Tutorial 07: Previous exam

M2 MIAGE ID Apprentissage

November 9, 2023

Dauphine | PSL
UNIVERSITÉ PARIS

Note: This is the exam given in December 2022. It was available in both an English version and a French version. Students could use any of those languages to write their paper. For sake of brevity, we only provide here the English version.

Foreword

In this exam, we explore several optimization problems that arise during the training of neural network architectures. Every exercise follows the same template by considering a dataset, a model/a neural architecture as well as an optimization problem representing training.

- Dimensions of vectors and matrices will always be assumed to be greater than or equal to 1.
- The notation $\|\cdot\|$ will be used for the Euclidean norm.
- Given a vector $\boldsymbol{u} \in \mathbb{R}^d$ with $d \geq 1$, the ith coordinate of this vector will be denoted by $[\boldsymbol{u}]_i$.
- For any integer $d \geq 1$, the zero vector in $\mathbb{R}^d$ will be denoted by $\mathbf{0}_{\mathbb{R}^d}$.

Exercise 1: Finite-sum problems

We consider the following optimization problem

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

where every $f_i : \mathbb{R}^d \to \mathbb{R}$ is $\mathcal{C}^1$.

a) We first study the solutions of this problem.

i) Give the definition of a global minimum for problem (1).

ii) Why are the solutions of problem (1) and that of the problem

$$\underset{\boldsymbol{w}\in\mathbb{R}^d}{\text{minimize}}\, f(\boldsymbol{w}) := \frac{1}{n}\sum_{i=1}^{n} f_i(\boldsymbol{w}) \tag{2}$$

identical?

iii) If f is a strongly convex function, what can be said about the minima of problem (2) (and thus about those of problem (1))?

b) In the rest of the exercise, we will focus on problem (2).
   i) Write down the gradient descent iteration with a constant stepsize for this problem.
   ii) Suppose that the function f is $\mathcal{C}_L^{1,1}$. In that case, how can the constant stepsize be chosen?
   iii) Give two other possible choices of stepsizes that are not constant ones.

c) We now consider that every f_i in the finite sum depends solely on the ith example of a dataset containing n examples.
   i) In terms of accesses to data points, what is the cost of an iteration of gradient descent applied to problem (2)?
   ii) Write down the stochastic gradient iteration applied to problem (2) with a constant stepsize.
   iii) Compare the cost of stochastic gradient in terms of accesses to data points with that of gradient descent as established in question c-i).

d) We now assume that the various examples of the dataset are distributed over r processors with r between 1 and n.
   i) Write down the batch stochastic gradient iteration with constant batch size equal to n_b and constant stepsize.
   ii) Recall the definition of an epoch. How many iterations of the method described in question d-i) does an epoch correspond to?
   iii) In the context of this question, what can be the advantage of choosing $n_b = r$?
   iv) If $r \approx n$, what can be the drawback of choosing $n_b = r$?

e) Suppose that the gradients ∇f_i are sparse. What variant of (batch) stochastic gradient could you use to benefit from that property, and why?

Exercise 2: Matrix completion

Let $\boldsymbol{X} \in \mathbb{R}^{d\times d}$ be a data matrix such that only a subset of its entries $\mathcal{S} \subset \{1,\dots,d\}^2$ are known with $|\mathcal{S}| = n \leq d^2$. We consider the problem

$$\underset{\boldsymbol{W}\in\mathbb{R}^{d\times d}}{\text{minimize}}\ f(\boldsymbol{W}) := \frac{1}{2n}\sum_{(i,j)\in\mathcal{S}} ([\boldsymbol{W}]_{ij} - [\boldsymbol{X}]_{ij})^2. \tag{3}$$

a) When $\mathcal{S} = \{1,\dots,d\}^2$, justify that $\boldsymbol{W}^* = \boldsymbol{X}$ is the unique solution of the problem.

b) Problem (3) is convex in the coefficients of $\boldsymbol{W}$. Letting $\boldsymbol{w} \in \mathbb{R}^{d^2}$ denoting the column vector formed by stacking all columns of the matrix $\boldsymbol{W}$ in order, we can reformulate the problem as

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

The function $\hat{f}$ is convex and $\mathcal{C}^1$.

i) What convergence rate guarantee can we provide on gradient descent when applied to problem (4)? What quantity does this rate apply to?

