CEG 7570

Pattern Recognition

Project Report

By

Name: Aman Ali Pogaku

UID: U00878439

Email: pogaku.6@wright.edu

INTRODUCTION

Bayesian Classifier is one of the "Probabilistic Models". Conceived with the application of Bayes' theorem on datasets with features that are generally considered independent of each other. It is used in the classification of data into different classes.

The implementation of the algorithm is easy and produces efficient results on datasets with smaller entities. There are a few major applications of this algorithm. They are:

1) Categorizing news: Initially, based on the contents of the news, the data is classified into different categories. Now the system knows how to classify the other set of news as and when it comes. It is used a lot by the standard news channels.

2) Spam filter design: This is one of the most important applications of this algorithm. It can be used to categorize the new incoming emails into spam and not spam. It is implemented by companies like Google, Microsoft etc for their mailing services.

3) Face recognition: Face recognition software incorporates the use of Bayes' classifiers to identify and recognize distinctive features of a face. This has a lot of applications in the security domain.

4) Sentiment analysis: Sentiment analysis uses Bayes' classifiers for classifying the different sentiments. And then a probabilistic model created. Now based on these probabilities, predictive models are developed for research purposes.

In this project, we will be developing a Bayes' classifier and we will be working on two datasets.

Description of dataset 1:

1) This data was taken from images of Bank notes.

2) It has two classes.

3) It has 4 features and 1372 instances.

4) Further details can be found on this link in the references [1].

Description of dataset 2:

1) This data describes the geometrical properties of three varieties of seeds.

2) There are three classes in this dataset.

3) It has 7 features and 210 instances.

4) Further details can be found on this link in the references [2].

IMPLEMENTATION:

To implement this we need to select a tool that can easily perform statistical calculations. There are many programming languages and tools that can be used to implement this project. For example: Java, Python, MATLAB, C++ etc. Of all the options that are available, I feel Matlab is the best fit for this project. This is because MATLAB is a complete toolbox which can be used for this project without the requirement to install extra dependencies like installing Numpy in the case of Python.

Another added advantage of MATLAB is that we can easily define matrices and perform algebraic operation on it with the help of various inbuilt commands. If we had to perform the same thing in another language, then we might have to think of a computational logic and then translate that into a few lines of code that might involve use of iterative and conditional statements. Thus, MATLAB saves a few lines of code and saves time for the developer to focus on the main critical aspects of the program than getting distracted by the verbose code that needs to be realized.

THE PROGRAMMING STYLE:

The whole program is divided into the main program and set of various functions and those functions are called as and when it is required. The advantage of this method is that it would be easy for debugging and maintaining the code.

PROJECT PART 1

DESCRIPTION:

In this part, we will find the best feature of the given datasets using Fisher's discriminant ratio. Fisher's discriminant ratio is a class separability measure, which is used to find the feature that best distinguishes one class from another. We will find Fisher's discriminant ratio for each feature. The higher the value of Fisher's discriminant ratio, better the feature for recognizing further new data that is added to the classifier also known as the test dataset.

ALGORITHM:

1) Read the file into the main program.

2) Divide the data into classes and then find the class with least data entities.

3) Let the least number be X. Move first X/2 points of each class into training dataset and move the next X/2 points to the test dataset.

4) Combine the training and test datasets for normalization. Now, one can use the inbuilt functions of MATLAB. But since it's not clear as to how MATLAB implements the functions. It is advisable to calculate the mean and the variance on our own using the formulas given in the formulas section.

5) Now, divide the normalized data into test and training datasets by following the approach given in step 3.

6) Find mean and variance of all the classes in training data using the formulas in the formula section.

7) Find the Fisher's discriminant ratio for each of the feature.

8) The feature with the maximum Fisher's discriminant ratio is the best feature of that dataset.

COMMENTS ON DATASET 1:

1) Uneven class distribution.

2) It has two classes.

COMMENTS ON DATASET 2:

1) Equal class distribution.

2)It has 3 classes. While calculating Fisher's discriminant ratio we need to find the Fisher's discriminant ratio for 3 pairs of classes and then take a sum of the three pairs.

