Competition I: CS4786/5786-Machine Learning for Data Science (Fall 2017)

Classifying handwritten digits.

In-Class Kaggle competition for CS4786: Machine Learning for Data Science

Fall 2017

The first in-class Kaggle competition for our class involves a classification challenge:

You are provided very few labeled examples (60 out of the 10000). So this is mostly unsupervised. You are provided with a data set created based on 10000 hand written digits (from '0' - '9').

For all the 10000 data points, you are only provided with features extracted based on the images of the handwritten digits. The underlying label of which digit each of the handwritten digit is is not provided to you. Your task in this competition is to cluster/classify (based on weak supervision) these data points into 10 clusters such that each cluster corresponds to one of the digits from '0' to '9'. Seed labels of a few data points (60 of them, 6 for each digit) are provided.

Here is what you are provided with

- The **Features describing the image of handwritten digits**: For each handwritten digit, a 1084 dimensional feature vector is extracted based on the image of the hand written digit. This is provided to you in the **Extracted_features.csv** file. The file has 10000 lines (one for each hand written digit) in the comma separated values format.
- A **similarity graph for just the first 6000 data points**: While we don't have much labels, based on a cleaner version of the first 6000 images, we have a graph with 6000 nodes representing the first 6000 data points. Two nodes in this graph are connected if they are similar. **Graph.csv** file is a csv file where each row has two entries representing an edge in the graph. For example, a row with entry "60,1035" means that node 60 in this graph is connected to node 1035 (ie. they are similar).
- **6 labeled points for eachof the 10 classes**: To help you identify your 10 clusters with the right digit from '0'-'9' we provide 6 example data-points for each digit in the data set in the file **Seed.csv.** The file consists of 60 lines. Each line has 2 numbers. The first number gives the id of datapoint (starting from 1) and the second number is the label. For example, a row with value "2017, 3" implies that the 2017'th data point is the digit '3'. Of course there are only 60 labeled examples and so not much supervision by these seed points.

Motivation: This problem is more realistic than you think! Kaggle and other competitions and basically most of ML in academic settings give a misleading view that most ML tasks are supervised with nice labeled examples. But in reality, obtaining large labeled sets is quite hard especially because often class labels are not crisp. However obtaining similarity scores between points for instance or grouping similar points can be done easily by humans. In this competition the supervision is in the form of these similarity graph while the task still remains classification.

Task: For the last 4000 handwritten digits, predict what the corresponding label from '0' to '9' is. The competition will be hosted on in-class-Kaggle.

Kaggle Link:

To be announced later...

Download the data below as zip file. When unzipped you will find the three files, **Graph.csv**, **Seed.csv** and **Features.csv**

fa2017_competition_1.zip

Group size: Group of size 1-4 students.

Due date The deadline is **11:59 pm, Thursday, 2nd November**. The due date for the report on CMS will be announced soon and is a couple days after the competition closes on Kaggle. Submit what you have at least once by an hour before that deadline, even if you haven't quite added all the finishing touches — CMS allows resubmissions up to, but not after, the deadline. If there is an emergency such that you need an extension, contact the professors.

1. *Footnote: The choice of the number "four" is intended to reflect the idea of allowing collaboration, but requiring that all group members be able to fit "all together at the whiteboard", and thus all be participating equally at all times. (Admittedly, it will be a tight squeeze around a laptop, but please try.)*

Deliverables:

1. **Early Report:** Each group should submit a one page preliminary report that includes preliminary thoughts about how you plan to attempt the competition. For each individual in the group, include what the individual plans to do for the competition. This report is due on **October 16th.** All the group members can merge their preliminary reports into one preliminary_writeup.pdf on CMS that is 1 page long. **(worth 10% of the competition grade)**
2. **Report:** In the end of the competition each group should submit a 5-15 page writeup that includes visualization, clear explanation of methods etc. See grading guidelines for details about what is expected from the writeup. **(worth 50% of the competition grade)**
3. **Predictions:** Competition is held on Kaggle in-class as a competition. You can submit your predictions to kaggle to compete with your friends. You should also submit your predictions on CMS. **(worth 40% of the competition grade)**

Collaboration and academic integrity policy

Students may discuss and exchange ideas with students not in their group, but only at the conceptual level.

We distinguish between "merely" violating the rules for a given assignment and violating academic integrity. To violate the latter is to commit fraud by claiming credit for someone else's work. For this assignment, an example of the former would be getting detailed feedback on your approach from person X who is not in your group but stating in your homework that X was the source of that particular answer. You would cross the line into fraud if you did not mention X. The worst-case outcome for the former is a grade penalty; the worst-case scenario in the latter is academic-integrity hearing procedures.

The way to avoid violating academic integrity is to always document any portions of work you submit that are due to or influenced by other sources, even if those sources weren't permitted by the rules.[2]

2. Footnote: We make an exception for sources that can be taken for granted in the instructional setting, namely, the course materials. To minimize documentation effort, we also do not expect you to credit the course staff for ideas you get from them, although it's nice to do so anyway.

Grading Guidelines: (still under construction)

You are allowed to use methods from outside of what is covered in class. But if you do, provide a clear comparison with ``reasonable methods'' covered in class.

1. Visualization (10%)
    a. Inclusion of plots/diagrams (5%)
    b. Explanation of how visuals helped develop the model (5%)
2. Algorithms (30%)
    a. Correct use of algorithms (15%)
        i. Used principled approach to extract information from similarity graph and provided clear explanation and reasoning (5%)
        ii. Extracted and used common information from both the features and the similarity graph in a principled fashion and provided clear explanation and reasoning(5%)
        iii. Used clustering algorithms to cluster datapoints into classes, clearly explained and analyzed the method (5%)
    b. Explanation of how algorithms helped to develop model (15%)
        i. Showed evident understanding of each algorithm used
3. Model (40%)
    a. Use of data (30%)
        i. Individual testing (10%)
            1. Tested performance on just features, just graph
        ii. Combining data (10%)
            1. Combined data from features and graph to develop model
        iii. Partial supervision (10%)
            1. Used seeds to classify points into classes
    b. Parameters (10%)
        i. Evident testing of different parameters (5%)
        ii. Reasons for choosing certain parameters (5%)
4. Failed Attempts (20%)
    a. Explanation (10%)
        i. Explained how they developed their failed models and why they think those models failed
    b. Improvement (10%)
        i. Explained how failed attempts led them to develop their final model

Bonus (at the discretion of the graders):

- Tried new or more methods not necessarily covered in class
- Developed new algorithm or methods, tweaked existing methods to fit the problem better

- ...
-