NAIVE BAYES CLASSIFIER

Prepared by

Büşra Erkoç

Classification of Email as Spam or Non-Spam using Naive Bayes

1.Introduction

The purpose of study:

Teaching the classification of emails (spam or not) by using training sets to Naive Bayes Classifier distributions. The data set was divided into training sets and test sets at a certain split rate. The effects of training set and testing set ratios on the classification were checked using different split rates. The split rate that gave the best result was determined and the distributions were study at this rate. Model was created by using Naive Bayes distributions. The model was trained by training set. The label of the test set was predicted to the trained model. The expected result and predicted result was evaluated based on confusion matrix and accuracy values.

2. Material and Method

a. Dataset Description

Dataset web page :https://archive.ics.uci.edu/ml/datasets/spambase

Dataset url : https://archive.ics.uci.edu/ml/machine-learning-databases/spambase/spambase.data

I used the spambase data set in the uci machine learning repository. 48 continuous real [0,100] attributes of type word_freq_WORD , 6 continuous real [0,100] attributes of type char_freq_CHAR, 1 continuous real [1,...] attribute of type capital_run_length_average, 1 continuous integer [1,...] attribute of type capital_run_length_longest, 1 continuous integer [1,...] attribute of type capital_run_length_total, 1 nominal {0,1} class attribute of type spam. The last attribute is class attribute. It indicate the e-mail is spam(1) or not(0).

Number of Instances: 4601

Number of Attributes : 57

Attribute characteristics is Integer, Real.

Attributes list:

word_freq_make:	continuous.	word_freq_lab:	continuous.
word_freq_address:	continuous.	word_freq_labs:	continuous.
word_freq_all:	continuous.	word_freq_telnet:	continuous.
word_freq_3d:	continuous.	word_freq_857:	continuous.
word_freq_our:	continuous.	word_freq_data:	continuous.

word_freq_over: continuous. word_freq_415: continuous.

word_freq_remove: continuous. word_freq_85: continuous.

word_freq_internet: continuous. word_freq_technology: continuous.

word_freq_order: continuous. word_freq_1999: continuous.

word_freq_mail: continuous. word_freq_parts: continuous.

word_freq_receive: continuous. word_freq_pm: continuous.

word_freq_will: continuous. word_freq_direct: continuous.

word_freq_people: continuous. word_freq_cs: continuous.

word_freq_report: continuous. word_freq_meeting: continuous.

word_freq_addresses: continuous. word_freq_original: continuous.

word_freq_free: continuous. word_freq_project: continuous.

word_freq_business: continuous. word_freq_re: continuous.

word_freq_email: continuous. word_freq_edu: continuous.

word_freq_you: continuous. word_freq_table: continuous.

word_freq_credit: continuous. word_freq_conference: continuous.

word_freq_your: continuous. char_freq_;: continuous.

word_freq_font: continuous. char_freq_(: continuous.

word_freq_000: continuous. char_freq_[: continuous.

word_freq_money: continuous. char_freq_!: continuous.

word_freq_hp: continuous. char_freq_$: continuous.

word_freq_hpl: continuous. char_freq_#: continuous.

word_freq_george: continuous. capital_run_length_average: continuous.

word_freq_650: continuous. capital_run_length_longest: continuous.

capital_run_length_total: continuous.

b. Theoretical details on the topic

Naive Bayes Classifier is based on Bayes’ theorem. Bayes theorem is proposed by Thomas Bayes (1702-1761). Bayes' theorem describes the probability of an event, based on prior knowledge of conditions that might be related to the event.

Bayes’ Theorem Formula:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

- P(A | B) is a conditional probability. (the likelihood od event A occurring given that B is true.)
- P(B | A) is a conditional probability.(the likelihood of event B occurring given that A is true.
- P(A) is a probability of event A
- P(B) is a probability of event B.

The Naive Bayes Classifier calculates the probability of each state for an element and classifies it based on the highest probability value. With the probability operations on the training data, the test data presented to the system are operated according to the previously obtained probability values and the class of the test data given is tried to be determined. It can do very successful works with a little training data. If a value in the test set has an unobservable value in the education set, it returns 0 as the probability value, that is, it cannot estimate. This condition is generally known as Zero Frequency. Correction techniques can be used to resolve this situation. One of the simplest correction techniques is known as Laplace prediction.

