ML Assignment-1 Report

## Group Details

|  |  |
| --- | --- |
| **Tangeda Sai Sharan** | **2017A7PS0241H** |
| **Sanjiv Yelthimar Shenoy** | **2017A7PS0224H** |
| **Manish Kumar Bachhu** | **2017A7PS0036H** |

## 1. Fisher’s Linear Discriminant Analysis

We performed Fisher Discriminant Analysis on the 2 datasets provided. The results are as shown below.

***To run the code:***

* In order to not repeat code, the program first asks us to enter the relative path of the data file. Based on what path is entered, the algorithm is run on the particular dataset.

Relative paths for the datasets based on the directory structure we submitted will be:

For file 1: ./a1\_data/a1\_d1.csv

For file 2: ./a1\_data/a1\_d2.csv

### Design Decisions/ Optimization Techniques used

* Numpy library was used everywhere for performing arithmetic operations using matrix multiplications, due to this optimization the code complexity is decreased, hence leading to faster output.
* No loop was used in any part of the application except in calculation of the Confusion matrix

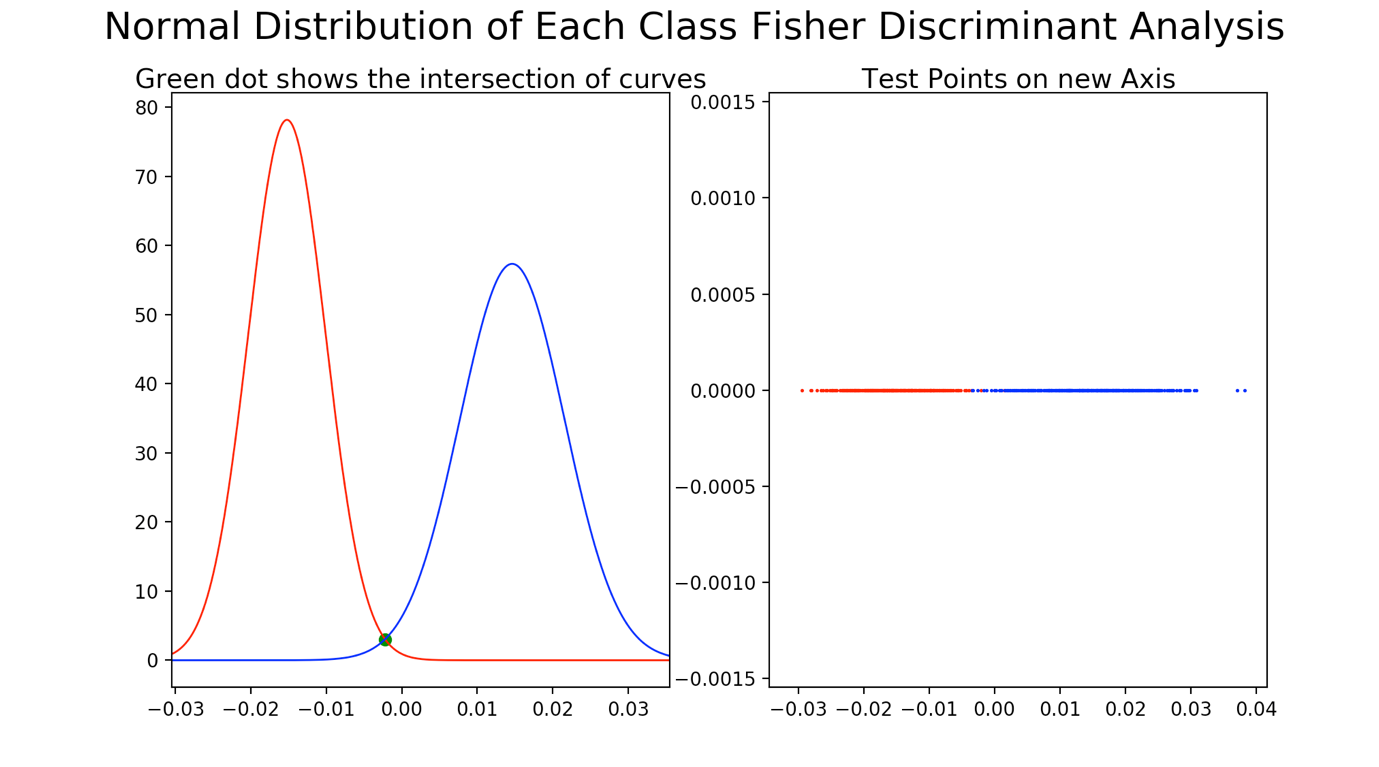
### 1.1. Data Set 1

Precision: 1.0

Recall: 0.9896

Accuracy: 0.995

F - Score: 0.9948



2 graphs are shown here.

The left graph shows the intersection of the normal distributions of the positive and the negative points on the training data set. The threshold point for separation is taken as the intersection of the normal distributions.

The right graph shows the visualisation of the testing points projected on a line got by the fisher discriminant analysis. As you can see, the positive and negative points are nearly separated out into 2 regions with just a little overlap.