FORMULA:

1) Mean : $\mu = (\Sigma Xi) / N$
2) Standard Deviation: $\sigma = sqrt[\Sigma (Xi – \mu)2 / N]$
3) Variance: $\sigma2 = \Sigma (Xi – \mu)2 / N$
4) Fisher's discriminant ratio : $FDR=(\mu1- \mu2)^2/ (\sigma1^2 + \sigma2^2)$

RESULTS:

Bank note dataset:

```
fisher dicrimant ratio for all four features is
    2.2003    0.5061    0.0514    0.0002

best feature based on fisher dicriminant ratio is feature number = 1
```

Output screenshot

The best feature for this dataset is feature “1” with a Fisher's discriminant ratio of 2.2003.

Seeds dataset:

```
fisher dicrimant ratio for all seven features is
   33.0481   33.8742    6.1695   17.7861   26.1194    2.6375   21.1666

best feature based on fischer discriminant ratio is feature number = 2
```

Output screenshot

The best feature for this dataset is feature “2” with a Fisher's discriminant ratio of 33.8742.

PROJECT PART 2

DESCRIPTION:

In the second part of the project, we will take the best feature that we had got in the first part of the project, and then create a Bayesian classifier.

ALGORITHM:

1) Find A-priori probabilities of all the classes in training dataset.

2) Calculate the class conditional probabilities of all classes training dataset.

3) Calculate posterior probabilities and plot its histogram.

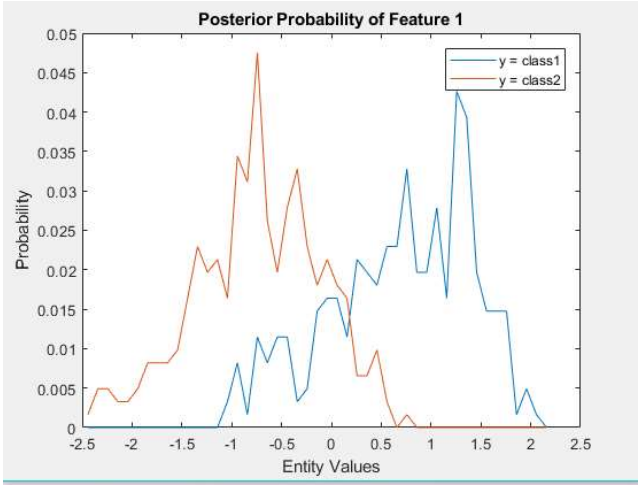
4) Once we get the histogram, we create a matrix. The total number of columns is equal to the total number of classes and will contain the posterior probability values of all the classes. Then we create a decision matrix with a single column. This column will have the class of a particular bin in the histogram which has the maximum value.

5) Now we will send the training data onto the program and then we will classify the values with the help of decision matrix. Now we have predicted values of classes.

6) By comparing the predicted values and the actual values of the test data set, we calculate the recognition rate.

RESULTS:

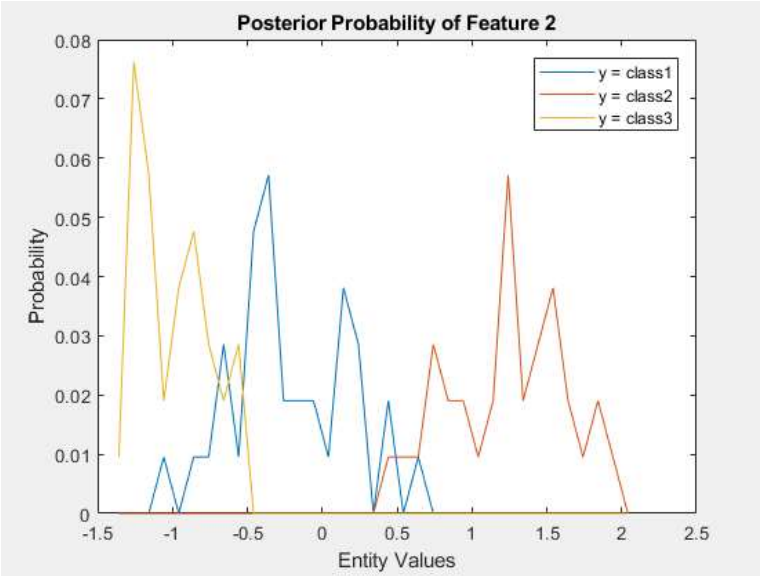
Bank data set:



Histogram

In the previous part we got Feature “1” as the best feature. Its recognition rate calculated by the recognizer is 84.3%.

