

assignment-4-mllab-ashwinravi

March 11, 2024

Ashwin Ravi

CSE-A

3122 21 5001 014

1 A4:Classification of Email Spam and MNIST data using SVM

Github link:- <https://github.com/SolitudeAsh/Machine-Learning-Laboratory/tree/main/A4>

2 Aim

Train a ML Model using SVM to Classification of Email Spam and perform all machine learning steps. A support vector machine (SVM) is a machine learning algorithm that uses supervised learning models to solve complex classification, regression, and outlier detection problems by performing optimal data transformations that determine boundaries between data points based on predefined classes, labels, or outputs.

```
[2]: # This Python 3 environment comes with many helpful analytics libraries
      ↵installed
      # It is defined by the kaggle/python Docker image: https://github.com/kaggle/
      ↵docker-python
      # For example, here's several helpful packages to load

      import numpy as np # linear algebra
      import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)

      # Input data files are available in the read-only "../input/" directory
      # For example, running this (by clicking run or pressing Shift+Enter) will list
      ↵all files under the input directory

      import os
      for dirname, _, filenames in os.walk("C:/Users/ashwi/Downloads/ML Lab/
      ↵A4_MLLab_AshwinRavi/input"):
          for filename in filenames:
              print(os.path.join(dirname, filename))
```

```
# You can write up to 20GB to the current directory (/kaggle/working/) that ↴  
gets preserved as output when you create a version using "Save & Run All"  
# You can also write temporary files to /kaggle/temp/, but they won't be saved ↴  
outside of the current session
```

```
C:/Users/ashwi/Downloads/ML Lab/A4_MLLab_AshwinRavi/input\prediction.csv  
C:/Users/ashwi/Downloads/ML Lab/A4_MLLab_AshwinRavi/input\spambase_csv.csv  
  
C:\Users\ashwi\AppData\Local\Temp\ipykernel_17700\2812097460.py:6:  
DeprecationWarning:  
Pyarrow will become a required dependency of pandas in the next major release of  
pandas (pandas 3.0),  
(to allow more performant data types, such as the Arrow string type, and better  
interoperability with other libraries)  
but was not found to be installed on your system.  
If this would cause problems for you,  
please provide us feedback at https://github.com/pandas-dev/pandas/issues/54466
```

```
import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)
```

3 Import Libraries

```
[3]: import pandas as pd  
import seaborn as sns  
from sklearn.preprocessing import MinMaxScaler  
from sklearn.model_selection import train_test_split  
from sklearn.linear_model import LogisticRegression  
from sklearn.neural_network import MLPClassifier  
from sklearn.ensemble import RandomForestClassifier  
from sklearn.tree import DecisionTreeClassifier  
from sklearn.neighbors import KNeighborsClassifier  
from sklearn.metrics import confusion_matrix  
import matplotlib.pyplot as plt  
from sklearn.metrics import accuracy_score  
from sklearn.metrics import f1_score  
from sklearn.metrics import recall_score  
from sklearn.metrics import precision_score  
from sklearn.metrics import precision_recall_curve  
from sklearn import svm  
from sklearn.metrics import classification_report  
from sklearn.metrics import roc_auc_score
```

4 Read Data

```
[4]: data=pd.read_csv("C:/Users/ashwi/Downloads/ML Lab/A4_MLLab_AshwinRavi/input/  
↳spambase_csv.csv")
```

5 Show 5 frist row of data

```
[5]: data.head()
```

```
[5]:    word_freq_make  word_freq_address  word_freq_all  word_freq_3d  \  
0          0.00           0.64           0.64           0.0  
1          0.21           0.28           0.50           0.0  
2          0.06           0.00           0.71           0.0  
3          0.00           0.00           0.00           0.0  
4          0.00           0.00           0.00           0.0  
  
    word_freq_our  word_freq_over  word_freq_remove  word_freq_internet  \  
0          0.32           0.00           0.00           0.00  
1          0.14           0.28           0.21           0.07  
2          1.23           0.19           0.19           0.12  
3          0.63           0.00           0.31           0.63  
4          0.63           0.00           0.31           0.63  
  
    word_freq_order  word_freq_mail  ...  char_freq_%3B  char_freq_%28  \  
0          0.00           0.00  ...           0.00           0.000  
1          0.00           0.94  ...           0.00           0.132  
2          0.64           0.25  ...           0.01           0.143  
3          0.31           0.63  ...           0.00           0.137  
4          0.31           0.63  ...           0.00           0.135  
  
    char_freq_%5B  char_freq_%21  char_freq_%24  char_freq_%23  \  
0          0.0           0.778           0.000           0.000  
1          0.0           0.372           0.180           0.048  
2          0.0           0.276           0.184           0.010  
3          0.0           0.137           0.000           0.000  
4          0.0           0.135           0.000           0.000  
  
    capital_run_length_average  capital_run_length_longest  \  
0                  3.756                   61  
1                  5.114                   101  
2                  9.821                   485  
3                  3.537                   40  
4                  3.537                   40  
  
    capital_run_length_total  class  
0                  278           1
```

```
1          1028      1
2          2259      1
3          191       1
4          191       1
```

[5 rows x 58 columns]