ii) What is the corresponding convergence rate for the accelerated gradient method due to Nesterov? Is it better than that of gradient descent?

iii) When $n = d^2$, the function $\hat{f}$ is a strongly convex quadratic function. Aside from Nesterov's method, what other approach can we use to obtain better convergence rates than gradient descent?

c) We now suppose that the data matrix $\boldsymbol{X}$ is symmetric, positive semidefinite and of rank $1 \ll d$. In this setting, rather than seeking an arbitrary matrix $\boldsymbol{W}$ to approximate $\boldsymbol{X}$, we can force the matrix to be rank one by writing it $\boldsymbol{u}\boldsymbol{u}^{\mathrm{T}}$ where $\boldsymbol{u} \in \mathbb{R}^d$. Problem (3) then becomes

$$\underset{\boldsymbol{u}\in\mathbb{R}^d}{\text{minimize}}\ \tilde{f}(\boldsymbol{u}) := \frac{1}{2n} \sum_{(i,j)\in\mathcal{S}} ([\boldsymbol{u}\boldsymbol{u}^{\mathrm{T}}]_{ij} - [\boldsymbol{X}]_{ij})^2. \tag{5}$$

The objective function of problem (5) is $\mathcal{C}^2$ and nonconvex.

i) State the first-order necessary optimality conditions for problem (5).

ii) What is the convergence rate of gradient descent for this problem? What quantity does this rate apply to?

iii) State the second-order necessary optimality conditions for problem (5).

iv) Under certain assumptions on $\boldsymbol{X}$ and $\mathcal{S}$, one can show that all the local minima of this problem are global. In that case, what technique guarantees almost surely that gradient descent will converge to such a point?

Exercise 3: Linear model

In this exercise, we consider a dataset under the form of a matrix $\boldsymbol{X} \in \mathbb{R}^{n\times d}$ and a vector $\boldsymbol{y} \in \mathbb{R}^n$. We seek a linear model of the data $\boldsymbol{x} \mapsto \boldsymbol{x}^{\mathrm{T}}\boldsymbol{w}$ parameterized by a vector $\boldsymbol{w} \in \mathbb{R}^d$, where we assume that $d \gg n$. Due to this last property, there exist numerous models that perfectly explain the data and satisfy $\boldsymbol{X}\boldsymbol{w} = \boldsymbol{y}$. Given a cost funtion $c : \mathbb{R}^d \to \mathbb{R}$, we consider the problem

$$\underset{\boldsymbol{w}\in\mathbb{R}^d}{\text{minimize}}\ c(\boldsymbol{w}) \quad \text{subject to} \quad \boldsymbol{X}\boldsymbol{w} = \boldsymbol{y}. \tag{6}$$

a) In this question, we assume that $c(\boldsymbol{w}) = \boldsymbol{c}^{\mathrm{T}}\boldsymbol{w}$ for $\boldsymbol{c} \in \mathbb{R}^d$.

i) Given this definition for c, which class of optimization problems does (6) belong to?

ii) What is the practical advantage of having to solve a problem of that class?

b) Suppose now that $c(\boldsymbol{w}) = \frac{1}{2}\|\boldsymbol{w}\|^2$.

i) Justify that the solution of (6) is of minimal norm among all solutions of $\boldsymbol{X}\boldsymbol{w} = \boldsymbol{y}$.

ii) Under the problem's assumptions, one can show that the vector $\boldsymbol{X}^\dagger \boldsymbol{y}$[1] is the unique solution of

$$\underset{\boldsymbol{w}\in\mathbb{R}^d}{\text{minimize}}\ \frac{1}{2}\|\boldsymbol{X}\boldsymbol{w}-\boldsymbol{y}\|^2. \tag{7}$$

with minimal norm. How can you use this information to find the solution and optimal value of problem (6)?