Seeds dataset:



Histogram

In the previous part we got Feature “2” as the best feature. Its recognition rate calculated by the recognizer is 79.05%.

PROJECT PART 3

DESCRIPTION:

This part has two versions. In both the versions we will re-divide the datasets based on the total number of entities it has, unlike we did it in first part.

ALGORITHM FOR VERSION 1:

1) Use the recognizer created for part 2 and calculate the recognition rates for new training datasets.

2) Select the best feature (one having the maximum recognition rate) from this and create a recognizer from this feature.

3) Now evaluate the test dataset using the recognizer using the previous step.

4) Have the final recognition rate calculated.

ALGORITHM FOR VERSION 2:

1) Create recognizers for each feature of the dataset.

2) Now send test data set into each recognizer.

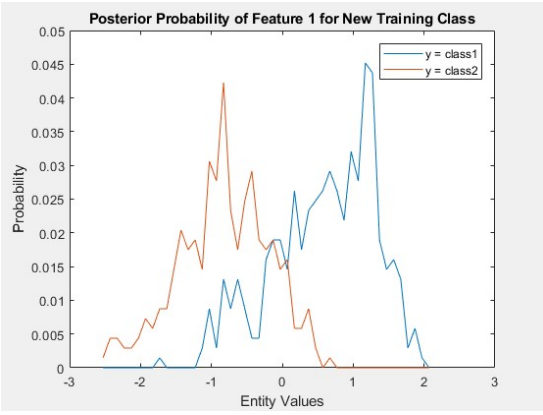
3) Best feature is the feature having the maximum recognition rate.

COMMENTS ON DATASET 1:

1) When the data is re-divided, we will have a change in the training and test data entities.

RESULTS:

Bank note dataset (version 1):

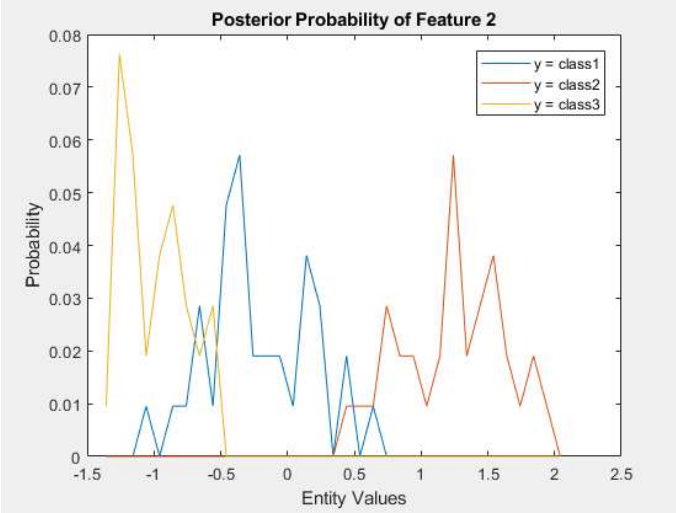


Histogram

Final recognition rate for test data using best feature recognizer = 83.527%

The best feature = feature “1”

Seed dataset (version 1):

