UE17CS303-Machine Learning Assignment

Classification of stars and quasars using Naïve Bayes Classifier

AKASH P S (PES1201701773)
Computer Science
PES University
Bangalore, India
akkuin003@gmail.com

***Abstract*—This project deals with the classification of stars and quasars. Naïve Bayes classifier is used to classify the two data sets. The classification has been performed on four data catalogs. Accuracy and hence the corresponding efficiency of the classifier have also been indicated. Confusion matrix is used to calculate the accuracy for each case.**

Keywords—Naïve Bayes Classifier, Accuracy, Confusion Matrix, Data Set.

I. INTRODUCTION

A lot of surveys indicate the difficulty of classifying stars and quasars. Photometric surveys require such classification as the one discussed in the present project in order to differentiate the source of light correctly. There are several classification methods which can be used. K nearest neighbours, Support Vector machines, Decision trees are some of the methods used. This project however uses Naïve Bayes classification owing to its ease of implementation and an acceptable value of accuracy given by it. Naïve Bayes classification is a probabilistic classifier based on the Bayes theorem. Confusion matrix is used to obtain an accuracy value. Accuracy is computed for three values of split ratio to show the consistency of the result for different sizes of training data.

II. PROBLEM STATEMENT

Classify stars and quasars by using any classifier of choice with an acceptable accuracy.

Four catalogs containing data are provided.

III. ML TECHNIQUES EMPLOYED

Several machine learning techniques were employed in completing this project.

A. *Naïve Bayes Classifier*

Naïve Bayes is a conditional probability model.

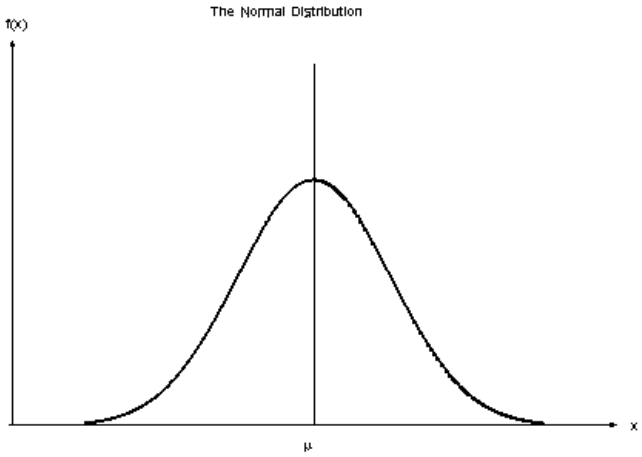
$$p(C_k \mid \mathbf{x}) = \frac{p(C_k)\, p(\mathbf{x} \mid C_k)}{p(\mathbf{x})}$$

If x represents the n different independent variables, C_k is the set of k outcomes.

Given x is true the probability of occurrence of C_k is probability of C_k times probability of x given C_k is true divided by probability of x.

This is the basic probability model used in this project.

Gaussian Naïve Bayes is used because predictors take almost a continuous value. The values are assumed to be from a Gaussian distribution



The formula used for conditional probability is:

$$P(x_i|y) = \frac{1}{\sqrt{2\pi\sigma_y^2}} exp\left(-\frac{(x_i-\mu_y)^2}{2\sigma_y^2}\right)$$

Where μ_y represents the mean of the distribution.

σ_y represents the standard deviation.

B. *Confidence Matrix*

- Confidence matrix is used to calculate the accuracy of the classification.
- The observed class values are given in the csv files. We have to predict the classes and compare them with the given classes in order to find the accuracy of the classification.
- Positive is the number of real positive cases in the data
- Negative is the number of real negative cases in the data
- A confidence matrix contains the following elements:
  - True Positive
  - True negative
  - False Positive
  - False Negative
- True Positive (TP): When it is predicted to be positive and is really positive.
- True Negative (TN): When it is predicted to be negative and is really negative.

- False Positive (FP): When it is predicted to be positive but is really positive. Type I error
- False Negative (FN): When it is predicted to be negative but is really positive. Type II error.

