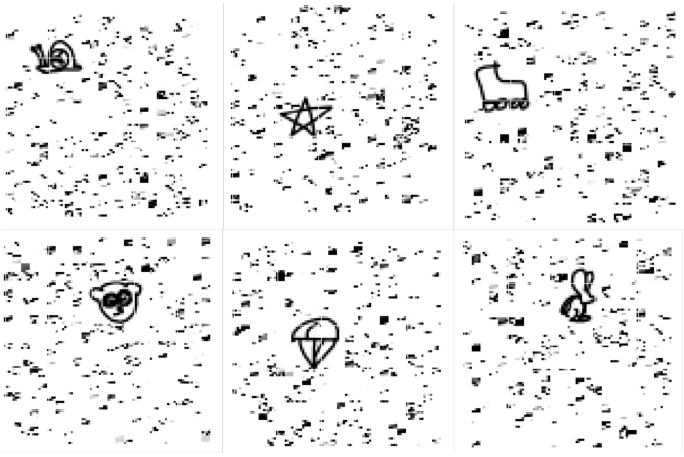
COMP 551 Fall 2018

Kaggle Competition

Due November 26, 8pm EST

1 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 drawn images as well as reason about their appearance. The dataset we have prepared is a variant of google's quick draw dataset. For that dataset, a popular goal has been to simply identify the given human drawn images. For our variant, we've reduced the number of classes to 31. To create an image, we've appended the drawing to a random location on a larger 100x100 pixel blank canvas image. Additionally, we've added randomly generated noise around the drawing. Note that one of the classes is called "empty" and consists of only noise and no human drawings. The dataset consists of 10k images of size (100,100) for the training set and 10k for test set. You will be evaluated on test accuracy. Examples of the training samples are shown here:



The competition, including the data, is available here:
`https://www.kaggle.com/c/f2018-hand-drawn-pictures`

We expect you to be working in groups of exactly 3. In addition, do note that you cannot work with any of the same group members for the final project.

2 Kaggle Team formation

Each team should consist of exactly 3 members. To form a team:

- Fill out the google form `https://goo.gl/forms/0IXrePHebFfOFB6c2` with the your team information by Nov 5^{th} at 5:00 pm. Any teams not registered or registered late will not be graded.
- Register as an individual Kaggle user
- Enter the competition and accept terms and conditions.
- Go to `https://www.kaggle.com/c/f2018-hand-drawn-pictures/team`
- In the "Invite Others" section, enter your teammates' names, or team name.
- Your teammate has the option to accept your merge. The person accepting a merge is the team leader.

**** IMPORTANT NOTE ****
The maximum amount of submissions is 2 per day, per TEAM. Any team whose individual members have a submission count larger than what is allowed up to-date will be UNABLE to form a team.

Example: Today is the first day of competition. A,B,C are three teammates who haven't formed a team yet.
A submitted 0 times.
B submitted 2 times.
C submitted 1 times.

Because the maximum amount of submissions is 2 per team per day, the total possible submissions for a team is 2. However, the cumulative submission count for A,B,C is 3. Therefore, they will be unable to form a team (They will need to wait for tomorrow, and not submit any submissions for the next day).

Any members at the end of competition that are not in a team will not receive any marks. We suggest forming your teams BEFORE attempting any solutions.

3 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, CNNs, 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 not allowed to use any supplementary data to enrich the training set. Note that there is a maiximum of 2 prediction submissions per day on Kaggle. You can submit 2 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.

4 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
- 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 naive 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 doublecolumn, 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

5 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.

6 Submission Instructions

For this project, you will submit the report to Gradescope, and the code to MyCourses.

- Submit a zipped folder to MyCourses 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 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.
- Your group report should be submitted to Gradescope. One submission per team is sufficient.

7 Late Submission Policy

Submit your report up to 3 days late with 30% penalty.

8 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 simpler classifier, 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 dont 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.

9 Exact Deadlines

MyCourses submission closing November 26, 8:00pm EST. Kaggle submission closing November 26, 8:00pm EST.

10 Questions and clarifications

For questions, please use the following channels:

- The corresponding reddit thread.
- For more detailed questions, please go to the office hours of the following TAs: Junhao, Xin Tong, Nadeem, Prasanna