Exercise 4: Proximal gradient

We again consider a dataset formed by $\boldsymbol{X} \in \mathbb{R}^{n\times d}$ and $\boldsymbol{y} \in \mathbb{R}^n$. Given this dataset, we form a linear regression problem with so-called *elastic net* regularization :

$$\underset{\boldsymbol{w}\in\mathbb{R}^d}{\text{minimize}}\ \frac{1}{2n}\|\boldsymbol{X}\boldsymbol{w}-\boldsymbol{y}\|^2+\frac{\lambda_2}{2}\|\boldsymbol{w}\|^2+\lambda_1\|\boldsymbol{w}\|_1, \tag{8}$$

where $\lambda_2 \geq 0$ and $\lambda_1 \geq 0$.

a) What is the role of a regularization term in general?

b) What is the purpose of the regularization term when $\lambda_1 = 0$ and $\lambda_2 > 0$?

c) What is the purpose of the regularization when $\lambda_2 = 0$ and $\lambda_1 > 0$?

d) Recall that the gradient of the function $\phi : \boldsymbol{w} \mapsto \frac{1}{2n}\|\boldsymbol{X}\boldsymbol{w}-\boldsymbol{y}\|^2$ is given by

$$\nabla\phi(\boldsymbol{w}) = \frac{1}{n}\boldsymbol{X}^{\mathrm{T}}(\boldsymbol{X}\boldsymbol{w}-\boldsymbol{y}).$$

Using this formula, write down the iteration of proximal gradient for problem (8).

e) When $\lambda_2 = 0$ and $\lambda_1 > 0$, which algorithm is proximal gradient equivalent to?

f) When $\lambda_1 = 0$, one iteration of proximal gradient is equivalent to solving an optimization problem belonging to a specific class of problems. What is this problem class?

g) When $\lambda_1 > 0$ and $\lambda_2 > 0$, there does not exist an explicit formula for the proximal gradient iterates, and the proximal subproblem has to be solved approximately at every iteration. Propose an algorithm among those seen in class that could be employed to compute such an approximate solution, and justify your choice of that particular method.

[1] $\boldsymbol{X}^\dagger$ denotes the generalized inverse of $\boldsymbol{X}$. Its precise meaning is not needed to answer to question.

Solutions

Solutions to Exercise 1

a) (Problem solutions)

i) A point $\bar{\boldsymbol{w}} \in \mathbb{R}^d$ is a global minimum of problem (1) if

$$\sum_{i=1}^{n} f_i(\bar{\boldsymbol{w}}) \leq \sum_{i=1}^{n} f(\boldsymbol{w}) \quad \forall \boldsymbol{w} \in \mathbb{R}^d.$$

ii) The two problems are equivalent and have the same solution set. Indeed, since $\frac{1}{n} > 0$, for any global minimum $\bar{\boldsymbol{w}} \in \mathbb{R}^d$ of problem (1), we have

$$\begin{aligned} \textstyle\sum_{i=1}^{n} f_i(\bar{\boldsymbol{w}}) \quad &\leq \sum_{i=1}^{n} f(\boldsymbol{w}) \quad \forall \boldsymbol{w} \in \mathbb{R}^d \\ \Leftrightarrow \quad \tfrac{1}{n}\textstyle\sum_{i=1}^{n} f_i(\bar{\boldsymbol{w}}) \quad &\leq \frac{1}{n}\sum_{i=1}^{n} f(\boldsymbol{w}) \quad \forall \boldsymbol{w} \in \mathbb{R}^d, \end{aligned}$$

showing that $\bar{\boldsymbol{w}}$ is also a global minimum of (2). Since the relation above is an equivalence, the solution sets are indeed identical, i.e.

$$\operatorname*{argmin}_{\boldsymbol{w} \in \mathbb{R}^d} f(\boldsymbol{w}) = \operatorname*{argmin}_{\boldsymbol{w} \in \mathbb{R}^d} \sum_{i=1}^{n} f_i(\boldsymbol{w}).$$

iii) If f is strongly convex on $\mathbb{R}^d$, then it has a unique global minima.

b) Problem (2).

i) The kth iteration of gradient descent applied to this problem with a constant stepsize $\alpha > 0$ is

$$\boldsymbol{w}_{k+1} = \boldsymbol{w}_k - \alpha \nabla f(\boldsymbol{w}_k) = \boldsymbol{w}_k - \frac{\alpha}{n} \sum_{i=1}^{n} \nabla f_i(\boldsymbol{w}_k).$$

ii) When the function f is $\mathcal{C}_L^{1,1}$, one can choose $\alpha = \frac{1}{L}$ as constant stepsize. *NB: Any value in* $]0, \frac{2}{L}[$ *is correct.*

iii) Instead of using a constant stepsize throughout, one can use a predefined, decreasing stepsize sequence (e.g. $\alpha_k = \frac{\alpha_0}{k+1}$ with $\alpha_0 > 0$), or compute the stepsizes at every iteration in an adaptive fashion, for instance via backtracking line search.

