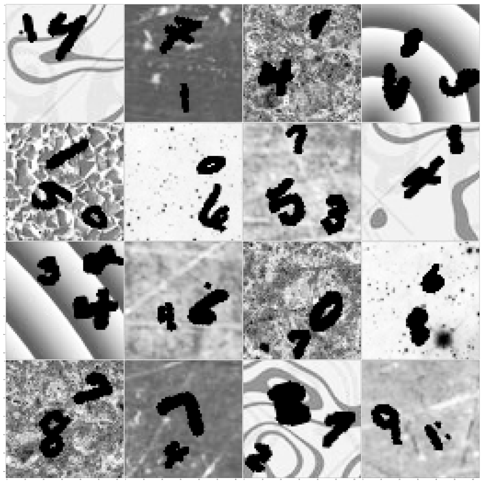
COMP-551: Applied Machine Learning

Project #4: Find the “Largest” Digit

Due March 21, 11:59pm EST

Background

For this project, you will take part in a Kaggle competition based on image analysis. The goal is to design a machine learning algorithm that can automatically identify hand-written digits as well as reason about their appearance. The dataset we have prepared is a variant of the classic MNIST dataset. For that dataset, a popular goal has been to simply identify the given hand-written digit. For our variant, we’ve randomly generated various grey-scale images containing two or three digits with different sizes, randomly scaled to 40/60/80/100/120 percent of the original digit size. The correct label for each image corresponds to the digit with the maximum area. To be more precise, it is based on the area of the rectangle which encompasses the digit. The dataset consists of 50k grayscale images of size (64,64) for the training and 10k for validation. Examples of the training samples are shown here:



The competition, including the data, is available here:

https://www.kaggle.com/c/comp551w18-modified-mnist

We expect you to be working in groups of 3 (strict maximum), with the restriction that the group members must be in the same sections. In addition, do note that you cannot work with any of the same group members for the final project.

Instructions

To participate in the competition, you must provide a list of predicted outputs for the instances on the Kaggle website. To solve the problem, we expect you to try the following methods:

- A baseline linear learner consisting of SVM or logistic regression, implemented by hand or using a library.
- A fully connected feed forward neural network trained by backpropagation, where the architecture of the network (number of nodes, layers, learning rate, etc.) are determined by cross-validation. This must be fully implemented by hand, and the corresponding code should be submitted. You are, however, allowed to use algebra libraries (e.g. numpy).
- Any other ML method of your choice. Be creative! Some suggestions are k-NN, random forests, kernalized SVM, CNN's, etc.

  For the Kaggle competition, you can submit results from you best performing system, whichever method (from the above three categories) it may fall under. You are allowed to use supplementary data to enrich the training set though you must provide references in the report. Note that you can submit predictions multiple times, so we suggest you *start early* and *submit your first model early* so you know how well you are doing.

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 compare them. The report should contain the following sections and elements:

- Project title.
- Section number (551-001 or 551-002), Team name on Kaggle, as well as the list of team members, including their full name, McGill email and student number.
- 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 in the class notes (e.g. SVM derivation, etc.), unless necessary to understand other details.

- Methodology: Include any decisions about training/validation split, distribution choice for naïve bayes, regularization strategy, any optimization tricks, setting hyper-parameters, 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 hyperparameters and all 3 methods you implemented.

- Discussion: Discuss the pros/cons of your approach & methodology and suggest areas of future work.

- Statement of Contributions. Briefly describe the contributions of each team member towards each of the components of the project (e.g. defining the problem, developing the methodology, coding the solution, performing the data analysis, writing the report, etc.) At the end of the Statement of Contributions, add the following statement: "We hereby state that all the work presented in this report is that of the authors."

- References (optional).

- Appendix (optional). Here you can include additional results, more detail of the methods, etc.

  The main text of the report should not exceed 6 pages. References and appendix can be in excess of the 6 pages. The format should be double-column, 10pt font, min. 1" margins. You can use the standard IEEE conference format, e.g. https://www.ieee.org/conferences_events/ conferences/publishing/templates.html

Submission Requirements

- You must submit the code developed during the project. The code can be in a language of your choice. The code must be well-documented. The code should include a README file containing instructions on how to run the code. Submit the code as an attachment (see *Submission Instructions*).

- The prediction file must be submitted online at the Kaggle website.

- You must submit a written report according to the general layout described earlier

Submission Instructions

For this project, you will submit all your materials on MyCourses.

- Submit a single zipped folder with your McGill id as the name of the folder. For example if your McGill ID is 12345678, then the submission should be 12345678.zip.
- Your zip file should contain the following:
  1. Your report stored as report.pdf
  2. A folder called code which contains all code and data
- Make sure all the data files needed to run your code is within the folder and loaded with relative path. We should be able to run your code without making any modifications.

Once the deadline expires, you will not be able to submit files. If you are submitting the project late (up to one week; subject to automatic 30% penalty), submit your code to a special folder on myCourses for late submissions.

Evaluation Criteria

Marks will be attributed based on: 30% for performance on the private test set in the competition, 70% for the written report. The code will not be marked, but will be used to validate other components. For the competition, the performance grade will be calculated as follows: The top team, according to the score on the private test set, will receive 100%. A random predictor, entered by the instructor, will score 0%. All other grades will be calculated according to interpolation of the private test set scores between those two extremes.

For the written report, the evaluation criteria include:

- Technical soundness of the methodology (pre-processing, feature selection, validation, algorithms, optimization).
- Technical correctness of the description of the algorithms (may be validated with the submitted code).
- Meaningful analysis of final and intermediate results.
- Clarity of descriptions, plots, figures, tables.
- Organization and writing. Please use a spell-checker and don't underestimate the power of a well-writen report!!

Do note that the grading of the report will place emphasis on the quality of the implemented linear and non linear classifiers as well as the rationale behind the pre-processing and optimization techniques. The code should be clear enough to reflect the logic articulated in the report. We are looking for a combination of insight and clarity when grading the reports.

Exact Deadlines

MyCourses submission closing March 21, 11:59pm EST.
Kaggle submission closing March 21, 11:59pm EST (=4:59am UTC on next day)

Questions and clarifications

For questions, please use the following channels:

- The course discussion forum.
- For more detailed questions, please go to the office hours of the following TAs: Philip, Koustuv (Section 1); Harsh, Ali (Section 2).