Lab Assignment - 4
CS 416: Statistical Pattern Recognition Lab
Course Instructor : Prof. Prabuchandran
Teaching Assistant : Gowramma B. H.
Date: 21 Mar 2021

INSTRUCTIONS: This is a group assignment. You have to give clear and detailed plots and solution to each of the questions. **Send one single pdf file containing solutions to all problems in the google form link before 1st April, 8.30 am before the SPR class. Only one member of the team has to submit the assignment. Use the link** `https://forms.gle/rwPeNWnM6tao8yEm6` **to submit your assignment. Name your pdf with** *rollno*1_*rollno*2_*rollno*3**.** For example 190010005_190010006_190010007.*pdf*. Late submissions will not be graded. Students can discuss but must write their solutions based on their understanding independently. Do not use web resources or answers from your peers to obtain solutions. If anyone is involved in malpractice of any sort, then suitable disciplinary action will be taken.

In this assignment you are supposed to evaluate and compare different linear classification methods that you have learnt on three datasets. The different linear classifiers are perceptron (POCKET algorithm, modified version of the Perceptron algorithm for non separable data), Linear Least Squares for classification, Logistic Regression and Fisher Linear Discriminant.

Code all the classifiers yourself and compare their performances with standard implementation available on scikit-learn. On the completion of tasks, each one of you is to make a detailed report on your observations. In the report, clearly mention what objective/loss function each classifier is trying to minimize.

The first dataset is synthetic and the other two are real datasets.

1. Consider a 2-class problem. The class conditional densities are 10-dimensional Gaussian with mean vector for class 1 all identically one and the mean vector for class 0 all identically zero. Assume both densities have identity covariance matrix. Assume prior probabilities for classes to be 0.5. Generate 2000 training data 1000 test data points points by sampling from this prior and class conditional densities. First sample the class according to the prior probability and then sample the 10-d feature vector conditioned on the sampled class.

    - Vary the prior probabilities and compare the performance of different algorithms. You could increase the size of the training dataset while you vary the prior probabilities.
    - Repeat the experiment for 2-dimensional data. Plot and compare the decision boundaries of all the algorithms.
    - In the 2-dimensional data setting, consider using the same mean for both classes but different covariance matrices. For class 1 consider setting (1, 0.9;0.9, 1). How all the algorithms perform on this dataset? If they do not perform well then make suitable transformations to your dataset and then compare algorithms.

- Repeat the experiment with setting mean for class 1 = [3 6], covariance for class 1 (1/2 0;0 2) and mean for class 2 = [3 -2], covariance for class 2 (2 0;0 2). Take more training points and compare the decision boundaries.
- In all examples compare your classifier boundary with Bayes Classifier decision boundary.

2. Check this link: `https://archive.ics.uci.edu/ml/datasets/statlog+(german+credit+data)`

   Use the file with name german.data-numeric

   The features represent various financial attributes of a person and the class labels denote whether or not a person is 'good' for extending credit or not.

   Use different train and test split on this dataset to compare the performance of classifiers. Consider using 70:30 and 80:20 for the train and test split. Report train accuracy, test accuracy, confusion matrix and F1 score. Check for class imbalance and modify the loss if required based on the type of classifier.

3. Check this link: `https://www.kaggle.com/c/porto-seguro-safe-driver-prediction/data?select=train.csv#`

   Use the file with name train.csv

   Use different train and test split on this dataset to compare the performance of classifiers. Consider using 70:30 and 80:20 for the train and test split. Report train accuracy, test accuracy, confusion matrix and F1 score. Check for class imbalance and modify the loss if required based on the type of classifier. Compare the results with at least two different non-linear classifiers of your choice(you can use library functions directly for non-linear classifiers)

Note: In all experiments, write your observations. Add GitHub link to the code for each question.