Data Mining - Project

Naïve Bayes Classifier

Yassine Berrehouma

**1- Approach:**

During the Naive Bayes approach you make all variables conditionally independent based on a certain outcome. This algorithm is an intuitive method that utilizes the probabilities of every single attribute held by each class to output a prediction. From there you compute your confusion matrix and evaluate accuracy as shown in details through those steps:

1. **Clean Data**: First of all, once the data is loaded from the CSV file, we store in a data frame and start the data cleaning process which means handling errors and replacing non numerical attributes with numeric to be able to compute those values as well.
2. **Handle Data**: split the data it into test datasets that we can exploit to get the accuracy and training that Naïve Bayes can exploit to output predictions. Depending on the test and train ratio we have to split the data set arbitrary, upon the ratios, into train and dataset.
3. **Summarize Data**: To be able to make predictions and compute the probabilities, we must first outline the properties in the training dataset which includes the mean and the standard deviation by class value, for each attribute.
4. **Make a Prediction**: Output a single prediction with the use of the summaries of the dataset (selecting the class with the largest probability as the prediction).
5. **Make Predictions**: Taking into account the summarized training dataset and the test dataset, output prediction for every single test instance.
6. **Evaluate Accuracy**: Get the accuracy of predictions outputted for a test dataset as the percentage true from all the computed predictions.
7. **Confusion Matrix**: Generate true and false positive and true and false negative which will be used to calculate the recall and precision values.
8. **Graph**: Once done with all the steps above, the final values will be gathered and matched with it representative test and train ratio then print out its graphs where the Y- axis will either represent the precision, recall or accuracy value and the x axis holds out the splits (as shown below)

**2- Output:**

C:\Users\ycberrehouma\venv\naive-bayes\Scripts\python.exe C:/Users/ycberrehouma/PycharmProjects/naive-bayes/main.py

Split 45222 rows into train=27133 and test=18089 rows

Accuracy: 81.76792525844436 %

Recall: 66.5524135961154 %

Precision: 52.27731658065964 %

-------

Split 45222 rows into train=31655 and test=13567 rows

Accuracy: 81.76457580894818 %

Recall: 66.74285714285715 %

Precision: 52.251714882195046 %

-------

Split 45222 rows into train=36177 and test=9045 rows

Accuracy: 81.85737976782754 %

Recall: 68.15892557358701 %

Precision: 53.18777292576419 %

Process finished with exit code 0

