

Implementation of Multinomial Naïve Bayes from scratch

Theory:

The Multinomial Naive Bayes algorithm is a helpful tool for classifying text. It assumes that features, often related to word frequencies, are independent given the class label. By focusing on counts and probabilities, it predicts the likelihood of observing different feature values in different categories. This makes it useful for tasks like spotting spam or analyzing sentiment. Multinomial Naive Bayes is efficient and commonly used in natural language processing because of its simplicity and effectiveness with large datasets.

we calculate the likelihood of class A. It is based on the formula below:

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

Objective:


- To complete the fit(), predict(), and evaluate() methods.
- To implement the Naive Bayes algorithm from scratch without using any libraries.
- To find out the accuracy of our implemented model.

Result Analysis:

The accuracy of the implemented model is as follows:

```
        correct = correct + 1
    print("Accurate :")
    print(correct)
    print("Message :")
    print(len(y_test))
    correct = (correct/len(y_test)) * 100
    print("Accuracy :")
    print(correct)

evaluate(X_test, y_test)
```

 Accurate :
552
Message :
558
Accuracy :
98.9247311827957

In the test dataset, the model achieved a remarkable accuracy with 552 out of 558 messages correctly classified, resulting in an accuracy rate of 98.92%.