Optimization for Machine Learning

M2 MIAGE ID Apprentissage

Project - 2024/2025

- The last version of this document can be found at:
  https://www.lamsade.dauphine.fr/∼croyer/ensdocs/OID/ProjOID.pdf.
- Typos, questions, etc, can be sent to `clement.royer@lamsade.dauphine.fr`.
- Current version: March 20, 2025.
  - 2025.03.20: Fixed a typo in question 1.1.c).
  - 2025.01.17: Second version of the document.
  - 2024.12.20: First version of the document (discussed in class).

Assignment

- This project follows the template of the course notebooks by mixing optimization questions with implementation tasks.
- Students may discuss the project with their classmates, but they should submit the project in groups of n students with $n \in \{1, 2\}$.
- Students may submit their sources in either French or English.
- Students are expected to submit sources that include:
  - Their answers to the questions (in PDF or notebook format).
  - A Python script or a notebook to run the methods and reproduce the results.
- Please send your sources to `clement.royer@lamsade.dauphine.fr` under the form of a compressed folder. Those sources must include the first and last name of every member of the group.
- The deadline to send the sources is **March 24, 2025 AOE** (Anywhere On Earth).

Project: Regularization and optimization

Introduction

In the course, we covered problems of the form

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

where each f_i depends on a certain sample from a dataset of size n. The function f is called a *data-fitting* term, since it quantifies how good a model is with respect to fitting the data.

Standard practice in optimization for machine learning consists in *regularizing* a problem, i.e. replacing problem (1) by

$$\underset{\boldsymbol{w} \in \mathbb{R}^d}{\text{minimize}}\, f(\boldsymbol{w}) + \lambda r(\boldsymbol{w}), \tag{2}$$

where $\lambda > 0$ and $r : \mathbb{R}^d \to \mathbb{R}$ is a *regularization term* used to enforce specific properties on the solution. This is particularly useful when problem (1) has multiple solutions, but regularization also helps in a less direct way, e.g. by improving the performance of the model corresponding to the solution of (1) on unseen data.

In this project, we consider ℓ_2 regularization, arguably the most classical regularization choice in the literature. Our final goal is to assess the interest of ℓ_2 regularization for improving generalization.

1 Optimization with ℓ_2 regularization

Given a (data-fitting) function $f : \mathbb{R}^d \to \mathbb{R}$, we consider the family of problems

$$\operatorname*{minimize}_{\boldsymbol{w} \in \mathbb{R}^d} f(\boldsymbol{w}) + \frac{\lambda}{2}\|\boldsymbol{w}\|^2, \tag{3}$$

where $\|\boldsymbol{w}\|^2 = \sum_{j=1}^d w_j^2$ and $\lambda \geq 0$ is a regularization parameter. When $\lambda = 0$, we recover the original problem (1). When $\lambda \to \infty$, one can show that the problem (3) is equivalent to

$$\operatorname*{minimize}_{\boldsymbol{w} \in \mathbb{R}^d} \frac{1}{2}\|\boldsymbol{w}\|^2. \tag{4}$$

Question 1.1 *Consider the toy function* $f^{toy}(\boldsymbol{w}) = \frac{1}{4}\|\boldsymbol{X}\boldsymbol{w} - \boldsymbol{y}\|^2$ *with*

$$\boldsymbol{X} = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \quad \textit{and} \quad \boldsymbol{y} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}.$$

As seen in class, this function is $\mathcal{C}^1$ *and convex.*

a) Consider the problem

$$\operatorname*{minimize}_{\boldsymbol{w} \in \mathbb{R}^d} f^{toy}(\boldsymbol{w}). \tag{5}$$

Show that f^{toy} *has an infinite number of global minima, and give a characterization of the set of global minima for* f^{toy}.

b) Consider the problem

$$\operatorname*{minimize}_{\boldsymbol{w} \in \mathbb{R}^d} f^{toy}(\boldsymbol{w}) + \frac{\lambda}{2}\|\boldsymbol{w}\|^2 \tag{6}$$

with $\lambda > 0$. *Show that problem (6) has a unique solution.*

c) Suppose now that we are given a new data sample $\left(\boldsymbol{x}^+ = \begin{bmatrix} -1 \\ 0 \end{bmatrix}, y^+ = 1\right)$. *Given a global minimum* $\boldsymbol{w}_0$ *for* $\operatorname{minimize}_{\boldsymbol{w} \in \mathbb{R}^d} f^{toy}(\boldsymbol{w})$ *and the global minimum* $\boldsymbol{w}_\lambda$ *for (6), show that there is a range of values for* λ *such that*

$$\frac{1}{2}([\boldsymbol{x}^+]^\mathrm{T}\boldsymbol{w}_\lambda - y^+)^2 < \frac{1}{2}([\boldsymbol{x}^+]^\mathrm{T}\boldsymbol{w}_0 - y^+)^2.$$

Question 1.2 *The gradient of* $\boldsymbol{w} \mapsto \frac{\lambda}{2}\|\boldsymbol{w}\|^2$ *is* $\boldsymbol{w} \mapsto \lambda\boldsymbol{w}$. *Using this expression, write down the iteration of gradient descent applied to problem (3) with a generic stepsize. How does this iteration change as* λ *increases? How does this illustrate the impact of regularization at every iteration?*

2 Experiments on synthetic data

In this part, we will consider the synthetic data used in the course notebooks, in the context of linear regression.