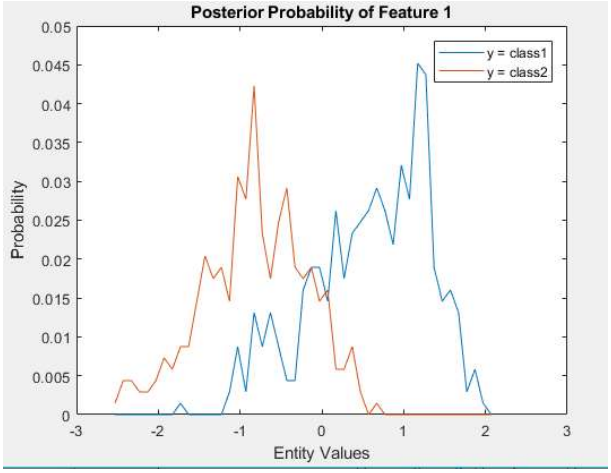


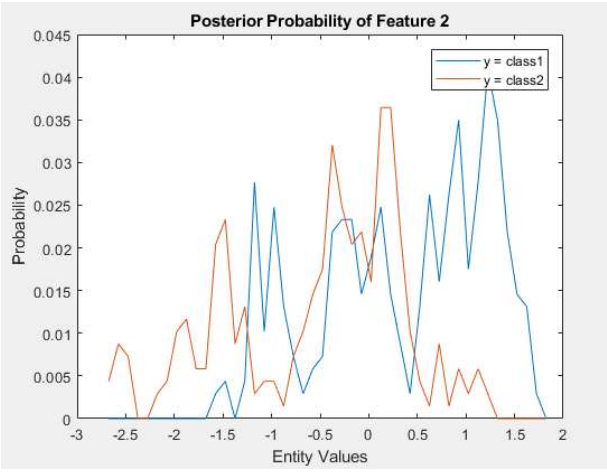
Histogram

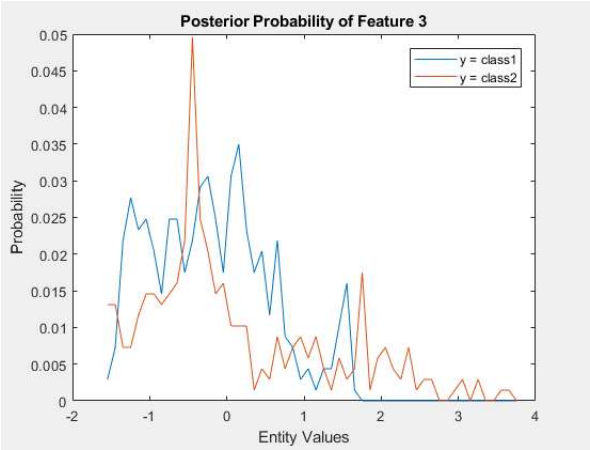
Final recognition rate for test data using best feature recognizer = 79.05%

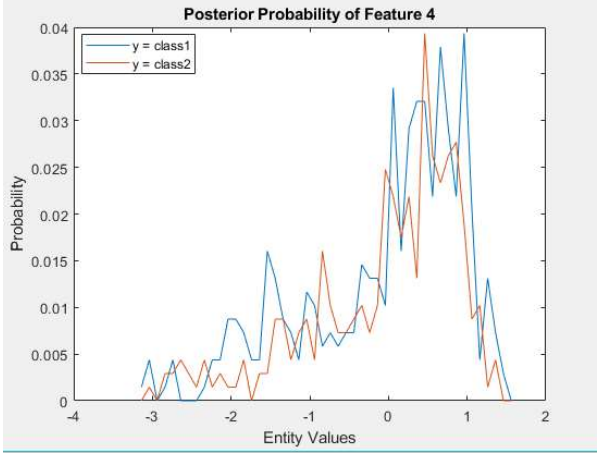
The best feature = feature “2”

Bank note dataset (version 2):