c) Finite-sum structure

i) An iteration of gradient descent applied to problem (2) requires n accesses to data points.

ii) The kth iteration of stochastic gradient applied to this problem with a constant stepsize $\alpha > 0$ is

$$\boldsymbol{w}_{k+1} = \boldsymbol{w}_k - \alpha \nabla f_{i_k}(\boldsymbol{w}_k),$$

where i_k is an index drawn at random in $\{1, \ldots, n\}$.

iii) An iteration of stochastic gradient only accesses one data point, hence it is n times cheaper than an iteration of gradient descent in terms of this metric.

d) (Parallel calculations over $r \in \{1, \dots, n\}$ processors)

i) The kth iteration of batch stochastic gradient applied to this problem with a constant stepsize $\alpha > 0$ and batch size n_b is

$$\boldsymbol{w}_{k+1} = \boldsymbol{w}_k - \frac{\alpha}{n_b} \sum_{i \in \mathcal{S}_k} \nabla f_i(\boldsymbol{w}_k),$$

where $\mathcal{S}_k$ is a set of n_b indices drawn at random from $\{1, \dots, n\}$ (with or without replacement).

ii) An epoch is a unit of cost corresponding to n accesses to data points in a dataset of n elements. As a result, for a batch stochastic gradient method with batch size n_b, an epoch represents the cost of $\frac{n}{n_b}$ iterations.

iii) Choosing $n_b = r$ allows to compute each gradient from the batch in parallel, by distributing the calculations on the r processors.

iv) If $r \approx n$, choosing $n_b = r$ means that the method will operate in a large batch regime. Consequently, its behavior can be very similar to that of gradient descent, in that the method can converge significantly more slowly than vanilla stochastic gradient because the iterations remain expensive in terms of accesses to data points.

e) The variant `Adagrad` is particularly well suited for problems with sparse gradients, because it tunes the stepsize to every coordinate. *NB: With the same justification, `RMSProp` is a valid answer.*

Solutions to Exercise 2

a) The function $f(\boldsymbol{W})$ is always nonnegative (as a sum of squares, i.e. nonnegative numbers). When $n = d^2$, we have that

$$f(\boldsymbol{W}) = 0 \quad \Leftrightarrow \quad ([\boldsymbol{W}]_{ij} - [\boldsymbol{X}]_{ij})^2 = 0 \ \forall (i,j) \in \{1, \dots, d\}^2 \quad \Leftrightarrow \quad \boldsymbol{W} = \boldsymbol{X}.$$

As a result, the problem has a single global minimum given by $\boldsymbol{W}^* = \boldsymbol{X}$.

b) Convex formulation

i) Since the problem is convex, we know that after $K \geq 1$ iterations of gradient descent, the iterate $\boldsymbol{w}_K$ satisfies

$$\hat{f}(\boldsymbol{w}_K) - \min_{\boldsymbol{w} \in \mathbb{R}^{d^2}} \hat{f}(\boldsymbol{w}) \leq \mathcal{O}\left(\frac{1}{K}\right).$$

Gradient descent thus converges at a rate $\frac{1}{K}$.

ii) The rate for accelerated gradient on such a problem is $\frac{1}{K^2}$, which is a better rate as it converges more quickly to 0.

iii) When $\hat{f}$ is a strongly convex quadratic function, the heavy-ball method (aka Polyak's method) attains the optimal rate of convergence for strongly convex functions, which is better than gradient descent. *NB: The value of that rate is not required to answer the question.*

c) (Nonconvex case)

i) If $\bar{\boldsymbol{u}} \in \mathbb{R}^d$ is a local minima of problem (5), then $\nabla \tilde{f}(\bar{\boldsymbol{u}}) = \mathbf{0}$.

ii) For this problem, after $K \geq 1$ iterations of gradient descent, we have

$$\min_{0 \leq k \leq K-1} \|\nabla f(\boldsymbol{w}_k)\| \leq \mathcal{O}\left(\frac{1}{\sqrt{K}}\right),$$

hence the convergence rate of gradient descent is in $\frac{1}{\sqrt{K}}$.

iii) If $\bar{\boldsymbol{u}} \in \mathbb{R}^d$ is a local minima of problem (5), then $\nabla \tilde{f}(\bar{\boldsymbol{u}}) = \mathbf{0}$ and $\nabla^2 \tilde{f}(\bar{\boldsymbol{u}}) \succeq \mathbf{0}$.

iv) Initializing gradient descent with a random point guarantees almost surely that it will converge to a local minima under the assumptions of this question.

