UIT2601-Pattern Recognition and machine Learning

Experiment 2

Name: Saruleka P

Reg no: 3122235002114

Class: IT-C

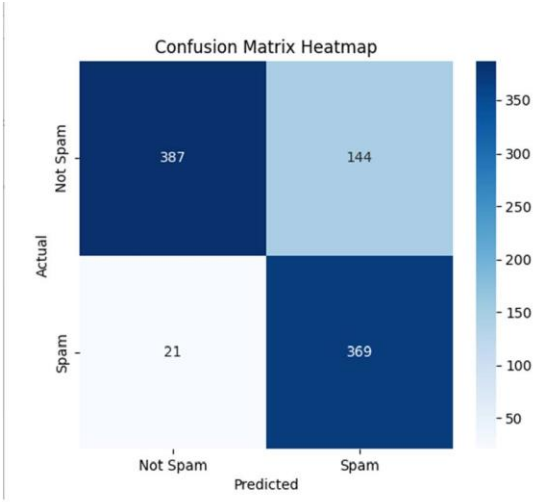
Aim: **Spam Email Detection Using Naïve Bayes Classifier**

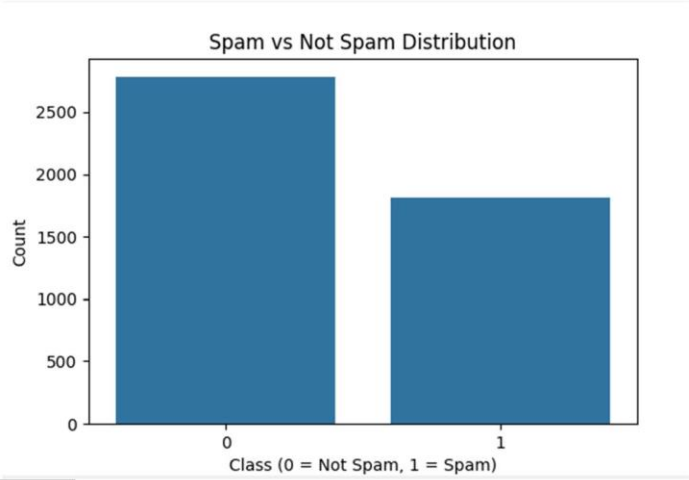
CODE:

```
import pandas as pd

import numpy as np

from sklearn.model_selection import train_test_split

from sklearn.naive_bayes import GaussianNB

from sklearn.metrics import accuracy_score, confusion_matrix,
classification_report


df = pd.read_csv(r"spambase.data", header=None)


print(df.head(10))

print(df.shape)

print(df.isnull().sum())

print(df.duplicated().sum())


x = df.iloc[:, :-1]

y = df.iloc[:, -1]


print(y.value_counts())
```

```
X_train, X_test, Y_train, Y_test = train_test_split(
    x, y, test_size=0.2, random_state=42
)

model = GaussianNB()
model.fit(X_train, Y_train)

y_pred = model.predict(X_test)

accuracy = accuracy_score(Y_test, y_pred) * 100
cm = confusion_matrix(Y_test, y_pred)
report = classification_report(Y_test, y_pred)

print("Accuracy:", accuracy)
print("Confusion Matrix:\n", cm)
print("Classification Report:\n", report)
```

OUTPUT:






```
Name: count, dtype: int64
Accuracy: 82.08469055374593
Confusion Matrix:
 [[387 144]
 [ 21 369]]
Classification Report:
              precision    recall  f1-score   support

           0       0.95      0.73      0.82       531
           1       0.72      0.95      0.82       390

    accuracy                           0.82       921
   macro avg       0.83      0.84      0.82       921
weighted avg       0.85      0.82      0.82       921
```

```
False Positives: 144
False Negatives: 21
Class Distribution Ratio:
 57
0    0.605955
1    0.394045
Name: proportion, dtype: float64
PS Z:\vihashni\ML>
```

```
PS Z:\vihashni\ML> & "C:/Program Files/Python313/python.exe" z:/vihashni/ML/gauss.py
     0     1     2     3     4     5     6     7     8     9     10    11    12    13    14    15  ...    42    43    44    45    46    47    48     49    50     51     52     53     54    55    56  57
0  0.00  0.64  0.64  0.0   0.32  0.00  0.00  0.00  0.00  0.00  0.00  0.64  0.00  0.00  0.00  0.32  ...  0.00  0.00  0.00  0.00  0.0   0.0   0.00  0.000  0.0   0.778  0.000  0.000  3.756  61    278   1
1  0.21  0.28  0.50  0.0   0.14  0.28  0.21  0.07  0.00  0.94  0.21  0.79  0.65  0.21  0.14  0.14  ...  0.00  0.00  0.00  0.00  0.0   0.0   0.00  0.132  0.0   0.372  0.180  0.048  5.114  101   1028  1
2  0.06  0.00  0.71  0.0   1.23  0.19  0.19  0.12  0.64  0.25  0.38  0.45  0.12  0.00  1.75  0.06  ...  0.12  0.00  0.06  0.06  0.0   0.0   0.01  0.143  0.0   0.276  0.184  0.010  9.821  485   2259  1
3  0.00  0.00  0.00  0.0   0.63  0.00  0.31  0.63  0.31  0.63  0.31  0.31  0.31  0.00  0.00  0.31  ...  0.00  0.00  0.00  0.00  0.0   0.0   0.00  0.137  0.0   0.137  0.000  0.000  3.537  40    191   1
4  0.00  0.00  0.00  0.0   0.63  0.00  0.31  0.63  0.31  0.63  0.31  0.31  0.31  0.00  0.00  0.31  ...  0.00  0.00  0.00  0.00  0.0   0.0   0.00  0.135  0.0   0.135  0.000  0.000  3.537  40    191   1
5  0.00  0.00  0.00  0.0   1.85  0.00  0.00  1.85  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  ...  0.00  0.00  0.00  0.00  0.0   0.0   0.00  0.223  0.0   0.000  0.000  0.000  3.000  15    54    1
6  0.00  0.00  0.00  0.0   1.92  0.00  0.00  0.00  0.00  0.64  0.96  1.28  0.00  0.00  0.00  0.96  ...  0.00  0.00  0.00  0.00  0.0   0.0   0.00  0.054  0.0   0.164  0.054  0.000  1.671  4     112   1
7  0.00  0.00  0.00  0.0   1.88  0.00  0.00  1.88  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  ...  0.00  0.00  0.00  0.00  0.0   0.0   0.00  0.206  0.0   0.000  0.000  0.000  2.450  11    49    1
8  0.15  0.00  0.46  0.0   0.61  0.00  0.30  0.00  0.92  0.76  0.76  0.92  0.00  0.00  0.00  0.00  ...  0.30  0.00  0.00  0.00  0.0   0.0   0.00  0.271  0.0   0.181  0.203  0.022  9.744  445   1257  1
9  0.06  0.12  0.77  0.0   0.19  0.32  0.38  0.00  0.06  0.00  0.00  0.64  0.25  0.00  0.12  0.00  ...  0.00  0.06  0.00  0.00  0.0   0.0   0.04  0.030  0.0   0.244  0.081  0.000  1.729  43    749   1

[10 rows x 58 columns]
(4601, 58)
```

Class imbalance impact:

- If one class dominates (usually Not Spam), the model may show high accuracy but poor recall for Spam.
- Precision and Recall become more important than Accuracy in such cases.
- A low Recall for Spam means many spam emails are missed.
- A low Precision means many normal emails are wrongly marked as spam.