Examples of usage areas are real-time prediction, multi-class prediction, text classification, spam filtering and recommendation systems.

In Naive Bayes classification, the commonly used probability distributions include Gaussian distribution, Multinomial distribution, Bernoulli distribution. Feature distributions creates difference between them.

Gaussian Naive Bayes : In Gaussian Naive Bayes, continuous values associated with each feature are assumed to be distributed according to a Gaussian distribution. A Gaussian distribution is also called Normal distribution. When the estimators take a continuous value and are not discrete, we assume that these values are sampled from a gaussian distribution.

Multinomial Naive Bayes : Feature vectors represent the frequencies with which certain events have been generated by a multinomial distribution. This is the event model typically used for document classification.

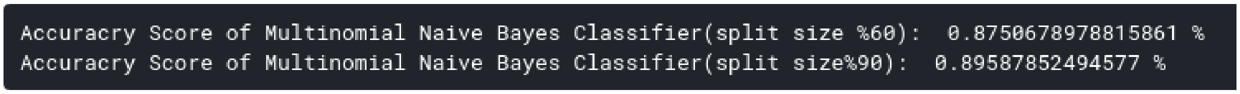
Bernoulli Naive Bayes: It is similar to multinomial naive bayes, but estimators are boolean variables. Features are independent booleans (binary variables) describing inputs.

c. Python codes

I did my study using a jupyter notebook from my Kaggle account. Jupyter notebook link in my Kaggle account:

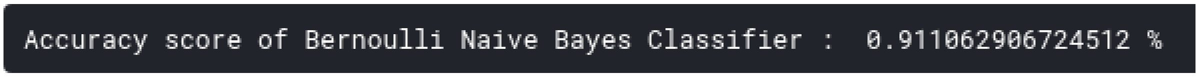
https://www.kaggle.com/busraerkoc/classification-of-email-as-spam-or-non-spam

3. Result and Discussion

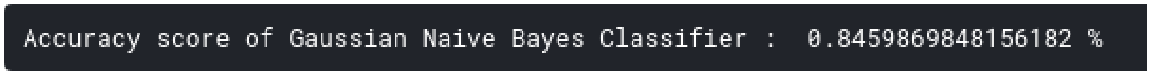


```
Accuracry Score of Multinomial Naive Bayes Classifier(split size %60):  0.8750678978815861 %
Accuracry Score of Multinomial Naive Bayes Classifier(split size%90):  0.89587852494577 %
```

I checked the effect of the split rate on the model. First I trained with a set of %60 training set. Then I trained with %90 of the training set. Looking at the accuracy values, the model with higher training set rate has classified more accurately. Therefore I continue my study with split size 90% training data.

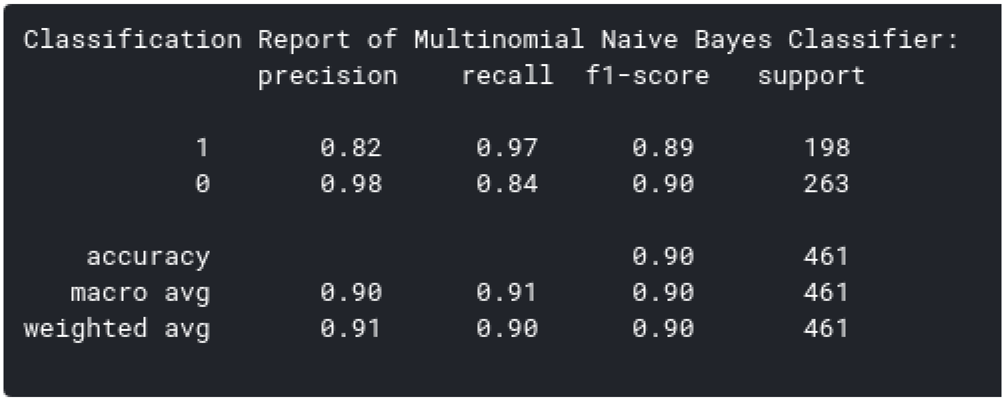


```
Accuracy score of Bernoulli Naive Bayes Classifier :  0.911062906724512 %
```