Solutions to Exercise 3

a) $c(\boldsymbol{w}) = \boldsymbol{c}^{\mathrm{T}} \boldsymbol{w}$

i) With that choice of function c, the optimization problem (6) is a linear program.

ii) Linear programs can be solved efficiently by state-of-the-art solvers, even with a large number of variables.

b) $c(\boldsymbol{w}) = \frac{1}{2}\|\boldsymbol{w}\|^2$.

i) Let $\bar{\boldsymbol{w}}$ be a solution of the problem. Then $\boldsymbol{X}\bar{\boldsymbol{w}} = \boldsymbol{y}$ by definition, and for any $\boldsymbol{v} \in \mathbb{R}^d$ such that $\boldsymbol{X}\boldsymbol{v} = \boldsymbol{y}$, we must have

$$c(\bar{\boldsymbol{w}}) \leq c(\boldsymbol{v}) \quad \Leftrightarrow \quad \|\bar{\boldsymbol{w}}\| \leq \|\boldsymbol{v}\|,$$

hence $\bar{\boldsymbol{w}}$ is of minimal norm among all solutions of the linear system. Conversely, if a point has minimal norm among all feasible points, then it is a solution.

ii) Since the linear system $\boldsymbol{X}\boldsymbol{w} = \boldsymbol{y}$ has a solution, the fact that $\boldsymbol{X}^\dagger \boldsymbol{y}$ solves problem (7) means that

$$\frac{1}{2}\|\boldsymbol{X}\boldsymbol{X}^\dagger \boldsymbol{y} - \boldsymbol{y}\|^2 = 0,$$

hence $\boldsymbol{X}^\dagger \boldsymbol{y}$ is feasible for problem (6). As a result, this vector is the unique point with minimal norm among all feasible points, hence it is the unique solution of problem (6). Finally, we obtain that

$$\min_{\boldsymbol{w} \in \mathbb{R}^d} \left\{ c(\boldsymbol{w}) = \frac{1}{2}\|\boldsymbol{w}\|^2 \middle| \boldsymbol{X}\boldsymbol{w} = \boldsymbol{y} \right\} = c(\boldsymbol{X}^\dagger \boldsymbol{y}) = \frac{1}{2}\|\boldsymbol{X}^\dagger \boldsymbol{y}\|^2.$$

Solutions to Exercise 4

a) A regularization term enforces a desired structure on the optimization variables.

b) When $\lambda_1 = 0$ and $\lambda_2 > 0$, the regularization term is an ℓ_2 regularization term, that aims at reducing the variance of the solution with respect to the data.

c) When $\lambda_2 = 0$ and $\lambda_1 > 0$, the regularization term is an ℓ_1 regularization term, that aims at promoting sparse solutions.

d) The kth iteration of proximal gradient applied to problem (8) is

$$\boldsymbol{w}_{k+1} \in \operatorname*{argmin}_{\boldsymbol{w}\in\mathbb{R}^d} \left\{ \phi(\boldsymbol{w}_k) + \frac{1}{n}(\boldsymbol{X}\boldsymbol{w}_k - \boldsymbol{y})^{\mathrm{T}}\boldsymbol{X}(\boldsymbol{w} - \boldsymbol{w}_k) + \frac{1}{2\alpha_k}\|\boldsymbol{w} - \boldsymbol{w}_k\|^2 + \frac{\lambda_2}{2}\|\boldsymbol{w}\|^2 + \lambda_1\|\boldsymbol{w}\|_1 \right\}$$

where $\alpha_k > 0$.

e) When $\lambda_1 = 0$, the proximal subproblem is a quadratic optimization problem (and even a linear least-squares problem).

f) A subgradient algorithm can be used to solve the subproblem, since it is defined even in the presence of a nonsmooth function such as the ℓ_1 norm. The iteration cost would involve computing a subgradient. Alternatively, one may want to apply proximal gradient to this subproblem while treating the ℓ_1 norm as a regularization term. This would correspond to the ISTA method (with a quadratic cost function), and would still be tractable given that the iterations of ISTA are explicitly defined.