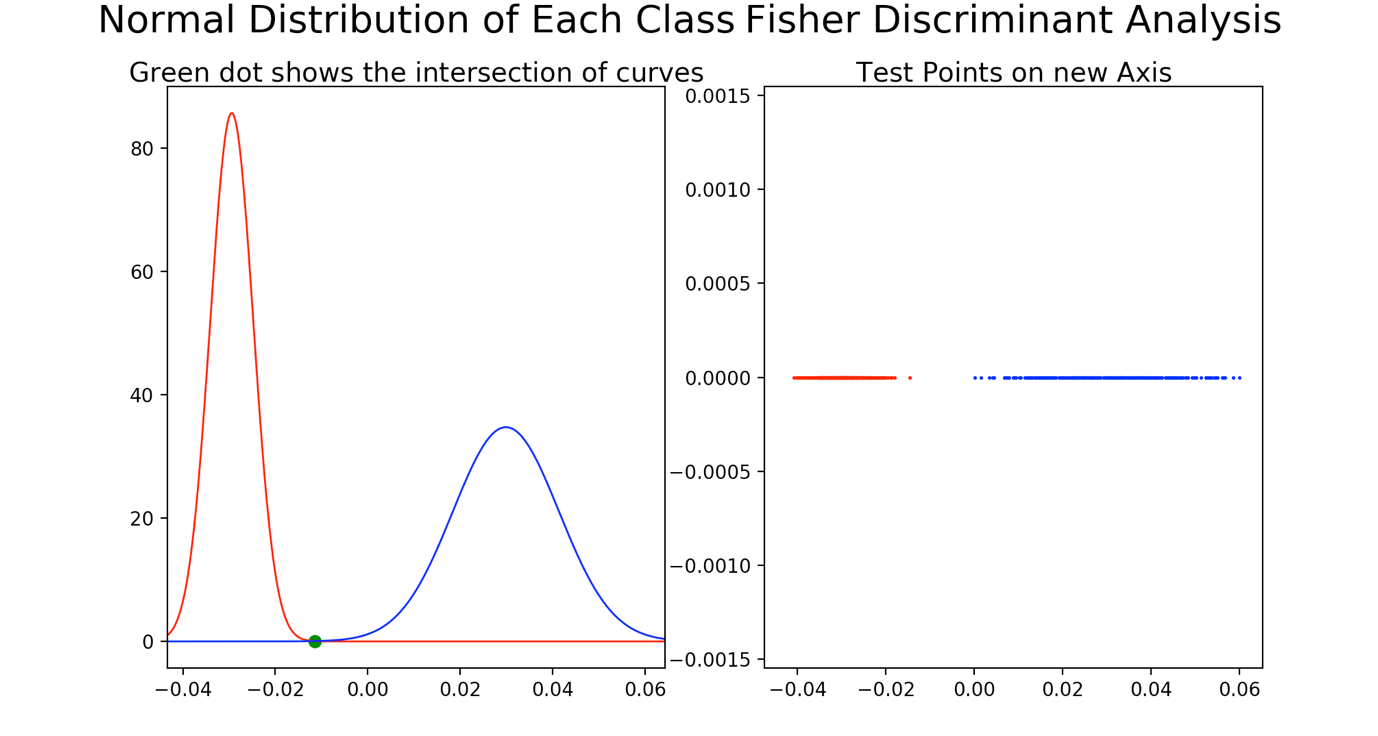
### 1.2. Data Set 2

Precision: 1.0

Recall: 1.0

Accuracy: 1.0

F - Score: 1.0



2 graphs are shown here.

The left graph shows the intersection of the normal distributions of the positive and the negative points on the training data set. The threshold point for separation is taken as the intersection of the normal distributions.

The right graph shows the visualisation of the testing points projected on a line got by the fisher discriminant analysis. As you can see, the positive and negative points are clearly separated out in this dataset thereby giving us 100% accuracy.

Conclusion:

For both the data sets, our model performed quite well almost entirely separating out the points accurately.

## 2. Naïve Bayes

We are submitting 2 files for this question. Both work individually. One uses stemming for an external library and the other one doesn’t.

Process:

1. We first did some basic pre-processing such as removing punctuations and making everything into lower case.
2. This step is different for the 2 files we submit for this question. We stemmed the words in one of the files using Porter’s stemming algorithm. In the other one we didn’t.
3. Then we did laplace scaling to even things out in the probabilities and reduce the effect false negatives in our results. False negatives were a menace as if even if 1 word in the test sentence belonged to a negative class, then automatically the entire sentence would be termed negative.(since we are multiplying probabilities in the numerator, even if one is 0, the entire fraction becomes 0).
4. After this we applied the naïve bayes algorithm to get the result.

Accuracy and F-Score Values without Stemming

Average Accuracy: 0.808 +- 0.0289137

Average F Score: 0.81267 +- 0.03214

Accuracy is individual folds of the 5-fold cross validation:

0.815, 0.855, 0.765, 0.800, 0.805

F Score is individual folds of the 5-fold cross validation:

0.828, 0.869, 0.781, 0.794, 0.791

Accuracy and F-Score Values with Stemming

Average Accuracy: 0.825 +- 0.0255

Average F Score: 0.826 +- 0.033

Accuracy is individual folds of the 5-fold cross validation:

0.845, 0.865, 0.80, 0.810, 0.805

F Score is individual folds of the 5-fold cross validation:

0.853, 0.876, 0.811, 0.802, 0.789

## Conclusion:

Stemming increased the accuracy by 1.7% and F-Score by 1.4%. As they say every little bit counts.

We got an accuracy above 80%. A random method of selecting 0 or 1 would yield 50% probability. Our model performs significantly better.