EE 518: Machine Learning Theory & Algorithms **Spring 2024**

Mini Project 1

Due: April 26, 2024, 11:59PM PT

Student Name: *Instructor Name: John Lipor*

Introduction

Most homework problems encountered in machine learning courses are well-formulated problems on nicely curated datasets. However, real world problems are never this nice. First, the actual task at hand is typically poorly defined (e.g., "here's my data, tell me something from it"). Second, even when a task is clearly defined, datasets contain features of different types (binary, real-valued, categorical, location), missing data, and outliers. One aim of these mini projects is to acquaint you with the task of dealing with real problems.

One popular source of data-centric problems is Kaggle. Beyond containing a large number of interesting datasets, Kaggle also hosts competitions that often give prizes and allows users to post their solutions via Jupyter notebooks. In this project, you will work with the Titanic dataset [1], which is the most popular introductory dataset on Kaggle. The task for this dataset is clearly defined: use the given features to determine which passengers survived the Titanic shipwreck. On the other hand, this is a messy dataset, which will give you the experience with data preprocessing needed to work on more interesting datasets in the future.

Your main task is to achieve the best performance you can on the Titanic dataset using the least squares and logistic regression classifier you implemented. Several people on Kaggle have achieved perfect scores on this dataset, but these likely use nonlinear methods like the ones we'll begin covering next week. Your task is to get the best possible error with a linear classifier.

Notes:

1. DSS rules apply for all mini projects.
2. Be sure to cite any sources you used, including/especially Kaggle notebooks.
3. **LLMs:** You may use an LLM (e.g., ChatGPT) to aid with this project. However, for any section that you find an LLM useful, you must specify how you used it. For example, you may need to state something like, "ChatGPT helped me understand the components of exploratory data analysis, and I used some lines of generated code to complete this task." You also must understand every line of code used in your project. **Please ask clarifying questions** in the `#homework-questions` channel.

Requirements

You must create a report in either LaTeXor a Jupyter notebook that contains the following sections:

1. Problem description
    - Explain what data you have to work with, what algorithms you will use, what your goal is, and why anyone should care.

2. Exploratory data analysis (EDA)
    - There are many excellent tutorials on EDA online. Be sure to cite whichever ones you found helpful and comment on them in Sec. 6.
    - Be sure to make any significant findings stand out and try to keep this concise.
3. Challenges
    - What were the challenges you encountered when applying machine learning to this dataset?
    - Did these challenges mainly result from the data? From results? Installing libraries to perform preprocessing?
4. Approach
    - Provide a thorough but concise description of your approach, including (but not limited to) your approach to wrangling, preprocessing, and improving the performance of your linear classifiers.
    - This should be much clearer than what is typically found in Kaggle notebooks, but make sure it stays concise. A summary paragraph at the beginning of this section could be helpful.
5. Evaluation and summary
    - Make use of any meaningful metrics (e.g., classification error, precision, recall, confusion matrix, f1 score), but only include metrics that tell you something different.
    - Consider diving into the data to see if you can determine any trends regarding which points were misclassified.
    - Likewise consider if there were any features that were particularly important or unimportant.
    - Summarize your solution, describing what worked, what didn't, what the main limitations are (e.g., if nonlinear classifiers would improve results), and any general conclusions about the dataset.
6. What I learned
    - Describe the main skills/tools that you learned and used for this project and how you learned them.

Grading

The grading breakdown is below. Each section will be graded according to technical correctness, effort, and creativity. Note that clarity of writing is a major component. You should put yourself in the place of writing for a boss or senior in a workplace. If your writing is terrible, they will assume your work is terrible.

Item	**Percentage**
Description	5%
EDA	20%
Challenges	10%
Approach	30%
Evaluation	20%
What I learned	5%
Clarity/conciseness of written communication	10%

Advice

- You may add a bias term to the LS and LR classifiers but you must implement this yourself.
- You may make use of linear dimensionality reduction methods such as PCA or dropping features when classifying.
- While you are restricted to linear methods for classification, you may make use of nonlinear embedding/dimensionality reduction methods such as t-SNE or UMAP for visualization. You could also use these as evidence that a nonlinear classifier would perform better.
- You may find articles on "data wrangling" and "exploratory data analysis" useful. Some examples are [2, 3]. Please feel free to share these in the `#interesting-reading` channel.
- If time permits, explore notebooks from other datasets to give you ideas as well. A great example of really thinking through the results of an algorithm is given in [4].

References

[1] (2018) Titanic: Machine learning from disaster. [Online]. Available: https://www.kaggle.com/c/titanic/overview

[2] (2023) Data wrangling in python. [Online]. Available: https://www.geeksforgeeks.org/data-wrangling-in-python/

[3] (2023) Exploratory data analysis in python. [Online]. Available: https://www.geeksforgeeks.org/exploratory-data-analysis-in-python/

[4] (2019) Ml explainability: Deep dive into ml model! [Online]. Available: https://www.kaggle.com/niyamatalmass/ml-explainability-deep-dive-into-the-ml-model