TIPR Assignment - I

TA: Anirban

January 24, 2019

Instructions

- This is a coding assignment. The code has to be written in Python. You can use Python version 2 or 3 to solve all the problems. Please mention the version you have used in the report.
- The assignment deliverable is your python code, results, plots and your report of observations.
- Send the link of your github repo to **tipr-e1313@outlook.com** with the following format for the subject name of email: `TIPR_1_<last 5 digits of serial number>`.
- You are encouraged to discuss among yourselves, but **DO NOT COPY** solutions or code. The consequences will be severe.
- Read the complete assignment carefully, before attempting to solve it.
- Follow the instructions provided for how your code is to be run, the formats for input and output files and the naming conventions. — Detailed Instructions
- The submission deadline is **February 7, 2019**.

Best of luck for the assignment. HAPPY CODING!

Amar, Akbar and Anthony : Dimensionality!

Amar and Akbar are two very close friends of each other. Apart from many other similarities between them, there is one important commonality - both of them are Machine Learning enthusiast. One fine day, while having lunch at the collge canteen, they were discussing about the dimensionality of data and they ended up having a huge and heated debate on that topic! Amar thinks that high-dimensional data is more useful in classification, whereas Akbar's opinion is low-dimensional data makes more sense.

To come up with a conclusion and putting an end to the debate, they decided to meet their professor Anthony to seek for suggestions. Anthony, being an expert in this domain, upon hearing their query, did not give any straight-away answers. Rather, he gave two different assignments to Amar and Akbar! He gave them several labeled datasets with high-dimensional data and asked them to check the classification results on both the high as well as low dimension.

He asked Akbar, who is a fan of lower dimensions, to classify the data on low-dimension. For that he asked him to come up with some nice strategy to convert the original data into lower dimensions and then check the classification accuracy using Bayes and NN classifier. On the other hand, he asked higher-dimension-fan Amar, to check the same, but with the original high-dimensional data.

In this assignment, you have to play the role of both Amar and Akabr! You have to do the following tasks sequentially to reach a conclusion.

Task - I (10 Points)

Implement Random Projections algorithm to convert the high-dimensional data into lower dimension. If the original data dimension is K, your algorithm should be able to lower it down to $D(D < K)$. Experiment with $D = 2, 4, ..., \lceil K/2 \rceil$ for all the datasets given and save them into files of suitable format for later use. **DO NOT** use any python library. Implement on your own.

Task - II (15 Points)

Design *Bayes classifier* and *Nearest Neighbour* classifier. For Bayes classifier, you need to estimate class conditional densities (follow a suitable scheme e.g. maximum likelihood) using training data. You'll be using both of these for the classification problem on the datasets provided herewith. **You are not allowed to use any package**. Design from scratch.

Task - III (10 Points)

Divide the data (both the original/high and low-dimensional) into train and test set using cross-validation technique. Measure **accuracy** and **F1-score**(Macro and Micro) for all the data sets. Plot the results for different values of D and K. What is your take-away? Which side are you in - Amar's or Akbar's?

Task - IV (5 Points)

Do the same as in task-III, but this time use *scikit-learn* library for both Bayes and NN classifier.

Task - V (5 Points)

Compare the results obtained in Task-III and Task-IV. Plot the comparisons accordingly. Is your classifier able to outperform scikit-learn's? If not, what might be the possible reasons you can think of?

Task - VI (10 Points + 15 Bonus)

Implement **Locality Sensitive Hashing**(LSH) to reduce the dimensionality of the data. Preferably, do it on your own. `Bonus credits for those who will be doing it independently, without using any library`.

Task- VII (10 Points)

Perform the classification task(similar to what you did in Task-III) using the LSH algorithm you developed in the previous question. Compare your results with PCA. You are allowed to use the standard python library for PCA.

Datasets Description

1. Dolphins

It is an *undirected social network* of frequent associations between 62 dolphins in a community of Doubtful Sound, New Zealand. Each dolphin is represented by 32-dimensional vector and there are two communities, leading two different labels. You are provided with two files - **dolphins.csv** which contains the vectors associated with the dolphins and **dolphins_label.csv** containing the corresponding labels. The indices in both the files are consistent.

2. Twitter

It consists around 20000 labelled tweets for the *Message Polarity Classification* problem. This dataset contains only english tweets. There are three sentiments - positive, negative and neutral. Again two files are given - **twitter.txt** and **twitter_labels.txt**.

Hint - Represent each tweet as a feature vector (e.g., using bag of words representation)

3. PubMed

It consists of scientific publications from PubMed database related to diabetes classified into one of three classes: Experimental, Type1 or Type2. This network has a total of 19717 nodes and 3 labels. Each node is represented by 128 dimensional vector. Again, there are two files - the data file is name **pubmed.csv** and the label file is named as **pubmed_.label.csv**.

How to run

Far each of the three datasets mentioned above, there will be hidden test files. The performance of your code will be tested on these test data. I should be able to run the code in the following format:

```
python main.py --test-file <test_input_file_path>
```

Assignment Deliverable

- Prepare a report and name it `TIPR_Report_1.pdf`. In your report, you need to briefly describe what you have done, present the results (in a form you think is good) and provide a brief discussion of the results you obtained. It should contain the answers of the tasks, plots and output data for the given datasets. Please make it concise and to-the-point.
- The name of the plots asked for should follow the following naming convention: `task_j.png`. For example, the filename for task-I should be `_task_1.png`. Put these plots into the `output_plots/` directory.
- The name of output file for various tasks needs to be `task_i.txt` or `task_i.pdf`. Put them into the `output_data/` directory.

Sample Input and Output

Here is how the sample output would look like in the terminal when I would run the code with test data for, e.g., *twitter* dataset.

```
Test accuracy ::  78.32
Test Macro F1-score ::  47.35
Test Micro F1-score ::  53.23
```