6 Show columns of data

```
[6]: x=1
for col in data.columns:
    print(f'columns {x} is:',col)
    x+=1
```

```
columns 1 is: word_freq_make
columns 2 is: word_freq_address
columns 3 is: word_freq_all
columns 4 is: word_freq_3d
columns 5 is: word_freq_our
columns 6 is: word_freq_over
columns 7 is: word_freq_remove
columns 8 is: word_freq_internet
columns 9 is: word_freq_order
columns 10 is: word_freq_mail
columns 11 is: word_freq_receive
columns 12 is: word_freq_will
columns 13 is: word_freq_people
columns 14 is: word_freq_report
columns 15 is: word_freq_addresses
columns 16 is: word_freq_free
columns 17 is: word_freq_business
columns 18 is: word_freq_email
columns 19 is: word_freq_you
columns 20 is: word_freq_credit
columns 21 is: word_freq_your
columns 22 is: word_freq_font
columns 23 is: word_freq_000
columns 24 is: word_freq_money
columns 25 is: word_freq_hp
columns 26 is: word_freq_hpl
columns 27 is: word_freq_george
columns 28 is: word_freq_650
columns 29 is: word_freq_lab
columns 30 is: word_freq_labs
columns 31 is: word_freq_telnet
columns 32 is: word_freq_857
```

```
columns 33 is: word_freq_data
columns 34 is: word_freq_415
columns 35 is: word_freq_85
columns 36 is: word_freq_technology
columns 37 is: word_freq_1999
columns 38 is: word_freq_parts
columns 39 is: word_freq_pm
columns 40 is: word_freq_direct
columns 41 is: word_freq_cs
columns 42 is: word_freq_meeting
columns 43 is: word_freq_original
columns 44 is: word_freq_project
columns 45 is: word_freq_re
columns 46 is: word_freq_edu
columns 47 is: word_freq_table
columns 48 is: word_freq_conference
columns 49 is: char_freq_%3B
columns 50 is: char_freq_%28
columns 51 is: char_freq_%5B
columns 52 is: char_freq_%21
columns 53 is: char_freq_%24
columns 54 is: char_freq_%23
columns 55 is: capital_run_length_average
columns 56 is: capital_run_length_longest
columns 57 is: capital_run_length_total
columns 58 is: class
```

7 Show data corr

```
[7]: data.corr()
```

```
[7]:
```

	word_freq_make	word_freq_address	word_freq_all	\
word_freq_make	1.000000	-0.016759	0.065627	
word_freq_address	-0.016759	1.000000	-0.033526	
word_freq_all	0.065627	-0.033526	1.000000	
word_freq_3d	0.013273	-0.006923	-0.020246	
word_freq_our	0.023119	-0.023760	0.077734	
word_freq_over	0.059674	-0.024840	0.087564	
word_freq_remove	0.007669	0.003918	0.036677	
word_freq_internet	-0.003950	-0.016280	0.012003	
word_freq_order	0.106263	-0.003826	0.093786	
word_freq_mail	0.041198	0.032962	0.032075	
word_freq_receive	0.188459	-0.006864	0.048254	
word_freq_will	0.105801	-0.040398	0.083210	
word_freq_people	0.066438	-0.018858	0.047593	
word_freq_report	0.036780	-0.009206	0.008552	
word_freq_addresses	0.028439	0.005330	0.122113	

word_freq_free	0.059386	-0.009117	0.063906
word_freq_business	0.081928	-0.018370	0.036262
word_freq_email	0.053324	0.033500	0.121923
word_freq_you	0.128243	-0.055476	0.139329
word_freq_credit	0.021295	-0.015806	0.031111
word_freq_your	0.197049	-0.018191	0.156651
word_freq_font	-0.024349	-0.008850	-0.035681
word_freq_000	0.134072	-0.020502	0.123671
word_freq_monev	0.188155	0.001984	0.041145
word_freq_hp	-0.072504	-0.043483	-0.087924
word_freq_hpl	-0.061686	-0.038211	-0.062459
word_freq_george	-0.066424	-0.030307	-0.108886
word_freq_650	-0.048680	-0.029221	-0.050648
word_freq_lab	-0.041251	-0.021940	-0.057726
word_freq_labs	-0.052799	-0.027508	-0.032547
word_freq_telnet	-0.039066	-0.018097	-0.038927
word_freq_857	-0.032058	-0.003326	-0.061870
word_freq_data	-0.041014	-0.024903	-0.054759
word_freq_415	-0.027690	-0.004303	-0.061706
word_freq_85	-0.044954	-0.024058	-0.048335
word_freq_technology	-0.054673	-0.028198	-0.046504
word_freq_1999	-0.057312	-0.024013	-0.067015
word_freq_parts	-0.007960	-0.008922	0.032407
word_freq_pm	-0.011134	-0.019124	-0.014809
word_freq_direct	-0.036095	-0.014821	-0.047066
word_freq_cs	-0.009703	-0.015420	-0.030956
word_freq_meeting	-0.026070	-0.025177	-0.005811
word_freq_original	-0.024292	-0.002370	-0.044325
word_freq_project	-0.022116	-0.019739	-0.053464
word_freq_re	-0.037105	-0.016418	-0.050664
word_freq_edu	-0.034056	-0.023858	-0.056655
word_freq_table	-0.000953	-0.009818	0.029339
word_freq_conference	-0.017755	-0.015747	-0.026344
char_freq_%3B	-0.026505	-0.007282	-0.033213
char_freq_%28	-0.021196	-0.049837	-0.016495
char_freq_%5B	-0.033301	-0.018527	-0.033120
char_freq_%21	0.058292	-0.014461	0.108140
char_freq_%24	0.117419	-0.009605	0.087618
char_freq_%23	-0.008844	0.001946	-0.003336
capital_run_length_average	0.044491	0.002083	0.097398
capital_run_length_longest	0.061382	0.000271	0.107463
capital_run_length_total	0.089165	-0.022680	0.070114
class	0.126208	-0.030224	0.196988
		word_freq_3d word_freq_our word_freq_over \	
word_freq_make	0.013273	0.023119	0.059674
word_freq_address	-0.006923	-0.023760	-0.024840

