IFT 6390 Fundamentals of Machine Learning
Ioannis Mitliagkas

Kaggle Competition 2

November 16, 2023

1 Introduction

For this project, you will participate in a Kaggle competition on classification of sign language based on images of the hand. The goal of this competition is to correctly classify two given sign language images into their corresponding alphabet and then sum their respective ASCII values to provide the resultant final character corresponding to the summed ASCII value.

The competition aims to provide hands-on experience in a real-world problem of interpreting sign language from images with an extra twist of doing ASCII manipulation on them.

The data set that we will use for this competition is Sign Language MNIST. The dataset format is patterned to closely match the classic MNIST. Each training case represents a label (0-25) as a one-to-one map for each alphabetic letter A-Z (and no cases for 9=J or 25=Z because of gesture motions). The training data (27,455 cases) and test data (3000 cases) are approximately half the size of the standard MNIST but otherwise similar with a header row of label, pixel1,pixel2....pixel784, which represents a single 28x28 pixel image with grayscale values between 0-255. The original hand gesture image data represented multiple users repeating the gesture against different backgrounds. More details about the dataset and structure can be found on the "Data" page of the Kaggle competition.

The goal of this project is to implement and train several classification algorithms and further build the correct logic for the required ASCII manipulation. The evaluation will be based on the performance on the held out test set and a written report. The competition, including the data, is available here:

https://www.kaggle.com/t/26a859d3790f4c7b8ed61dd724c378ee

2 Important Dates

Please take into consideration the following important deadlines:

- November 20th 23:59 Deadline to enter the competition on Kaggle.
- December 5th 23:59 Competition ends. No more Kaggle submissions are allowed.
- December 12th 23:59 Reports and code are due on Gradescope.

Note on sharing and plagiarism: You are allowed to discuss general techniques with other teams. You are NOT allowed to share any of your code. This behavior constitutes plagiarism, and it is very easy to detect. All teams involved in the sharing of code will receive a grade of 0 in the data competition.

3 Join the Competition

IFT6390 students can work individually or in teams of two.(**IFT3395** students will work in teams of 2 or 3). Create a Kaggle account if you are not registered, and join the competition by following this link: https://www.kaggle.com/t/26a859d3790f4c7b8ed61dd724c378ee

Please register your Kaggle username in this form: https://forms.gle/uWCBysE9yPGmcarG7

Important note: The maximum amount of submissions per day is 3.

4 Baselines

We ask you to build a **random forest** and a **neural network** (CNN) classifier and beat the baselines highlighted in the leaderboard. These baselines are:

1. a dummy classifier initialized with random parameters.

2. a random forest classifier.

3. the best baseline of the TA.

To participate in the competition, you must provide a list of predicted outputs for the instances on the Kaggle website. You can submit 3 predictions per day over the course of the competition, so we suggest you start early, allowing yourself enough time to submit multiple times and get a sense of how well you are doing. For each of the 3 baselines that you beat, you get extra points. Your random forest classifier must be implemented from scratch. Apart from standard Python libraries, the only libraries allowed are numpy, sparse matrices from scipy (scipy.sparse), and pandas.

5 Other Methods

You must try at least **1 other model** in addition to the random forest and the neural network (CNN), and compare their performance. You are encouraged to implement techniques studied during the course and to look for other ways to solve this task. Here are a few ideas:

1. Support Vector Machines

2. Adaboost

3. XGBoost

The goal is to design the best performing method as measured by submitting predictions for the test set on Kaggle. Your final performance on Kaggle will count as a criterion for evaluation (see Section 6). If a tested model does not perform well, you can still add it in your report and explain why you think it is not appropriate for this task. This kind of discussion is an important feature that we will use to evaluate your final competition report.

For this part, you are free to use any library of your choice.

6 Report

In addition to your methods, you must write up a report that details the preprocessing, validation, algorithmic, and optimization techniques, as well as providing results that help you compare different methods/models. The report should contain the following sections and elements:

- Project title
- Full name, student number and Kaggle username.
- Introduction: briefly describe the problem and summarize your approach and results.
- Feature design: Describe and justify your pre-processing methods, and how you designed and selected your features.
- Algorithms: give an overview of the learning algorithms used without going into too much detail, unless necessary to understand other details.
- Methodology: include any decisions about training/validation split, regularization strategies, any optimization tricks, choice of hyperparameters, etc.
- Results: present a detailed analysis of your results, including graphs and tables as appropriate. This analysis should be broader than just the Kaggle result: include a short comparison of the most important hyper-parameters and all methods (at least 2) you implemented.
- Discussion: Discuss the pros/cons of your approach and methodology and suggest ideas for improvement.
- Statement of contributions. add the following statement: 'I hereby state that all the work presented in this report is that of the author'.
- References: very important if you use ideas and methods that you found in some paper or online; it is a matter of academic integrity.
- Appendix (optional). Here you can include additional results, more details of the methods, etc.

You will lose points if you do not follow these guidelines. **The main text of the report should not exceed 6 pages**. The references and Appendix can be in excess of six pages. You must submit your report and your code on Gradescope before December 12th 23:59.

7 Submission Instructions

- You must submit the code developed during the project. The code must be well documented. The code should include a README file containing instructions on how to run the code.
- The prediction file containing your predictions on the test set must be submitted online at the Kaggle website.
- The report in pdf format (written according to the general layout described above) and the code should be uploaded on Gradescope.

8 Evaluation Criteria

Marks will be attributed according to the following criteria:

- You will be assigned points for each one of the 3 baselines that you beat.
- You will be assigned points depending on your final performance at the end of the competition. The ranking that you can see is computed using 23% of the test set. The remaining 77% is used to compute a private ranking (which is not visible). Therefore, you can be ranking first on the public ranking but not on the private one. Your grade will be computed using both rankings.
- You will be assigned points depending on the quality and technical soundness of your final report (see above).