Machine Learning

Programming Assignment 2

Tentative Due Date: April 11, 2020

Assignment Guidelines

1. Any kind of plagiarism is not accepted. We will strictly follow institute policies for plagiarism.
2. Recommended programming languages: Python + sklearn
3. *Submission should include:* Working code for each question separately and a report to show the analysis of results and corresponding plots in each of the parts. Make sure to append all the relevant details and plots in your report to support your results.
4. *Submission Guideline:* upload a single .zip file with name 'YourRoll_PA1.zip' and each solution should be a .py file with name 'YourRoll_Q1.py' (consider for first question) and a report file 'YourRoll_report.pdf'. No other file should be added in the folder.
5. Your code should run as $python3 <YourRoll_Q2.py> <path to the dataset containing the train and test set downloaded from the link.>

Total: 160(+20) marks

1. **Naive Bayes Classification**

   Load the wine dataset using sklearn library. Shuffle the data with seed value 42 and perform a 70-30 stratified split of the data into a train and test set. Also, plot the class-wise distribution of data in the train and test set (one histogram for train set, one for test set). Compare the distributions. Now, perform classification as follows: **10 + (2*5) = 20 marks**

   (a) Train a Gaussian Naive Bayes classifier and report (a) the class priors, (b) mean and variance of each feature per class.

   (b) Train another Gaussian Naive Bayes classifier by setting prior probability for the classes. Repeat this experiment by setting priors in the ratios: (a) 40-40-20, (b) 33-33-34, and, (c) 80-10-10.

   For all the experiments above, report the (a) accuracy, (b) confusion matrix, on the train and test set. State your observations and analysis. The report should include the histograms, as well.

   10 + (10*3) = 40 marks

   (Bonus) Implement a simple Naive Bayes classifier from scratch for the above dataset. Compare your classifier with the results obtained above. Report your experiments thoroughly and document the code well. **20 marks**

2. **Bagging and Boosting**

   Download the **given dataset**. The "target"field refers to the presence of heart disease in the patient (0 signifies no presence of heart disease, while 1 signifies presence). Consider 80/20 split for train/test split. Implement Adaboost classifier to predict whether a patient suffers from heart disease or not, with n single node (1 - level) decision trees.

   Tasks:

   (a) Plot the test error of classifier vs number of trees used in the classifier (varying n from range [20,40,60,80,...240] at an interval of 20).

   (b) Plot the train error of classifier vs no. of trees used in the classifier (varying n from range [20,40,60,80,...240] at an interval of 20).

   (c) Create ensemble of classifiers with varying n in (50,100,150,200).

   (d) Report accuracy, f1_score and confusion matrix of classifier obtained in (c) on test data.

   (e) Compare the ensemble classifier obtained in (c) with single decision tree classifier trained on the training data (split criterion = entropy), report which is better and why.

5+5+15+5+20 = 50 marks

3. **KNN**

   Load the breast cancer dataset from the sklearn library (Instead you can choose to manually download the dataset). Consider a 70/20/10 split for train/validation/test data, respectively. Train a KNN classifier on the train data with the following variations:

   (a) Three different values of k (no. of neighbors).

   (b) Two different distance metrics (Minskowski and Euclidean).

   (c) All points in each neighborhood weighted equally.

   (d) Points in the neighborhood weighted by their distance such that the closer neighbors of a query point have a greater influence than neighbors which are farther away.

   Report classifier accuracy on the train and validation set. Identify the top-two classifiers among the above based on the classifier's performance on the validation set. Analyze why they perform better than the other classifiers. For these two classifiers:

   (a) Note and report the different parameter values (values of k, distance metric and weighting scheme used).

   (b) Report the accuracy on the test set. **25 + 5 = 30 marks**

   Using the parameter values obtained from above experiments, train two new classifiers as follows:

   (a) Use both training and validation split for training.

   (b) Use only the first two features of the dataset for training. **10 marks**

   Report accuracy of the above classifiers on the train and test set. Plot decision boundaries obtained and compare them. Report your observations and analysis. **10 marks**