word_freq_all	-0.020246	0.077734	0.087564
word_freq_3d	1.000000	0.003238	-0.010014
word_freq_our	0.003238	1.000000	0.054054
word_freq_over	-0.010014	0.054054	1.000000
word_freq_remove	0.019784	0.147336	0.061163
word_freq_internet	0.010268	0.029598	0.079561
word_freq_order	-0.002454	0.020823	0.117438
word_freq_mail	-0.004947	0.034495	0.013897
word_freq_receive	-0.012976	0.068382	0.053900
word_freq_will	-0.019221	0.066788	0.009264
word_freq_people	-0.013199	0.031126	0.077631
word_freq_report	0.012008	0.003445	0.009673
word_freq_addresses	0.002707	0.056177	0.173066
word_freq_free	0.007432	0.083024	0.019865
word_freq_business	0.003470	0.143443	0.064137
word_freq_email	0.019391	0.062344	0.078350
word_freq_you	-0.010834	0.098510	0.095505
word_freq_credit	-0.005381	0.031526	0.058979
word_freq_your	0.008176	0.136605	0.106833
word_freq_font	0.028102	-0.020207	0.007956
word_freq_000	0.011368	0.070037	0.211455
word_freq_money	0.035360	0.000039	0.059329
word_freq_hp	-0.015181	-0.072502	-0.084402
word_freq_hpl	-0.013708	-0.075456	-0.087271
word_freq_george	-0.010684	-0.088011	-0.069051
word_freq_650	-0.010368	-0.061501	-0.066223
word_freq_lab	-0.007798	0.032048	-0.048673
word_freq_labs	-0.010476	-0.052066	-0.048127
word_freq_telnet	-0.007529	-0.042535	-0.046383
word_freq_857	-0.006717	-0.026748	-0.036835
word_freq_data	-0.008075	-0.031998	-0.034164
word_freq_415	-0.006729	-0.026960	-0.037315
word_freq_85	-0.006122	-0.049732	-0.054315
word_freq_technology	-0.006515	-0.048844	-0.052819
word_freq_1999	-0.007761	-0.072599	-0.057465
word_freq_parts	-0.002669	0.130812	-0.017918
word_freq_pm	-0.004602	-0.042044	-0.047619
word_freq_direct	-0.007643	-0.021442	-0.029866
word_freq_cs	-0.005670	-0.047505	-0.029457
word_freq_meeting	-0.008095	0.115041	-0.054812
word_freq_original	-0.009268	-0.048879	-0.030616
word_freq_project	-0.005933	0.015234	-0.028826
word_freq_re	-0.012957	-0.042336	-0.053637
word_freq_edu	-0.009181	-0.077986	-0.033046
word_freq_table	-0.003348	-0.026900	-0.014343
word_freq_conference	-0.001924	-0.032005	-0.031693
char_freq_%3B	-0.000591	-0.032759	-0.019119

char_freq_%28	-0.012370	-0.046361	-0.008705
char_freq_%5B	-0.007148	-0.026390	-0.015133
char_freq_%21	-0.003138	0.025509	0.065043
char_freq_%24	0.010862	0.041582	0.105692
char_freq_%23	-0.000298	0.002016	0.019894
capital_run_length_average	0.005260	0.052662	-0.010278
capital_run_length_longest	0.022081	0.052290	0.090172
capital_run_length_total	0.021369	0.002492	0.082089
class	0.057371	0.241920	0.232604

	word_freq_remove	word_freq_internet	\
word_freq_make	0.007669	-0.003950	
word_freq_address	0.003918	-0.016280	
word_freq_all	0.036677	0.012003	
word_freq_3d	0.019784	0.010268	
word_freq_our	0.147336	0.029598	
word_freq_over	0.061163	0.079561	
word_freq_remove	1.000000	0.044545	
word_freq_internet	0.044545	1.000000	
word_freq_order	0.050786	0.105302	
word_freq_mail	0.056809	0.083129	
word_freq_receive	0.159578	0.128495	
word_freq_will	-0.001461	-0.002973	
word_freq_people	0.013295	0.026274	
word_freq_report	-0.022723	0.012426	
word_freq_addresses	0.042904	0.072782	
word_freq_free	0.128436	0.051115	
word_freq_business	0.187981	0.216422	
word_freq_email	0.122011	0.037738	
word_freq_you	0.111792	0.020641	
word_freq_credit	0.046134	0.109163	
word_freq_your	0.130794	0.156905	
word_freq_font	-0.002093	-0.016192	
word_freq_000	0.064795	0.089226	
word_freq_money	0.030575	0.034127	
word_freq_hp	-0.089494	-0.053038	
word_freq_hpl	-0.080330	-0.041450	
word_freq_george	-0.065893	-0.057189	
word_freq_650	-0.066947	-0.049988	
word_freq_lab	-0.048482	-0.037047	
word_freq_labs	-0.058101	-0.043405	
word_freq_telnet	-0.046280	-0.035816	
word_freq_857	-0.040538	-0.034276	
word_freq_data	-0.041372	-0.039220	
word_freq_415	-0.040910	-0.034811	
word_freq_85	-0.053202	-0.035174	
word_freq_technology	-0.053978	-0.033747	

word_freq_1999	-0.052035	-0.017466
word_freq_parts	-0.014781	-0.012119
word_freq_pm	-0.046978	-0.030392
word_freq_direct	-0.022121	-0.005988
word_freq_cs	-0.033120	-0.003884
word_freq_meeting	-0.049664	-0.043626
word_freq_original	-0.049079	-0.004542
word_freq_project	-0.034461	-0.030134
word_freq_re	-0.050811	-0.002423
word_freq_edu	-0.056166	-0.037916
word_freq_table	-0.017512	-0.006397
word_freq_conference	-0.031408	-0.021224
char_freq_%3B	-0.033089	-0.027432
char_freq_%28	-0.051885	-0.032494
char_freq_%5B	-0.027653	-0.019548
char_freq_%21	0.053706	0.031454
char_freq_%24	0.070127	0.057910
char_freq_%23	0.046612	-0.008012
capital_run_length_average	0.041565	0.011254
capital_run_length_longest	0.059677	0.037575
capital_run_length_total	-0.008344	0.040252
class	0.332117	0.206808

