

PROGRAMMING ASSIGNMENT 3
NAÏVE BAYES IMPLEMENTATION

BY

THOTA SAI ABHISHEK – SXT210056

RAHUL GAURI – RXG200002

Part A

Part A - Top 3 Spam and Non Spam words and Accuracy, Precision, Recall and F1 scores

Log odds ratio for top 3 spam words:

```
thomas  4.971528135965574
brazos  4.937041959894405
search  4.845234410641282
```

Log odds ratio for top 3 Non-spam words:

```
but     -5.21525689070451
i       -4.548427018461939
new     -4.539731311494385
```

```
{'Simple Naive Bayes': {'accuracy': 0.8447164948453608, 'precision': 0.8666666666666667, 'recall': 0.5488888888888889, 'f1_score': 0.6721088435374148}}
```

We have reported the top 3 spam and top 3 non spam words according to their log likelihoods score and also for the implementation of Simple_Naive_Bayes (Simple_NB()) , we have reported the accuracy, precision, recall and f1_score

Part B

We have used our implementation of Simple_NB() to predict whether a sentence is Spam or Not Spam based on their posterior likelihoods.

Part B:

```
Sample Email: Congratulations! Your raffle ticket has won yourself a house. Click on the link to avail prize
Spam Likelihood: 5.46194750432022e-221
Not Spam Likelihood: 6.254354629104953e-209
{0: 6.254354629104953e-209, 1: 5.46194750432022e-221}
```

```
Sample Email: Hello. This email is to remind you that your project needs to be submitted this week
Spam Likelihood: 1.0635249947035889e-215
Not Spam Likelihood: 3.6213304968004355e-205
{0: 3.6213304968004355e-205, 1: 1.0635249947035889e-215}
```

```
Sample Email: Hello. This is Machine Learning class CS6375
Spam Likelihood: 1.4931671906139711e-186
Not Spam Likelihood: 9.654583351728176e-181
{0: 9.654583351728176e-181, 1: 1.4931671906139711e-186}
```

```
Sample Email: Hi there! Have a great day!
Spam Likelihood: 1.5468638635433628e-172
Not Spam Likelihood: 5.8422755778120265e-167
{0: 5.8422755778120265e-167, 1: 1.5468638635433628e-172}
```

Part C

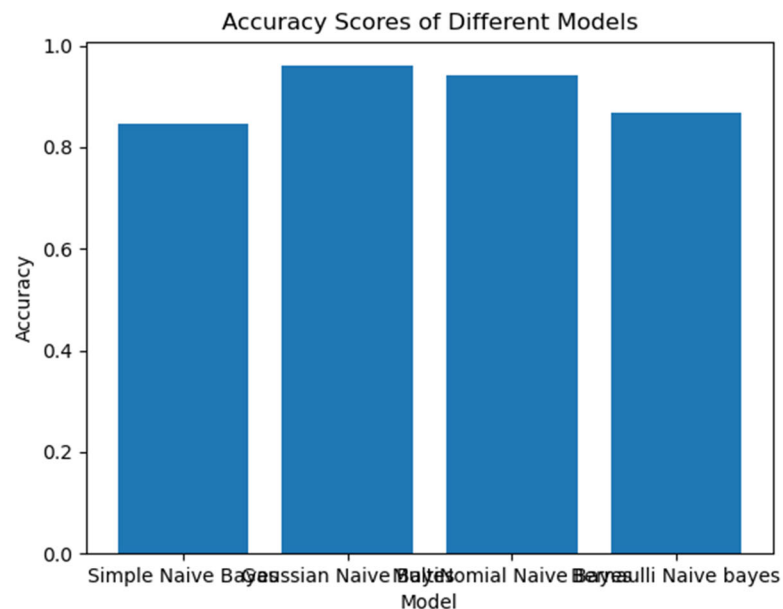
We have used scikit-learn function to get GaussianNB, MultinomialNB and BernoulliNB to get scores of parameters like accuracy, recall, precision and f1.