Histograms

Recognition rate for feature 1 = 0.8353

Recognition rate for feature 2 = 0.7638

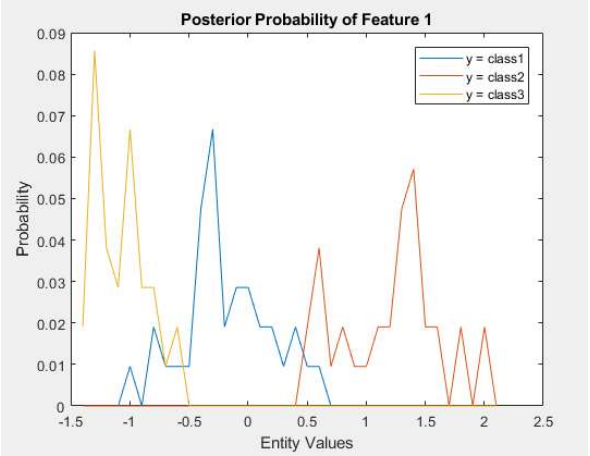
Recognition rate for feature 3 = 0.6837

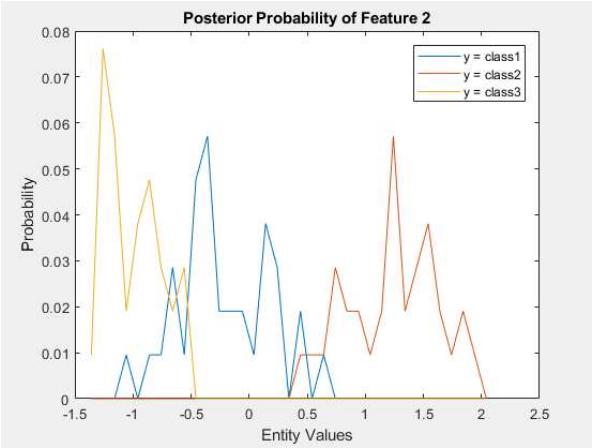
Recognition rate for feature 4 = 0.5481

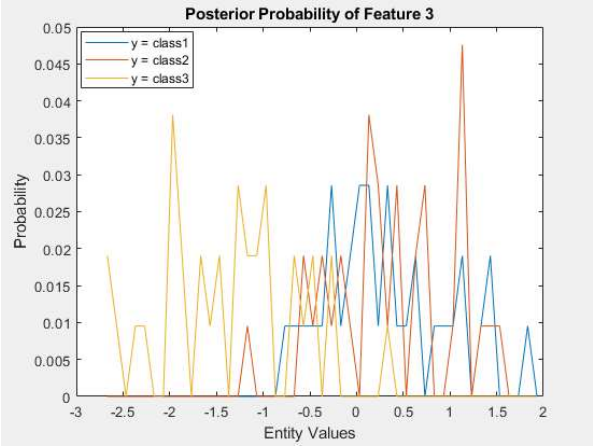
Best feature using recognition rate is feature number = 1

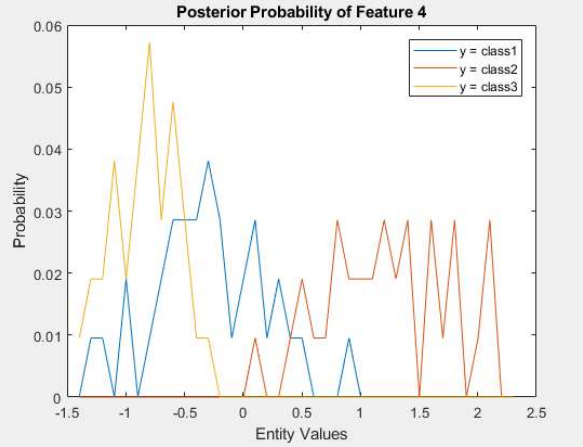
Recognition rate for the best feature = 83.53%

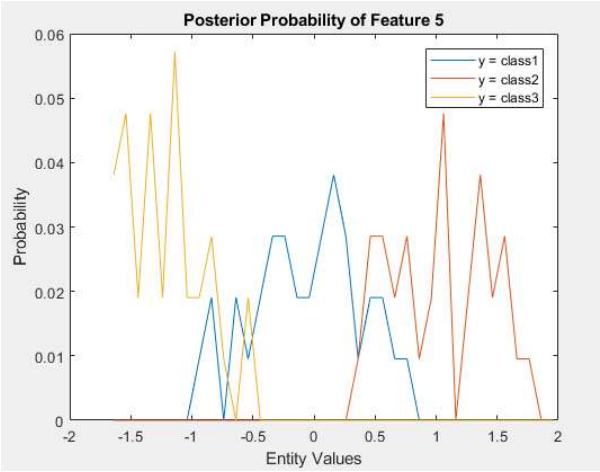
Seed dataset (version 2):