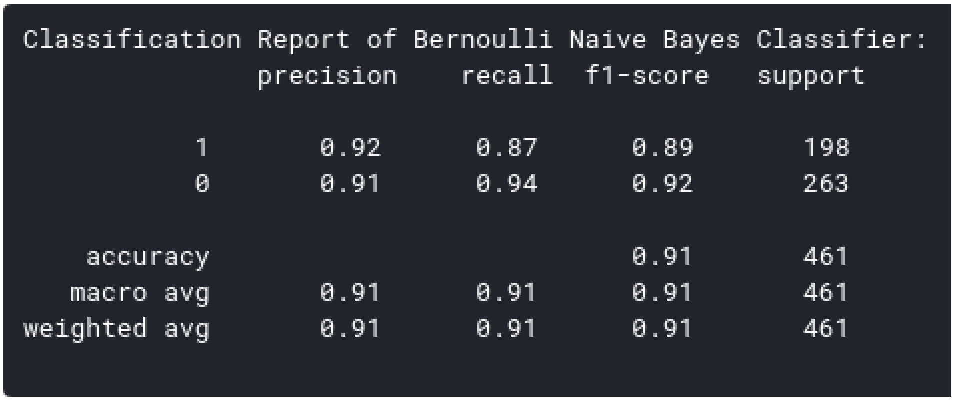


```
Accuracy score of Gaussian Naive Bayes Classifier :  0.8459869848156182 %
```

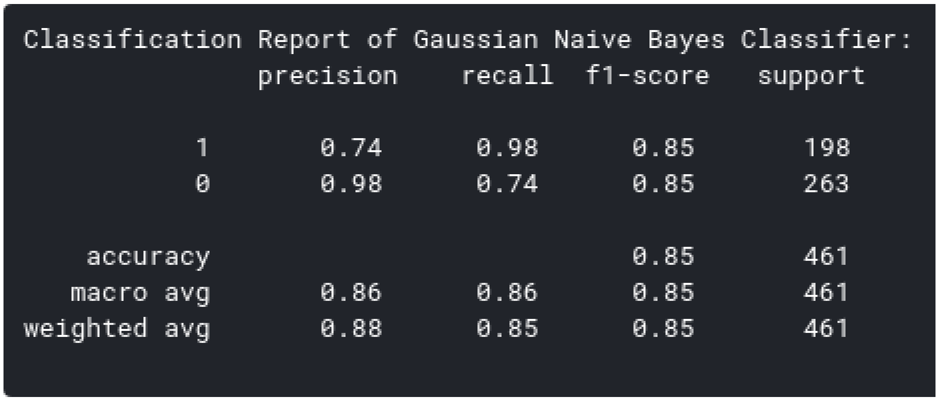
I applied Multinomial, Bernoulli and Gaussian Naive Bayes separately. The highest accuracy score is Bernoulli Naive Bayes Classifier.



```
Classification Report of Multinomial Naive Bayes Classifier:
              precision    recall  f1-score   support

           1       0.82      0.97      0.89       198
           0       0.98      0.84      0.90       263

    accuracy                           0.90       461
   macro avg       0.90      0.91      0.90       461
weighted avg       0.91      0.90      0.90       461
```



```
Classification Report of Bernoulli Naive Bayes Classifier:
              precision    recall  f1-score   support

           1       0.92      0.87      0.89       198
           0       0.91      0.94      0.92       263

    accuracy                           0.91       461
   macro avg       0.91      0.91      0.91       461
weighted avg       0.91      0.91      0.91       461
```

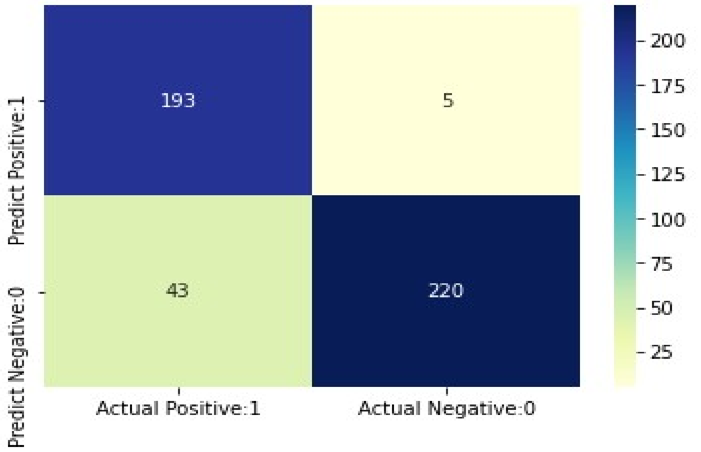


```
Classification Report of Gaussian Naive Bayes Classifier:
              precision    recall  f1-score   support

           1       0.74      0.98      0.85       198
           0       0.98      0.74      0.85       263

    accuracy                           0.85       461
   macro avg       0.86      0.86      0.85       461
weighted avg       0.88      0.85      0.85       461
```