	word_freq_order	word_freq_mail	...	\
word_freq_make	0.106263	0.041198	...	
word_freq_address	-0.003826	0.032962	...	
word_freq_all	0.093786	0.032075	...	
word_freq_3d	-0.002454	-0.004947	...	
word_freq_our	0.020823	0.034495	...	
word_freq_over	0.117438	0.013897	...	
word_freq_remove	0.050786	0.056809	...	
word_freq_internet	0.105302	0.083129	...	
word_freq_order	1.000000	0.130624	...	
word_freq_mail	0.130624	1.000000	...	
word_freq_receive	0.137760	0.125319	...	
word_freq_will	0.030344	0.071157	...	
word_freq_people	0.034738	0.045737	...	
word_freq_report	0.066840	0.017901	...	
word_freq_addresses	0.238436	0.160543	...	
word_freq_free	0.008269	0.025601	...	
word_freq_business	0.158390	0.081363	...	
word_freq_email	0.098804	0.035977	...	
word_freq_you	0.039017	0.093509	...	
word_freq_credit	0.123217	0.030859	...	
word_freq_your	0.159112	0.098072	...	
word_freq_font	-0.019648	0.008200	...	
word_freq_000	0.126800	0.096809	...	

word_freq_money	0.099461	0.052129	...
word_freq_hp	-0.069931	-0.033534	...
word_freq_hpl	-0.049775	-0.013045	...
word_freq_george	-0.064608	-0.067817	...
word_freq_650	-0.056764	0.019356	...
word_freq_lab	-0.044840	-0.026903	...
word_freq_labs	-0.043643	0.008677	...
word_freq_telnet	-0.040158	-0.024423	...
word_freq_857	-0.033984	-0.015137	...
word_freq_data	-0.014403	-0.035366	...
word_freq_415	-0.033601	-0.014434	...
word_freq_85	-0.041847	-0.020092	...
word_freq_technology	-0.056270	-0.016955	...
word_freq_1999	-0.033244	-0.004944	...
word_freq_parts	-0.002216	-0.017950	...
word_freq_pm	-0.040844	-0.016091	...
word_freq_direct	-0.009867	0.004163	...
word_freq_cs	-0.035177	-0.025084	...
word_freq_meeting	-0.048223	-0.054467	...
word_freq_original	-0.034190	0.023200	...
word_freq_project	-0.035159	-0.026654	...
word_freq_re	-0.075558	-0.032065	...
word_freq_edu	-0.056817	-0.030326	...
word_freq_table	0.007521	-0.015546	...
word_freq_conference	-0.026017	-0.016842	...
char_freq_%3B	-0.014646	0.011945	...
char_freq_%28	-0.031003	0.003936	...
char_freq_%5B	0.013601	0.007357	...
char_freq_%21	0.043639	0.036737	...
char_freq_%24	0.149365	0.075786	...
char_freq_%23	-0.000522	0.044830	...
capital_run_length_average	0.111308	0.073677	...
capital_run_length_longest	0.189247	0.103308	...
capital_run_length_total	0.248724	0.087273	...
class	0.231551	0.138962	...
	char_freq_%3B	char_freq_%28	char_freq_%5B \
word_freq_make	-0.026505	-0.021196	-0.033301
word_freq_address	-0.007282	-0.049837	-0.018527
word_freq_all	-0.033213	-0.016495	-0.033120
word_freq_3d	-0.000591	-0.012370	-0.007148
word_freq_our	-0.032759	-0.046361	-0.026390
word_freq_over	-0.019119	-0.008705	-0.015133
word_freq_remove	-0.033089	-0.051885	-0.027653
word_freq_internet	-0.027432	-0.032494	-0.019548
word_freq_order	-0.014646	-0.031003	0.013601
word_freq_mail	0.011945	0.003936	0.007357

word_freq_receive	-0.032410	-0.055089	-0.025183
word_freq_will	-0.027711	-0.030940	-0.044966
word_freq_people	-0.023445	-0.051151	-0.028283
word_freq_report	-0.019045	-0.005804	-0.014349
word_freq_addresses	-0.018277	-0.002551	-0.003111
word_freq_free	-0.026841	-0.046578	-0.029560
word_freq_business	-0.031542	-0.035897	-0.036691
word_freq_email	-0.039519	-0.035897	-0.017439
word_freq_you	-0.044314	-0.128882	-0.063826
word_freq_credit	-0.020851	-0.021431	-0.012071
word_freq_your	-0.058660	-0.085181	-0.045469
word_freq_font	0.416608	-0.046244	-0.001137
word_freq_000	-0.027362	-0.033174	-0.000467
word_freq_money	-0.019139	-0.033113	-0.020798
word_freq_hp	0.029181	0.136979	0.039723
word_freq_hpl	0.013558	0.144771	0.064349
word_freq_george	-0.022724	-0.028748	-0.017676
word_freq_650	-0.025020	0.313835	0.031979
word_freq_lab	-0.018502	0.158593	0.006575
word_freq_labs	-0.019845	0.224192	0.004667
word_freq_telnet	-0.016280	0.233392	0.010718
word_freq_857	-0.008853	0.304679	0.013805
word_freq_data	-0.005691	0.028655	0.113105
word_freq_415	-0.009290	0.303606	0.013687
word_freq_85	-0.021592	0.200713	0.034435
word_freq_technology	-0.018947	0.245454	0.001017
word_freq_1999	0.052138	0.107674	0.073010
word_freq_parts	0.007886	-0.010430	0.001767
word_freq_pm	0.034492	0.107928	0.038474
word_freq_direct	-0.018693	0.268701	0.014065
word_freq_cs	0.053034	0.017584	0.034408
word_freq_meeting	-0.007817	-0.013082	0.011091
word_freq_original	0.015385	0.060568	0.115548
word_freq_project	-0.007257	-0.003203	-0.010733
word_freq_re	-0.024698	0.001413	0.008838
word_freq_edu	0.015382	0.014763	-0.003168
word_freq_table	0.000995	-0.003085	-0.004592
word_freq_conference	-0.002290	-0.012795	-0.006310
char_freq_%3B	1.000000	0.049124	0.009070
char_freq_%28	0.049124	1.000000	0.022316
char_freq_%5B	0.009070	0.022316	1.000000
char_freq_%21	0.020539	-0.030354	-0.031769
char_freq_%24	0.006392	0.044722	-0.026400
char_freq_%23	0.055057	0.023322	-0.006863
capital_run_length_average	0.003443	0.034365	-0.008180
capital_run_length_longest	0.040829	0.370963	-0.013994
capital_run_length_total	0.055298	0.112209	0.006016

