MiniProject 2: Optimization and Text Classification

COMP 551, Fall 2021, McGill University

October 7, 2021

Please read this entire document before beginning the assignment

Preamble

- **Lead TA's**: Haque Ishfaq, Lucas Caccia
- This mini-project is due on **October 21st** at 11:59pm EST. Late work will be automatically subject to a 20% penalty, and can be submitted up to 5 days after the deadline. No submissions will accepted after this 5 day period.
- This mini-project is to be completed in groups of three. All group members should be familiar with all steps involved in the mini-project, to avoid potential failure of the entire project due to one member's failure to do their work. All members of a group will receive the same grade. It is not expected that all team members will contribute equally to all components. However every team member should make integral contributions to the project.
- Submit your assignment on MyCourses as a group. You must register your group on MyCourses and any group member can submit. See MyCourses for details.
- You are free to use libraries with general utilities, such as matplotlib, numpy, scipy, pandas and sklearn for Python.

Background

The goal of this project is twofold. First, we want you to gain some insight into the optimization process of gradient based methods. In particular, this section will ask you to implement different variations of gradient descent and analyze their impact on the convergence speed. In the second part of the assignment, you will be experimenting with text data. The section encourages you to explore text-specific preprocessing techniques, and to reapply concepts such as cross-validation that are central to machine learning. In both sections, you will be performing binary classification with Logistic Regression.

Part 1 : Optimization (80 points)

In this section you will be using the **diabetes** dataset, which you can find in the assignment folder. For this part, you do not need to perform any data cleaning or preprocessing, i.e. you can use the dataset *as-is*. For this section, we encourage you to leverage the Logistic Regression notebook, which you can find here. For each bullet point, make sure to include clear and concise plots to go with your discussion.

1. You should first start by running the logistic regression code using the given implementation. This will serve as a baseline for the following steps. Find a learning rate and a number of training iterations such that the model has fully converged to a solution. Make sure to provide empirical evidence supporting your decision (e.g. training and validation accuracy as a function of number of training iterations).
2. Implement *mini-batch stochastic gradient descent*. Then, using growing minibatch sizes (e.g. 8, 16, 32, ...) compare the convergence speed and the quality of the final solution to the fully batched baseline. What configuration works the best among the ones you tried ?
3. Add momentum to the gradient descent implementation. Trying multiple values for the momentum coefficient, how does it compare to regular gradient descent ? Specifically, analyze the impact of momentum on the convergence speed and the quality of the final solution.

4. repeat the previous step for a) the smallest batch size and b) largest batch size you tried in 2). In which setting (small mini-batch, large mini-batch, fully batched) is it the most / least effective ?

Part 2 : Text Classification (20 points)

In this part, you will be using the **fake news** dataset. The goal is to detect which articles are generated by a computer, and which ones are written by humans. The dataset has already been split into training, validation, and test. No preprocessing has been applied. A good place to start is the sklearn text data tutorial. For this part, we recommend using the sklearn's Logistic Regression package as your base model.

Get a basic version working. This includes building a preprocessing pipeline to map raw text to features on which you can train a model. You can go above and beyond, for example, to see if you can achieve more than 80% on the test set.

Deliverables

You must submit two separate files to MyCourses **(using the exact filenames and file types outlined below)**:

1. **code.zip**: Your entire code, which should consist of a jupyter notebook file (.ipynb), and additional python files (.py); **the notebook should contain the main body of your code, where we can see and easily reproduce the plots in your report.**

2. **writeup.pdf**: Your (max three pages) project write-up as a pdf (details below).

Project write-up

Your team must submit a project write-up that is a maximum of three pages (**single-spaced, 11pt font or larger; minimum 1 inch margins, an extra page for references/bibliographical content can be used**). We highly recommend that students use LaTeX to complete their write-ups. You have some flexibility in how you report your results, but you must adhere to the structure discussed in the first assignment. As before you can also have a statement of the breakdown of the workload across the team members.

Evaluation

The mini-project is out of 10 points, and the evaluation breakdown is as follows:

- Completeness (2 points)
  - Did you submit all the materials?
  - Did you run all the required experiments?
  - Did you follow the guidelines for the project write-up?
- Correctness (4 points)
  - Are your models used/implemented correctly?
  - Are you visualizations informative and visually appealing?
  - Are your reported accuracy close to (our internal) reference solutions?
  - Are you observing the expected trends?
- Writing quality (2.5 points)
  - Is your report clear and free of grammatical errors and typos?
  - Did you go beyond the bare minimum requirements for the write-up (e.g., by including a discussion of related work in the introduction)?
  - Do you effectively present numerical results (e.g., via tables or figures)?
- Originality / creativity (1.5 points)
  - Did you go beyond the bare minimum requirements for the experiments?

  - within the context of producing the required results did you propose a creative idea?
  - **Note:** Simply adding in a random new experiment will not guarantee a high grade on this section! You should be thoughtful and organized in your report in explaining why you performed an additional experiment and how it helped in evaluating your hypothesis.

Final Remarks

You are expected to display initiative, creativity, scientific rigour, critical thinking, and good communication skills. You don't need to restrict yourself to the requirements listed above - feel free to go beyond, and explore further.

You can discuss methods and technical issues with members of other teams, but **you cannot share any code or data with other teams**.