**Name:** Muzammil Rauf Patel

**Class:** SYMCA **Div:** B

**Roll No:** 45 **Batch:** B2

**Course:** ML Lab

------------------------------------------------------------------------------------------------

**Assignment No: 5**

**Problem Statement:**

Design and implement SVM for classification with the given data set. Test for Accuracy and Precision. Also analyze the performance of SVM and KNN.

**------------------------------------------------------------------------------------------------**

**Code:**

import pandas as pd

from sklearn.model\_selection import train\_test\_split

from sklearn.preprocessing import StandardScaler

from sklearn.svm import SVC

from sklearn.metrics import confusion\_matrix, accuracy\_score

from sklearn import metrics

data = pd.read\_csv('email.csv')

print(data.head())

data['Prediction'] = data['Prediction'].replace(0,"spam")

data['Prediction'] = data['Prediction'].replace(1,"ham")

X = data.iloc[:,1:-1]

y = data.iloc[:,-1]

print(X,y)

# Split the data into training and test sets

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2,random\_state=0)

# Scale the features using StandardScaler

scaler = StandardScaler()

X\_train = scaler.fit\_transform(X\_train)

X\_test = scaler.transform(X\_test)

#This SVC class allows us to build a kernel SVM model (Linear as well as non-linear)

#The default value of the kernel is 'rbf'. Why 'rbf', because it is nonlinear and gives better results

#kernel can be one of the ('linear', 'poly', 'rbf', 'sigmoid')

svm = SVC (kernel = 'rbf',random\_state = 0)

svm.fit(X\_train, y\_train)

y\_pred = svm.predict(X\_test)

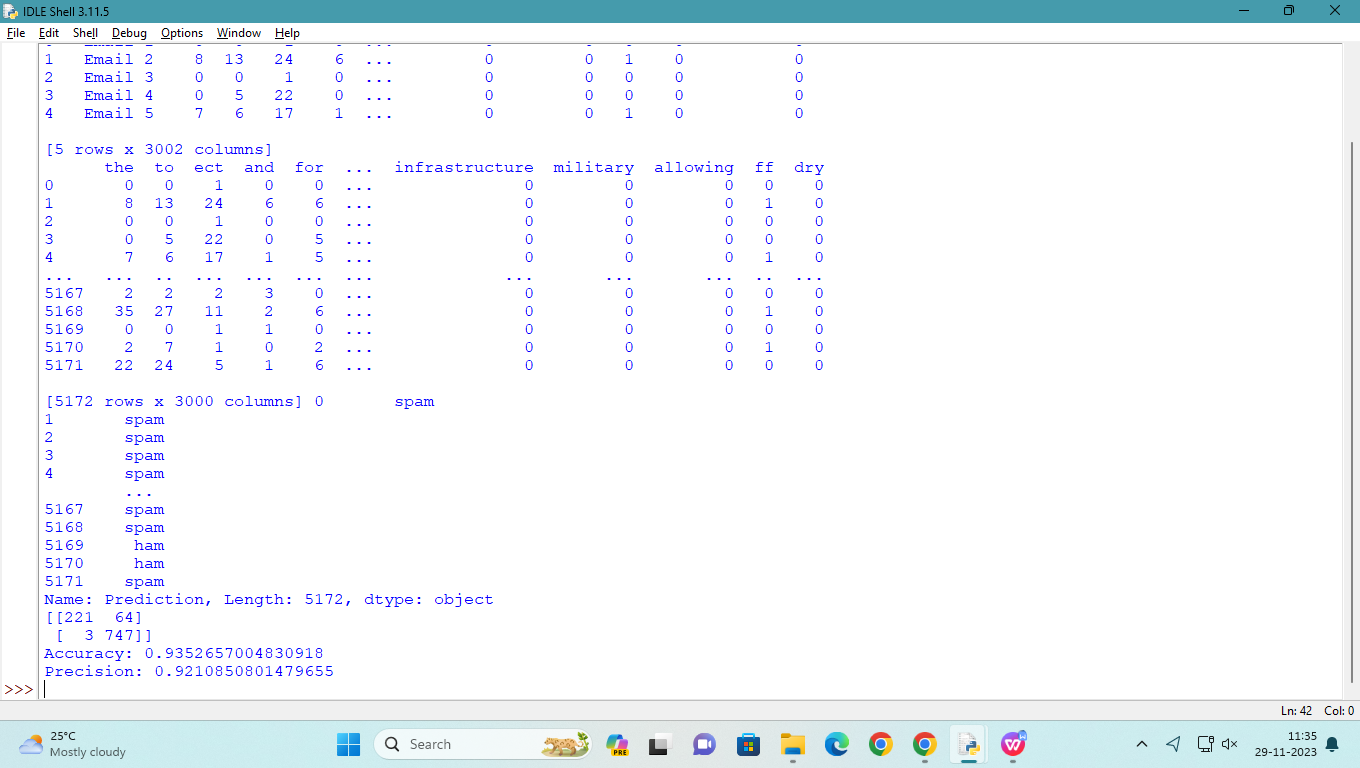
cm = confusion\_matrix(y\_test, y\_pred)

print(cm)

print("Accuracy:",metrics.accuracy\_score(y\_test, y\_pred))

print("Precision:",metrics.precision\_score(y\_test, y\_pred, pos\_label='spam'))

**Output:**

****