class	-0.059630	-0.089672	-0.064709
	char_freq_%21	char_freq_%24	char_freq_%23 \
word_freq_make	0.058292	0.117419	-0.008844
word_freq_address	-0.014461	-0.009605	0.001946
word_freq_all	0.108140	0.087618	-0.003336
word_freq_3d	-0.003138	0.010862	-0.000298
word_freq_our	0.025509	0.041582	0.002016
word_freq_over	0.065043	0.105692	0.019894
word_freq_remove	0.053706	0.070127	0.046612
word_freq_internet	0.031454	0.057910	-0.008012
word_freq_order	0.043639	0.149365	-0.000522
word_freq_mail	0.036737	0.075786	0.044830
word_freq_receive	0.024992	0.070227	0.001126
word_freq_will	0.013369	0.016723	-0.030445
word_freq_people	0.040737	0.205905	-0.014195
word_freq_report	-0.008499	0.080953	0.006545
word_freq_addresses	0.018607	0.123854	-0.005446
word_freq_free	0.104261	0.049953	0.035534
word_freq_business	0.077049	0.098323	-0.000466
word_freq_email	0.039350	0.063872	0.020978
word_freq_you	0.153381	0.091470	-0.002434
word_freq_credit	0.048350	0.034948	0.007214
word_freq_your	0.084017	0.141649	-0.004355
word_freq_font	-0.004838	-0.011036	0.184428
word_freq_000	0.070103	0.310971	0.020140
word_freq_mone	0.051076	0.104691	0.000703
word_freq_hp	-0.090862	-0.086634	0.058780
word_freq_hpl	-0.078367	-0.081198	-0.020691
word_freq_george	-0.067500	-0.068728	-0.020561
word_freq_650	-0.063495	-0.061441	-0.011438
word_freq_lab	-0.042330	-0.050231	0.002076
word_freq_labs	-0.061694	-0.065475	0.082593
word_freq_telnet	-0.045273	-0.047475	0.000225
word_freq_857	-0.041529	-0.043484	-0.010735
word_freq_data	-0.048493	-0.048101	-0.009928
word_freq_415	-0.038626	-0.039844	-0.010635
word_freq_85	-0.048822	-0.048947	-0.009650
word_freq_technology	-0.060379	-0.057933	0.006452
word_freq_1999	-0.054578	-0.063895	-0.022637
word_freq_parts	-0.015126	-0.012909	-0.003627
word_freq_pm	-0.024846	-0.044513	-0.011326
word_freq_direct	-0.032509	-0.016724	-0.010661
word_freq_cs	-0.025911	-0.036610	-0.011755
word_freq_meeting	-0.038094	-0.043653	-0.003873
word_freq_original	-0.049362	-0.054698	-0.013925
word_freq_project	-0.033837	-0.036241	0.001167

word_freq_re	0.067569	-0.049367	-0.023878
word_freq_edu	-0.028845	-0.050109	-0.015040
word_freq_table	-0.017679	-0.018549	0.000308
word_freq_conference	-0.026576	-0.030751	-0.008575
char_freq_%3B	0.020539	0.006392	0.055057
char_freq_%28	-0.030354	0.044722	0.023322
char_freq_%5B	-0.031769	-0.026400	-0.006863
char_freq_%21	1.000000	0.142913	0.020924
char_freq_%24	0.142913	1.000000	0.012613
char_freq_%23	0.020924	0.012613	1.000000
capital_run_length_average	0.054308	0.079998	0.013497
capital_run_length_longest	0.077392	0.183144	0.061657
capital_run_length_total	0.036321	0.201948	0.042568
class	0.241888	0.323629	0.065067

	capital_run_length_average \
word_freq_make	0.044491
word_freq_address	0.002083
word_freq_all	0.097398
word_freq_3d	0.005260
word_freq_our	0.052662
word_freq_over	-0.010278
word_freq_remove	0.041565
word_freq_internet	0.011254
word_freq_order	0.111308
word_freq_mail	0.073677
word_freq_receive	0.029258
word_freq_will	-0.010002
word_freq_people	-0.013446
word_freq_report	0.003023
word_freq_addresses	0.017383
word_freq_free	0.015036
word_freq_business	0.038126
word_freq_email	-0.007979
word_freq_you	-0.030592
word_freq_credit	0.067140
word_freq_your	0.041066
word_freq_font	0.021497
word_freq_000	0.008372
word_freq_money	0.007681
word_freq_hp	-0.017285
word_freq_hpl	-0.024234
word_freq_george	-0.025504
word_freq_650	-0.013757
word_freq_lab	-0.014936
word_freq_labs	-0.016599
word_freq_telnet	-0.010897