C. Equations

Sensitivity=(TP)/(TP+FN)

Specificity=(TN)/(TN+FP)

Precision=(TP)/(TP+FP)

Recall=(TP)/(TP+FN)

Accuracy=(TP+TN)/(TP+TN+FP+FN)

F score=2*(Precision*Recall)/(Precision+Recall)

D. Implementation of Naïve Bayes classifier

1. Load Csv Files

The csv file is first purified by removing columns such as the redshift, predicted class and other unwanted columns at the beginning. The file is then read using csv reader. The first line containing the headers is skipped and the resulting dataset is returned.

2. Train the data

The dataset is first split according to the split ratio. In this project three split ratios have been used. We select random data points from the dataset to form the training dataset after replacement until the size of the training set is the maximum.

3. Divide the dataset into two indices 1 and 0

The mean and standard deviation for each data point is calculated. If standard deviation equals zero, I have given a small value of 0.001 because standard deviation can not be zero for calculation of probability density function. This is called as smoothing. Depending on the value of observed data the dataset is included in 0 or 1.

4. Normal probability distribution

Probability of each data point is found out by using the probability distribution function of Gaussian or Normal probability distribution. Probability is predicted for each class and is stored.

5. Check accuracy

If class predicted by the classifier is the same as the observed class then the accuracy is increased. Accuracy is then returned.

IV. Summary of Result

The classifier worked well on the first three catalogues. In the below table I have shown accuracies of the classifier on all four catalogues for split ratio = 0.80

Catalogue	**Accuracy in %**
cat1.csv	**90.76**
cat2.csv	**91.64**
cat3.csv	**88.70**
cat4.csv	**71.25**

This shows that the classifier works well for the first three csv files where the accuracy is around ninety percentage. However, for the last csv file accuracy obtained is comparatively less.

The images given below show the results for cat1.csv with different split ratios namely 0.4, 0.6, 0.8.

1) Split ratio=0.8

```
akash@akash-VirtualBox:~/MLproject$ python3 bayesclassifier.py

FOR SRATIO =  0.8


 Confusion Matrix

      12       2
      11     105
False Positives
 [11  2]
False Negatives
 [ 2 11]
True Positives
 [ 12 105]
True Negatives
 [105  12]
Sensitivity
 [0.85714286 0.90517241]
Specificity
 [0.90517241 0.85714286]
Precision
 [0.52173913 0.98130841]
Recall
 [0.85714286 0.90517241]
Accuracy
 0.9
Fscore
[0.64864865 0.94170404]
649
```

2) Split ratio=0.6

```
FOR SRATIO =  0.6


 Confusion Matrix

      18       3
      32     207
False Positives
 [32  3]
False Negatives
 [ 3 32]
True Positives
 [ 18 207]
True Negatives
 [207  18]
Sensitivity
 [0.85714286 0.86610879]
Specificity
 [0.86610879 0.85714286]
Precision
 [0.36       0.98571429]
Recall
 [0.85714286 0.86610879]
Accuracy
 0.8653846153846154
Fscore
[0.50704225 0.922049  ]
649
```

3) Split ratio = 0.4

```
FOR SRATIO =  0.4


 Confusion Matrix

      23        5
      36      326
False Positives
 [36  5]
False Negatives
 [ 5 36]
True Positives
 [ 23 326]
True Negatives
 [326  23]
Sensitivity
 [0.82142857 0.90055249]
Specificity
 [0.90055249 0.82142857]
Precision
 [0.38983051 0.98489426]
Recall
 [0.82142857 0.90055249]
Accuracy
 0.8948717948717949
Fscore
[0.52873563 0.94083694]
649
```

CONCLUSION

This concludes that naïve Bayes classification is a good classifier and hence can be used to classify the stars and quasars. The last catalogue could not be classified to a very good accuracy because it was skewed towards the quasars.

An alternative to naïve Bayes classifier can be decision trees which gives better results when compared to Naïve Bayes classification.