```
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.naive_bayes import MultinomialNB
from sklearn.linear_model import LogisticRegression
from sklearn.svm import SVC
from sklearn.metrics import classification_report, confusion_matrix,accuracy_score
```

```
# Loading of dataset
data = pd.read_csv('/content/spam.csv', encoding='ISO-8859-1')
data
```

| | v1 | v2 | Unnamed: 2 | Unnamed: 3 | Unnamed: 4 |
|---|---|---|---|---|---|
| 0 | ham | Go until jurong point, crazy.. Available only ... | NaN | NaN | NaN |
| 1 | ham | Ok lar... Joking wif u oni... | NaN | NaN | NaN |
| 2 | spam | Free entry in 2 a wkly comp to win FA Cup fina... | NaN | NaN | NaN |
| 3 | ham | U dun say so early hor... U c already then say... | NaN | NaN | NaN |
| 4 | ham | Nah I don't think he goes to usf, he lives aro... | NaN | NaN | NaN |
| ... | ... | ... | ... | ... | ... |
| 5567 | spam | This is the 2nd time we have tried 2 contact u... | NaN | NaN | NaN |
| 5568 | ham | Will Ì_ b going to esplanade fr home? | NaN | NaN | NaN |
| 5569 | ham | Pity, * was in mood for that. So...any | NaN | NaN | NaN |

Next steps: Generate code with data View recommended plots

```
data.head()
```

| | v1 | v2 | Unnamed: 2 | Unnamed: 3 | Unnamed: 4 |
|---|---|---|---|---|---|
| 0 | ham | Go until jurong point, crazy.. Available only ... | NaN | NaN | NaN |
| 1 | ham | Ok lar... Joking wif u oni... | NaN | NaN | NaN |
| 2 | spam | Free entry in 2 a wkly comp to win FA Cup fina... | NaN | NaN | NaN |
| 3 | ham | U dun say so early hor... U c already then | NaN | NaN | NaN |

Next steps: Generate code with data View recommended plots

```
data.tail()
```

| | v1 | v2 | Unnamed: 2 | Unnamed: 3 | Unnamed: 4 |
|---|---|---|---|---|---|
| 5567 | spam | This is the 2nd time we have tried 2 contact u... | NaN | NaN | NaN |
| 5568 | ham | Will ì_ b going to esplanade fr home? | NaN | NaN | NaN |
| 5569 | ham | Pity, * was in mood for that. So...any other s... | NaN | NaN | NaN |
| 5570 | ham | The guy did some bitching but I acted | NaN | NaN | NaN |

```
data.shape
```

```
(5572, 5)
```

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5572 entries, 0 to 5571
Data columns (total 5 columns):
 #   Column      Non-Null Count  Dtype 
---  ------      --------------  ----- 
 0   v1          5572 non-null   object
 1   v2          5572 non-null   object
 2   Unnamed: 2  50 non-null     object
 3   Unnamed: 3  12 non-null     object
 4   Unnamed: 4  6 non-null      object
dtypes: object(5)
memory usage: 217.8+ KB
```

```
data.isna().sum()
```

```
v1              0
v2              0
Unnamed: 2    5522
Unnamed: 3    5560
Unnamed: 4    5566
dtype: int64
```

```
# Splitting data into X and y (target)
X = data['v2']
y = data['v1']
```

```
# Splitting data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

```
# Text vectorization (TF-IDF)
tfidf_vectorizer = TfidfVectorizer(max_features=5000)  # Can change parameter
X_train_tfidf = tfidf_vectorizer.fit_transform(X_train)
X_test_tfidf = tfidf_vectorizer.transform(X_test)
```

```
nb= MultinomialNB()
reg_model=LogisticRegression()
model=SVC()
lst_model=[nb,model,reg_model]
```

```
for i in lst_model:
  print("Model name: ",i)
  i.fit(X_train_tfidf,y_train)
  y_pred=i.predict(X_test_tfidf)
  print("***************************************")
  print(confusion_matrix(y_test,y_pred))
  print("Accuracy Score.............")
  print(accuracy_score(y_test,y_pred))
```

```
Model name:  MultinomialNB()
***************************************
[[965   0]
 [ 37 113]]
Accuracy Score.............
0.9668161434977578
Model name:  SVC()
***************************************
[[965   0]
 [ 22 128]]
Accuracy Score.............
0.9802690582959641
Model name:  LogisticRegression()
***************************************
[[964   1]
 [ 35 115]]
Accuracy Score.............
0.967713004484305
```