word_freq_857	-0.010498
word_freq_data	-0.015509
word_freq_415	-0.002404
word_freq_85	-0.013707
word_freq_technology	-0.019185
word_freq_1999	-0.014423
word_freq_parts	-0.006012
word_freq_pm	-0.014032
word_freq_direct	-0.003945
word_freq_cs	-0.008895
word_freq_meeting	-0.017899
word_freq_original	-0.017681
word_freq_project	-0.013157
word_freq_re	-0.026979
word_freq_edu	-0.017408
word_freq_table	-0.006465
word_freq_conference	-0.008114
char_freq_%3B	0.003443
char_freq_%28	0.034365
char_freq_%5B	-0.008180
char_freq_%21	0.054308
char_freq_%24	0.079998
char_freq_%23	0.013497
capital_run_length_average	1.000000
capital_run_length_longest	0.492638
capital_run_length_total	0.162314
class	0.109999

capital_run_length_longest \	
word_freq_make	0.061382
word_freq_address	0.000271
word_freq_all	0.107463
word_freq_3d	0.022081
word_freq_our	0.052290
word_freq_over	0.090172
word_freq_remove	0.059677
word_freq_internet	0.037575
word_freq_order	0.189247
word_freq_mail	0.103308
word_freq_receive	0.086791
word_freq_will	0.021774
word_freq_people	0.041962
word_freq_report	0.060993
word_freq_addresses	0.213992
word_freq_free	0.026528
word_freq_business	0.062672
word_freq_email	0.075122

word_freq_you	0.006530
word_freq_credit	0.099463
word_freq_your	0.085321
word_freq_font	0.027775
word_freq_000	0.123036
word_freq_money	0.044870
word_freq_hp	-0.051206
word_freq_hpl	-0.051806
word_freq_george	-0.054400
word_freq_650	-0.038772
word_freq_lab	-0.034733
word_freq_labs	-0.039001
word_freq_telnet	-0.027449
word_freq_857	-0.027732
word_freq_data	-0.025919
word_freq_415	-0.024532
word_freq_85	-0.030236
word_freq_technology	-0.038100
word_freq_1999	-0.033204
word_freq_parts	-0.009487
word_freq_pm	-0.029229
word_freq_direct	-0.004835
word_freq_cs	-0.023658
word_freq_meeting	-0.034585
word_freq_original	-0.017279
word_freq_project	-0.025918
word_freq_re	-0.051858
word_freq_edu	-0.033365
word_freq_table	-0.010154
word_freq_conference	-0.016894
char_freq_%3B	0.040829
char_freq_%28	0.370963
char_freq_%5B	-0.013994
char_freq_%21	0.077392
char_freq_%24	0.183144
char_freq_%23	0.061657
capital_run_length_average	0.492638
capital_run_length_longest	1.000000
capital_run_length_total	0.475486
class	0.216097

	capital_run_length_total	class
word_freq_make	0.089165	0.126208
word_freq_address	-0.022680	-0.030224
word_freq_all	0.070114	0.196988
word_freq_3d	0.021369	0.057371
word_freq_our	0.002492	0.241920

