem such as (4) is $\mathcal{O}(\frac{1}{K})$. What quantity does this rate apply to?

c) What is the corresponding rate for accelerated gradient? What is the algorithmic idea behind this method?

d) We consider the special case in which all entries of the matrix are observed, i.e. $\mathcal{S} = \{1, \ldots, d_1\} \times \{1, \ldots, d_2\}$.
   i) In that case, the objective function of (3) (or, equivalently, that of (4)) is strongly convex. What can be said about local minima of strongly convex functions?
   ii) Justify that the problem (3) has a unique global minimum in the context of this question. What is this minimum?
   iii) When all entries are observed, the objective function f is a strongly convex quadratic function. Name one algorithm other than accelerated gradient that achieves a better convergence rate than gradient descent on this problem.

Exercise 4: Stochastic gradient

In this exercise, we consider a finite-sum minimization problem of the form :

$$\underset{\boldsymbol{w}\in\mathbb{R}^d}{\text{minimize}}\, f(\boldsymbol{w}) := \frac{1}{n}\sum_{i=1}^{n} f_i(\boldsymbol{w}), \tag{5}$$

where every function f_i is assumed to be $\mathcal{C}^1$ and depends solely on the ith element in a dataset $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^n$.

a) Why is the finite-sum structure amenable to applying stochastic gradient techniques?

b) Write the stochastic gradient iteration with a decreasing step size proportional to $\frac{1}{k+1}$, with k being the iteration index.

c) What is the cost of a stochastic gradient iteration in terms of accesses to the dataset? How does this compare to the cost of a gradient descent iteration?

d) Suppose that we perform K iterations of stochastic gradient and K iterations of gradient descent where $K \geq n$. We wish to compare the performance of both algorithms.

   i) Justify that comparing the values of f obtained for the final iterates of both methods is not a good metric.

   ii) Propose a relevant metric for comparing both methods without performing more iterations.

e) We now assume that the various items in the dataset are distributed across r processors, with r being a value between 1 and n.

   i) Write the iteration of a batch stochastic gradient method with a constant batch size equal to n_b, and a constant step size.

   ii) What can be the computational advantage of setting $n_b = r$?

   iii) If $r \approx n$, however, what is a possible drawback of using $n_b = r$?

   iv) If $1 < r \ll n$, setting $n_b = r$ corresponds to doing mini-batching. Does that necessarily lead to a better performance than $n_b = 1$? Justify your answer.

f) We finally consider an iteration of the Adam variant on stochastic gradient. Explain how this iteration differs from the vanilla stochastic gradient iteration.