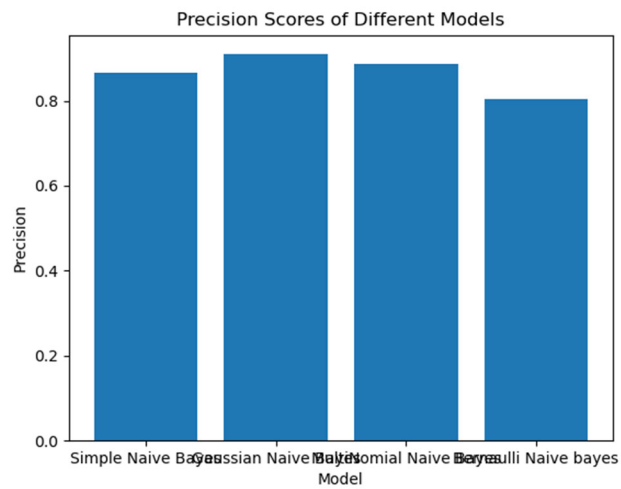
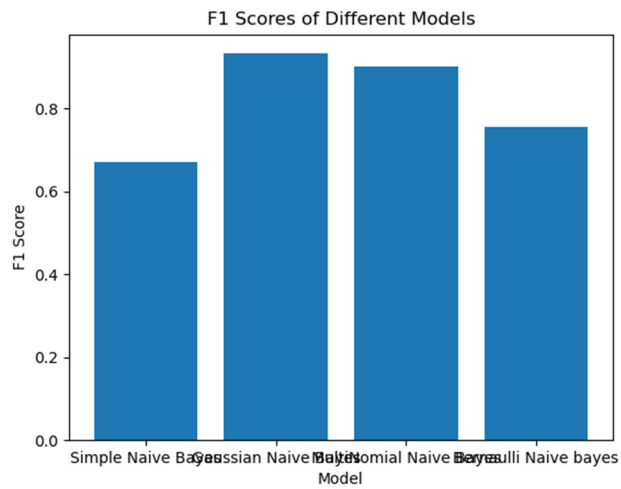
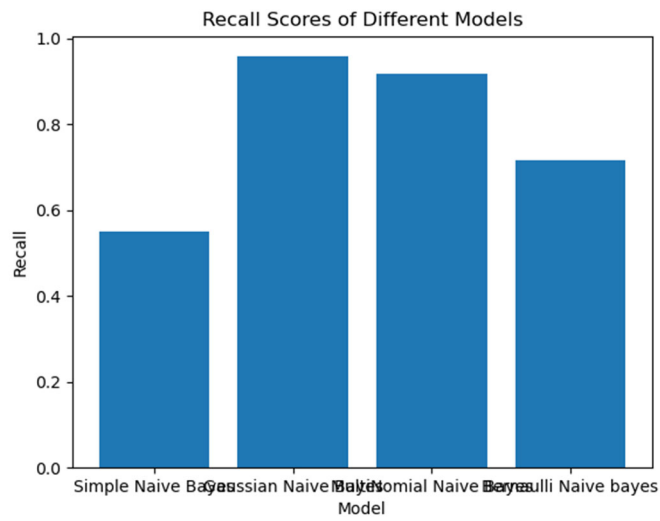
Part C - Scikit-learn prediction of Different Naive Bayes implementation

```
{'Bernoulli Naive bayes': {'accuracy': 0.8666237113402062,
                          'f1_score': 0.7567567567567567,
                          'precision': 0.8029925187032418,
                          'recall': 0.7155555555555555},
 'Gaussian Naive Bayes': {'accuracy': 0.9600515463917526,
                          'f1_score': 0.9329004329004329,
                          'precision': 0.9092827004219409,
                          'recall': 0.9577777777777777},
 'MultiNomial Naive Bayes': {'accuracy': 0.9420103092783505,
                             'f1_score': 0.9017467248908297,
                             'precision': 0.8862660944206009,
                             'recall': 0.9177777777777778},
 'Simple Naive Bayes': {'accuracy': 0.8447164948453608,
                       'f1_score': 0.6721088435374148,
                       'precision': 0.8666666666666667,
                       'recall': 0.5488888888888889}}
```

Part D

We have used matplotlib.pyplot to draw bar plots of scores of parameters like accuracy, f1, recall and precision of different Naïve Bayes implementation like Simple_NB, Gaussian NB, Multinomial NB and Bernoulli NB.





Part E – We have generated obj file of the model using pickle