word_freq_over	0.082089	0.232604
word_freq_remove	-0.008344	0.332117
word_freq_internet	0.040252	0.206808
word_freq_order	0.248724	0.231551
word_freq_mail	0.087273	0.138962
word_freq_receive	0.115055	0.234529
word_freq_will	0.020076	0.007741
word_freq_people	0.105150	0.132927
word_freq_report	0.169257	0.060027
word_freq_addresses	0.151626	0.195902
word_freq_free	0.003007	0.263215
word_freq_business	0.064261	0.263204
word_freq_email	0.046364	0.204208
word_freq_you	-0.007307	0.273651
word_freq_credit	0.075751	0.189761
word_freq_your	0.051797	0.383234
word_freq_font	0.103954	0.091860
word_freq_000	0.165977	0.334787
word_freq_money	0.080993	0.216111
word_freq_hp	-0.043267	-0.256723
word_freq_hpl	-0.059601	-0.232968
word_freq_george	-0.096548	-0.183404
word_freq_650	-0.067596	-0.158800
word_freq_lab	-0.056628	-0.133523
word_freq_labs	-0.064115	-0.171095
word_freq_telnet	-0.045923	-0.126912
word_freq_857	-0.046796	-0.114214
word_freq_data	0.006919	-0.119931
word_freq_415	-0.044529	-0.112754
word_freq_85	-0.045963	-0.149225
word_freq_technology	-0.045792	-0.136134
word_freq_1999	-0.003490	-0.178045
word_freq_parts	-0.013897	-0.031035
word_freq_pm	-0.049256	-0.122831
word_freq_direct	-0.028806	-0.064801
word_freq_cs	-0.026373	-0.097375
word_freq_meeting	-0.056511	-0.136615
word_freq_original	-0.036529	-0.135664
word_freq_project	-0.040661	-0.094594
word_freq_re	-0.095444	-0.140408
word_freq_edu	-0.046371	-0.146138
word_freq_table	0.005158	-0.044679
word_freq_conference	-0.010033	-0.084020
char_freq_%3B	0.055298	-0.059630
char_freq_%28	0.112209	-0.089672
char_freq_%5B	0.006016	-0.064709
char_freq_%21	0.036321	0.241888

```

char_freq_%24          0.201948  0.323629
char_freq_%23          0.042568  0.065067
capital_run_length_average 0.162314  0.109999
capital_run_length_longest 0.475486  0.216097
capital_run_length_total   1.000000  0.249164
class                  0.249164  1.000000

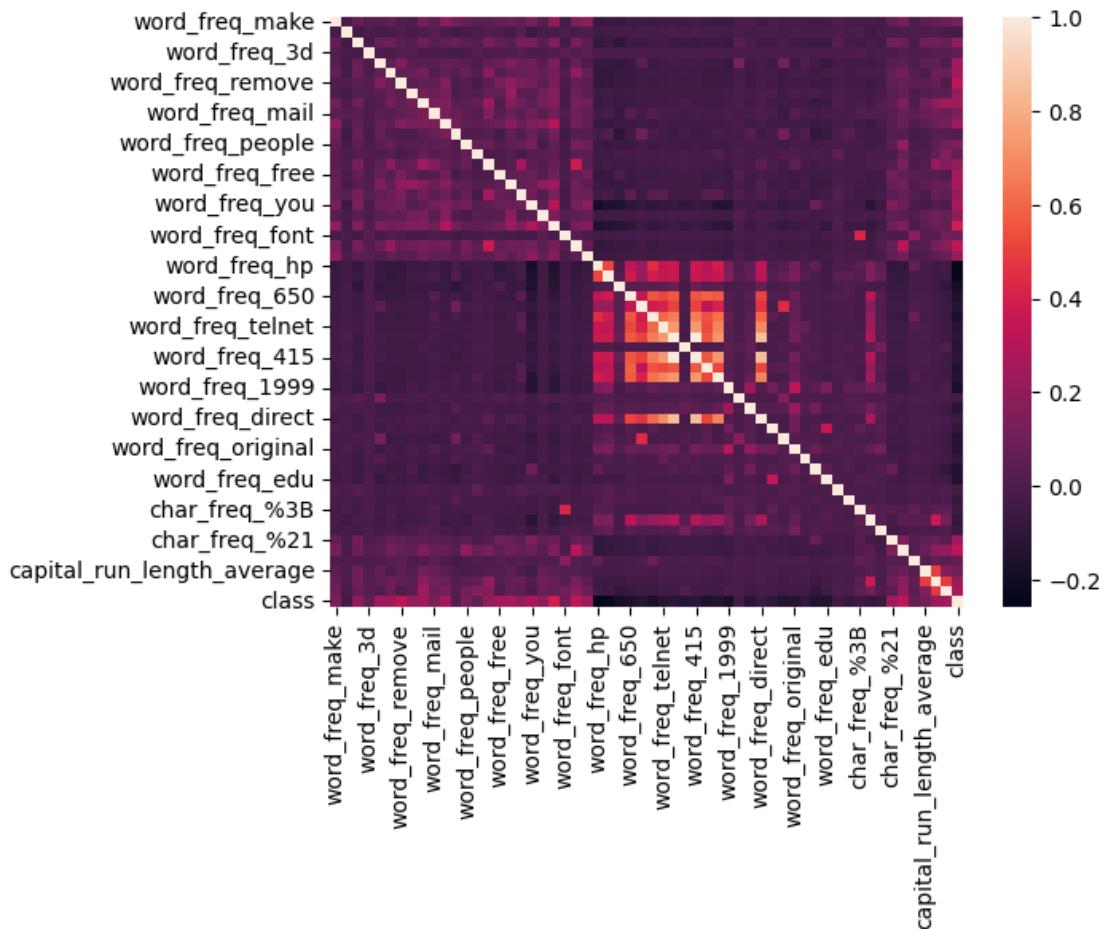
```

[58 rows x 58 columns]

8 Display data corr

[8]: sns.heatmap(data.corr())

[8]: <Axes: >



9 Show data info

```
[9]: data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4601 entries, 0 to 4600
Data columns (total 58 columns):
 #   Column           Non-Null Count  Dtype  
--- 
 0   word_freq_make    4601 non-null   float64
 1   word_freq_address 4601 non-null   float64
 2   word_freq_all     4601 non-null   float64
 3   word_freq_3d      4601 non-null   float64
 4   word_freq_our     4601 non-null   float64
 5   word_freq_over    4601 non-null   float64
 6   word_freq_remove  4601 non-null   float64
 7   word_freq_internet 4601 non-null   float64
 8   word_freq_order   4601 non-null   float64
 9   word_freq_mail    4601 non-null   float64
 10  word_freq_receive 4601 non-null   float64
 11  word_freq_will   4601 non-null   float64
 12  word_freq_people 4601 non-null   float64
 13  word_freq_report 4601 non-null   float64
 14  word_freq_addresses 4601 non-null   float64
 15  word_freq_free   4601 non-null   float64
 16  word_freq_business 4601 non-null   float64
 17  word_freq_email   4601 non-null   float64
 18  word_freq_you    4601 non-null   float64
 19  word_freq_credit 4601 non-null   float64
 20  word_freq_your   4601 non-null   float64
 21  word_freq_font   4601 non-null   float64
 22  word_freq_000    4601 non-null   float64
 23  word_freq_money  4601 non-null   float64
 24  word_freq_hp     4601 non-null   float64
 25  word_freq_hpl    4601 non-null   float64
 26  word_freq_george 4601 non-null   float64
 27  word_freq_650    4601 non-null   float64
 28  word_freq_lab    4601 non-null   float64
 29  word_freq_labs   4601 non-null   float64
 30  word_freq_telnet 4601 non-null   float64
 31  word_freq_857    4601 non-null   float64
 32  word_freq_data   4601 non-null   float64
 33  word_freq_415    4601 non-null   float64
 34  word_freq_85     4601 non-null   float64
 35  word_freq_technology 4601 non-null   float64
 36  word_freq_1999   4601 non-null   float64
 37  word_freq_parts  4601 non-null   float64
 38  word_freq_pm    4601 non-null   float64
```

```

39 word_freq_direct           4601 non-null   float64
40 word_freq_cs               4601 non-null   float64
41 word_freq_meeting          4601 non-null   float64
42 word_freq_original         4601 non-null   float64
43 word_freq_project          4601 non-null   float64
44 word_freq_re                4601 non-null   float64
45 word_freq_edu              4601 non-null   float64
46 word_freq_table             4601 non-null   float64
47 word_freq_conference       4601 non-null   float64
48 char_freq_%3B              4601 non-null   float64
49 char_freq_%28              4601 non-null   float64
50 char_freq_%5B              4601 non-null   float64
51 char_freq_%21              4601 non-null   float64
52 char_freq_%24              4601 non-null   float64
53 char_freq_%23              4601 non-null   float64
54 capital_run_length_average 4601 non-null   float64
55 capital_run_length_longest 4601 non-null   int64
56 capital_run_length_total   4601 non-null   int64
57 class                      4601 non-null   int64
dtypes: float64(55), int64(3)
memory usage: 2.0 MB