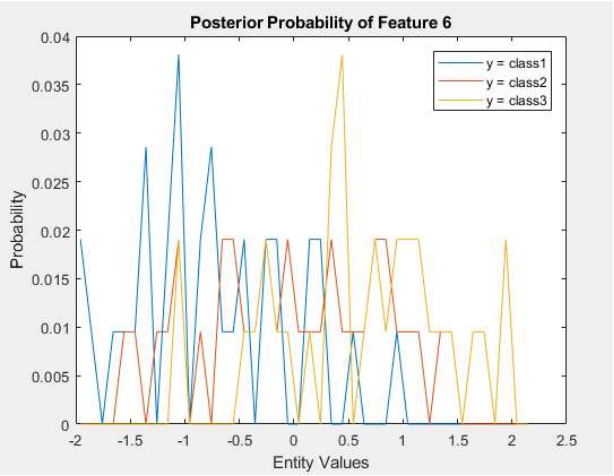


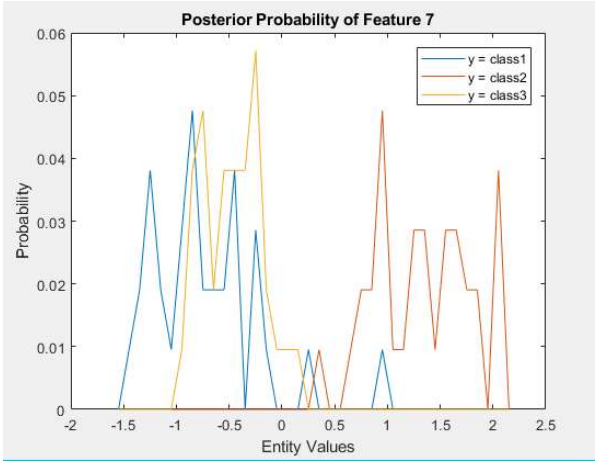



Posterior Probability of Feature 3
y = class1
y = class2
y = class3
Probability
Entity Values


Posterior Probability of Feature 4
y = class1
y = class2
y = class3
Probability
Entity Values


Posterior Probability of Feature 5
y = class1
y = class2
y = class3
Probability
Entity Values


Posterior Probability of Feature 6
y = class1
y = class2
y = class3
Probability
Entity Values


Posterior Probability of Feature 7
y = class1
y = class2
y = class3
Probability
Entity Values

Histograms

Recognition rate for feature 1 = 0.7905

Recognition rate for feature 2 = 0.7905

Recognition rate for feature 3 = 0.4571

Recognition rate for feature 4 = 0.6857

Recognition rate for feature 5 = 0.7619

Recognition rate for feature 6 = 0.4667

Recognition rate for feature 7 = 0.5524

Best feature based on recognition rate is feature number = 1

Recognition rate for the best feature = 79.05%

OBSERVATIONS:

1) Here for the **Seeds** dataset, we are getting the best feature as "1". This is different from the feature number we have got before, that is "2". This is because, both feature "1" and "2" have the same recognition rates and when MATLAB processes the matrix of the recognition rates, it gives us the first index encountered in a situation of two or more indices having the same value. Thus, there is a change in the value of the best feature.

2) There is a **change** of recognition rates in part 2 and part 3 of the **Bank Note** dataset. The reason for this is because there is a change in the data that we are giving to the code. Recognition rates depend a lot on the type of the data and the values of the data. That is why we are finding a difference. And this difference is expected.

Another important observation is: There is no change for **Seeds** dataset, since the data is the same.

3) Both the versions of the part 3 of the project give similar recognition rates. This is because the data that we are considering is the same. There is one another reason to this too. That is, since we are considering the best feature (which is the same in both the versions), the recognition rates are similar.

SUMMARY OF RESULTS

PART 1:

1) The best feature for Bank note dataset is feature “1” with a Fisher's discriminant ratio of 2.2003.
2) The best feature for Seeds dataset is feature “2” with a Fisher's discriminant ratio of 33.8742.

PART 2:

1) For bank dataset, feature “1” is the best feature. Its recognition rate calculated by the recognizer is 84.3%.
2) For Seeds dataset, feature “2” as the best feature. Its recognition rate calculated by the recognizer is 79.05%.

PART 3 VERSION 1:

1) Final recognition rate for Bank note dataset using best feature recognizer = 83.527%
   The best feature = feature “1”
2) Final recognition rate for Seeds dataset using best feature recognizer = 79.05%
   The best feature = feature “2”

PART 3VERSION 2:

1) Best feature for Bank note dataset using recognition rate is feature number = 1
   Recognition rate for the best feature = 83.53%
2) Best feature for Seeds dataset using recognition rate is feature number = 1
   Recognition rate for the best feature = 79.05%

CONCLUSION

Bayesian classifier is modeled using MATLAB for two data sets, and the results are consistent in both the versions of the project. We can further test this on a few more datasets. The procedure of implementation was concise, and this can be further enhanced to solve real world scenarios like Filtering of Emails etc based on the requirement.

REFERENCES:

[1] https://archive.ics.uci.edu/ml/datasets/banknote+authentication

[2] https://archive.ics.uci.edu/ml/datasets/seeds