Question 2.1 *Using the functions given in the lab sessions, generate two datasets for linear regression based on the same ground truth vector.*

Question 2.2 *Run gradient descent with and without regularization on the first instance, then compute the objective function corresponding to the second instance. Do you observe that using regularization in the first instance improves the objective function of the second instance?*

Question 2.3 *Reproduce the experiment of Question 2.2 using (batch) stochastic gradient instead of gradient descent.*

3 Binary classification on real-world data

In this final part, we will apply our techniques to a classification problem based on real-world data.

3.1 Dataset

We will rely on datasets from the libsvm repository, that can be downloaded from

https://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/

Recommended datasets are `a9a`,`covtype.binary`,`ijcnn1`. To load the dataset in Python, students may use the routine

`sklearn.datasets.load_svmlight_file`

from the scikit-learn library.

Implementation 3.1 *Select a dataset from the libsvm repository. The dataset should have at least 20 features and 1,000 training samples. It should also have both a training set (used for optimization) and a testing set.*

3.2 Optimization problem

Let $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^n$ denote the samples from the training set of the dataset. We formulate the following finite-sum optimization problem

$$\operatorname*{minimize}_{\boldsymbol{w}\in\mathbb{R}^d} g(\boldsymbol{w}) := \frac{1}{n}\sum_{i=1}^n g_i(\boldsymbol{w}), \quad g_i(\boldsymbol{w}) := \left(y_i - \frac{1}{1+\exp(-\boldsymbol{x}_i^{\mathrm{T}}\boldsymbol{w})}\right)^2, \tag{7}$$

where This problem is nonconvex in general. The function $t \mapsto \frac{1}{1+\exp(-t)}$ is called the sigmoid function. For any $i = 1, \dots, n$, the function g_i is $\mathcal{C}^1$, with its gradient being given by

$$\nabla g_i(\boldsymbol{w}) = -\frac{2\exp(\boldsymbol{x}_i^{\mathrm{T}}\boldsymbol{w})\left(\exp(\boldsymbol{x}_i^{\mathrm{T}}\boldsymbol{w})(y-1)+y\right)}{(1+\exp(\boldsymbol{x}_i^{\mathrm{T}}\boldsymbol{w}))^3}\boldsymbol{x}_i. \tag{8}$$

Implementation 3.2 *Given your dataset, implement the associated codes for $g_{\mathcal{S}}$, $\nabla g_{\mathcal{S}}$ where $\mathcal{S}$ is a set of random indices in $\{1, \dots, n\}$.*

In addition to the optimization problem above, we will be interested in the *generalization capabilities* of the model obtained by solving problem (7). Let $\{\tilde{\boldsymbol{x}}_i, \tilde{y}_i\}_{i=1}^m$ denote the samples from the **testing set**. Given a vector $\boldsymbol{w} \in \mathbb{R}^d$, our goal is to obtain a good value for

$$\tilde{g}(\boldsymbol{w}) = \frac{1}{m} \sum_{i=1}^m \left(\tilde{y}_i - \frac{1}{1 + \exp(-\tilde{\boldsymbol{x}}_i^{\mathrm{T}} \boldsymbol{w})} \right)^2 . \tag{9}$$

Since the training and testing datasets are supposed to originate from the same distribution, we expect that a solution to problem (7) will have good performance on the testing dataset, i.e. will yield a low value of $\tilde{g}(\boldsymbol{w})$.

3.3 Comparison of the algorithms

Our goal is now to assess the interest of ℓ_2 regularization for this problem. We will thus consider the family of problems

$$\operatorname*{minimize}_{\boldsymbol{w} \in \mathbb{R}^d} g(\boldsymbol{w}) + \frac{\lambda}{2} \|\boldsymbol{w}\|^2. \tag{10}$$

Implementation 3.3 *Implement (batch) stochastic gradient for problem (10), and run the method for several values of the regularization parameter.*

Question 3.1 *Compare the final values of the data-fitting term g for all methods, as well as that of the testing loss $\tilde{g}$. Can you find a value for λ that improves the testing error compared to $\lambda = 0$?*