```python
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import warnings
warnings.filterwarnings('ignore')
from sklearn.model_selection import train_test_split

df = pd.read_csv("D:\College\emails.csv")

df.drop(['Email No.'], axis=1, inplace=True)
X = df.drop(['Prediction'], axis=1)
y = df['Prediction']

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.20, random_state = 0)

"FeatureScaling"
from sklearn.preprocessing import StandardScaler
sc = StandardScaler()
X_train = sc.fit_transform(X_train)
X_test = sc.transform(X_test)

"**************KNN**************"
from sklearn.neighbors import KNeighborsClassifier
model_KNN = KNeighborsClassifier(n_neighbors = 5)
model_KNN.fit(X_train, y_train)

y_pred = model_KNN.predict(X_test)

from sklearn.metrics import mean_squared_error,accuracy_score


mse = mean_squared_error(y_test, y_pred)
rmse = mean_squared_error(y_test, y_pred, squared=False)
ac = accuracy_score(y_test,y_pred)

print("\n")
print("KNN Results")
print(f'Accuracy: {ac}')
print(f'mse: {mse}')
print(f'rmse: {rmse}')
```

```
"***************SVM***************"
from sklearn.svm import SVC
model_SVC = SVC(C=1)
model_SVC.fit(X_train,y_train)

y_pred_SVC = model_SVC.predict(X_test)

from sklearn.metrics import mean_squared_error,accuracy_score


mse = mean_squared_error(y_test, y_pred_SVC)
rmse = mean_squared_error(y_test, y_pred_SVC, squared=False)
ac = accuracy_score(y_test,y_pred_SVC)

print("\n")
print("SVM Results")
print(f'Accuracy: {ac}')
print(f'mse: {mse}')
print(f'rmse: {rmse}')
```

Output -

```
In [1]: runfile('C:/Users/Dell/untitled0.py', wdir='C:/Users/Dell')


KNN Results
Accuracy: 0.8231884057971014
mse: 0.17681159420289855
rmse: 0.42048970760637955


SVM Results
Accuracy: 0.9352657004830918
mse: 0.0647342995169082
rmse: 0.254429360563808
```