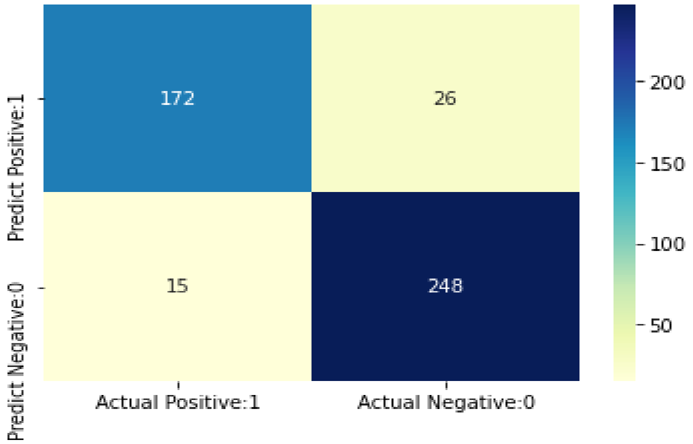
When we check at the Classification Report of distributions:
There were 198 label 1 and 263 label 0 in total. It classified label 1s better than label 0.
Multinomial Naive Bayes Classifier class label 1 correctly classified 97% of those. Class label 0 correctly classified 84% of those with correctly. It classified label 1s better than label 0.

Bernoulli Naive Bayes Classifier class label 1 correctly classified 87% of those. Class label 0 correctly classified 94% of those with correctly. It classified label 0s better than label 1.

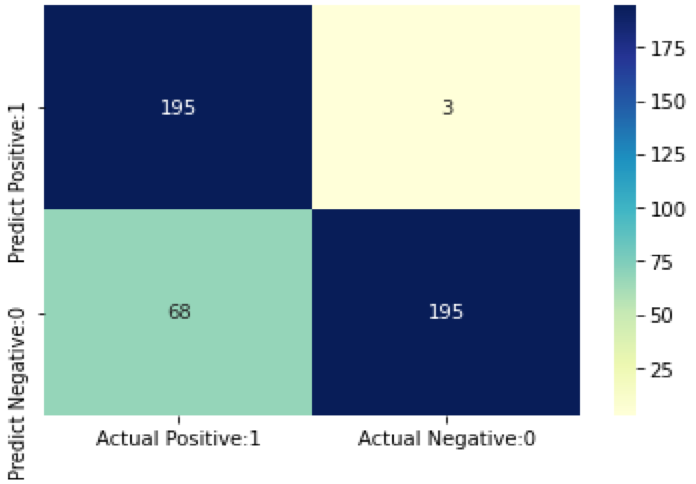
Gaussian Naive Bayes Classifier class label 1 correctly classified 98% of those. Class label 0 correctly classified 74% of those with correctly. It classified label 1s better than label 0.



Confusion Matrix of Multinomial Naïve Bayes



Confusion Matrix of Bernoulli Naïve Bayes



Confusion Matrix of Gaussian Naïve Bayes

When I check confusion matrix of distributions:
I can learn True Positive(TP), True Negative(TN), False Positive(FP), False Negatives(FN) values.
Multinomial Naive Bayes Classifier predicted correctly 413 instances label.

Bernoulli Naive Bayes Classifier predicted correctly 420 instances label.

Gaussian Naive Bayes Classifier predicted correctly 390 instances label.

As a result I conclude that Bernoulli Naïve Bayes Classifier gave the optimal and best result for this study.

4. References

- https://en.wikipedia.org/wiki/Naive_Bayes_classifier
- https://www.kaggle.com/prashant111/naive-bayes-classifier-tutorial
- https://archive.ics.uci.edu/ml/datasets/spambase
- https://scikit-learn.org/stable/modules/generated/sklearn.metrics.classification_report.html
- https://scikit-learn.org/stable/modules/naive_bayes.html
- Data Mining: Concepts and Techniques", by Jiawei Han, Micheline Kamber and Jian Pei (3rd edition)