COL 774: Assignment 2

Due Date: 11:50 pm, March 12 (Tuesday), 2019. Total Points: 33+

Notes:

- This assignment has two parts - Text Classification using Naïve Bayes and Handwritten digit classification using SVM.
- You should submit all your code (including any pre-processing scripts written by you) and any graphs that you might plot.
- Do not submit the datasets. Do not submit any code that we have provided to you for processing.
- Include a **single write-up (pdf) file** which includes a brief description for each question explaining what you did. Include any observations and/or plots required by the question in this single write-up file.
- You should use Python/MATLAB for all your programming solutions.
- Your code should have appropriate documentation for readability.
- You will be graded based on what you have submitted as well as your ability to explain your code.
- Refer to the course website for assignment submission instructions.
- This assignment is supposed to be done individually. You should carry out all the implementation by yourself.
- We plan to run Moss on the submissions. We will also include submissions from previous years since some of the questions may be repeated. Any cheating will result in a zero on the assignment, a penalty of -10 points and possibly much stricter penalties (including a **fail grade** and/or a **DISCO**).

1. **(33 points) Text Classification**
   In this problem, we will use the Naïve Bayes algorithm for text classification. The dataset for this problem is a subset of the Yelp dataset and has been obtained from this website. Given a users review, task is to predict the stars given by the reviewer. Read the website for more details about the dataset. You have been provided with separate training and test files containing 534K reviews (samples) and 133K reviews respectively. This data can be downloaded from the course website. A review comes from one of the five categories (class label). Here, class label represents stars given by the user along with the review. Please refer to *README* in data directory for more details.

   (a) **(10 points)** Implement the Naïve Bayes algorithm to classify each of the articles into one of the given categories. Report the accuracy over the training as well as the test set.

   Notes:

   - Make sure to use the Laplace smoothing for Naïve Bayes (as discussed in class) to avoid any zero probabilities. Use $c = 1$.
   - You should implement your algorithm using logarithms to avoid underflow issues.
   - You should implement Naïve Bayes from the first principles and not use any existing Matlab/Python modules.

   In the remaining parts below, we will only worry about test accuracy.

(b) **(2 points)** What is the test set accuracy that you would obtain by randomly guessing one of the categories as the target class for each of the review (random prediction). What accuracy would you obtain if you simply predicted the class which occurs most of the times in the training data (majority prediction)? How much improvement does your algorithm give over the random/majority baseline?

(c) **(3 points)** Read about the confusion matrix. Draw the confusion matrix for your results in the part (a) above (for the test data only). Which category has the highest value of the diagonal entry? What does that mean? What other observations can you draw from the confusion matrix? Include the confusion matrix in your submission and explain your observations.

(d) **(4 points)** The dataset provided to is in the raw format i.e., it has all the words appearing in the original set of articles. This includes words such as 'of', 'the', 'and' etc. (called stopwords). Presumably, these words may not be relevant for classification. In fact, their presence can sometimes hurt the performance of the classifier by introducing noise in the data. Similarly, the raw data treats different forms of the same word separately, e.g., 'eating' and 'eat' would be treated as separate words. Merging such variations into a single word is called stemming.

- Read about stopword removal and stemming (for text classification) online.
- Use the script provided with the data to you to perform stemming and remove the stop-words in the training as well as the test data. You are free to use other tools as well.
- Learn a new model on the transformed data. Again, report the accuracy.
- How does your accuracy change over test set? Comment on your observations.

(e) **(5 points)** Feature engineering is an essential component of Machine Learning. It refers to the process of manipulating existing features/constructing new features in order to help improve the overall accuracy on the prediction task. For example, instead of using each word as a feature, you may treat bi-grams (two consecutive words) as a feature. Come up with at least two alternative features and learn a new model based on those features. Add them on top of your model obtained in part (d) above. Compare with the test set accuracy that you obtained in parts (a) and parts (d). Which features help you improve the overall accuracy? Comment on your observations.

(f) **(3 points)** Read about another performance metric referred to as F1-score. For your best performing model obtained above, report the F1-score for each class in the test set. Also report the average of these numbers referred to as macro F1-score. Which metric, test error or macro-F1 score, do you think is more suited for this kind of dataset? Why?

(g) **(6 points)** You are also provided with full publicly available version of the Yelp dataset containing 5M training instances and 1M test instances. Train your model on this version of the dataset. Use the best performing model obtained in part (e) above. Note that this may take a while to train. Report your test set accuracy as well as Macro F1 score on the original test set (used in parts (a)- (f) above). What do you observe? You are free to run your model on the full test set (1M instances) though you do not have to report these numbers.

(h) **Extra Fun: No points** Till now we have been only using user reviews to predict stars, however the dataset also contains other fields such as number of votes the review received. Can the prediction accuracy be improved if you make use of other available fields? Experiment and report your findings.

Note: Full version of the data is only to be used for part (g), you should use subset of the data for parts (a)-(f).

2. Second question will be on SVMs will be posted later.