```

10 Show data contain null data

```
[10]: data.isnull().sum()
```

```

[10]: word_freq_make            0
word_freq_address              0
word_freq_all                  0
word_freq_3d                   0
word_freq_our                  0
word_freq_over                 0
word_freq_remove               0
word_freq_internet              0
word_freq_order                 0
word_freq_mail                  0
word_freq_receive               0
word_freq_will                  0
word_freq_people                 0
word_freq_report                 0
word_freq_addresses              0
word_freq_free                  0
word_freq_business                0
word_freq_email                  0
word_freq_you                   0
word_freq_credit                 0
word_freq_your                  0

```

```
word_freq_font          0
word_freq_000            0
word_freq_money          0
word_freq_hp              0
word_freq_hpl             0
word_freq_george          0
word_freq_650             0
word_freq_lab              0
word_freq_labs             0
word_freq_telnet           0
word_freq_857             0
word_freq_data             0
word_freq_415              0
word_freq_85                0
word_freq_technology        0
word_freq_1999             0
word_freq_parts             0
word_freq_pm                0
word_freq_direct             0
word_freq_cs                  0
word_freq_meeting             0
word_freq_original            0
word_freq_project             0
word_freq_re                  0
word_freq_edu                  0
word_freq_table                 0
word_freq_conference            0
char_freq_%3B                0
char_freq_%28                0
char_freq_%5B                0
char_freq_%21                0
char_freq_%24                0
char_freq_%23                0
capital_run_length_average      0
capital_run_length_longest       0
capital_run_length_total         0
class                         0
dtype: int64
```

11 Show data contain duplicate data

```
[11]: data.duplicated().sum()
```

```
[11]: 391
```

12 Show input X and output y

```
[12]: X=data.iloc[:, :-1]  
y=data.iloc[:, -1]
```

13 Show input X

```
[13]: X
```

```
[13]:    word_freq_make  word_freq_address  word_freq_all  word_freq_3d  \  
0          0.00           0.64          0.64          0.0  
1          0.21           0.28          0.50          0.0  
2          0.06           0.00          0.71          0.0  
3          0.00           0.00          0.00          0.0  
4          0.00           0.00          0.00          0.0  
...        ...           ...           ...           ...  
4596       0.31           0.00          0.62          0.0  
4597       0.00           0.00          0.00          0.0  
4598       0.30           0.00          0.30          0.0  
4599       0.96           0.00          0.00          0.0  
4600       0.00           0.00          0.65          0.0  
  
    word_freq_our  word_freq_over  word_freq_remove  word_freq_internet  \  
0          0.32           0.00          0.00          0.00  
1          0.14           0.28          0.21          0.07  
2          1.23           0.19          0.19          0.12  
3          0.63           0.00          0.31          0.63  
4          0.63           0.00          0.31          0.63  
...        ...           ...           ...           ...  
4596       0.00           0.31          0.00          0.00  
4597       0.00           0.00          0.00          0.00  
4598       0.00           0.00          0.00          0.00  
4599       0.32           0.00          0.00          0.00  
4600       0.00           0.00          0.00          0.00  
  
    word_freq_order  word_freq_mail  ...  word_freq_conference  \  
0          0.00           0.00  ...          0.0  
1          0.00           0.94  ...          0.0  
2          0.64           0.25  ...          0.0  
3          0.31           0.63  ...          0.0  
4          0.31           0.63  ...          0.0  
...        ...           ...  ...          ...  
4596       0.00           0.00  ...          0.0  
4597       0.00           0.00  ...          0.0  
4598       0.00           0.00  ...          0.0  
4599       0.00           0.00  ...          0.0
```

4600	0.00	0.00	...	0.0
	char_freq_%3B	char_freq_%28	char_freq_%5B	char_freq_%21 \
0	0.000	0.000	0.0	0.778
1	0.000	0.132	0.0	0.372
2	0.010	0.143	0.0	0.276
3	0.000	0.137	0.0	0.137
4	0.000	0.135	0.0	0.135
...
4596	0.000	0.232	0.0	0.000
4597	0.000	0.000	0.0	0.353
4598	0.102	0.718	0.0	0.000
4599	0.000	0.057	0.0	0.000
4600	0.000	0.000	0.0	0.125
	char_freq_%24	char_freq_%23	capital_run_length_average \	
0	0.000	0.000	3.756	
1	0.180	0.048	5.114	
2	0.184	0.010	9.821	
3	0.000	0.000	3.537	
4	0.000	0.000	3.537	
...	
4596	0.000	0.000	1.142	
4597	0.000	0.000	1.555	
4598	0.000	0.000	1.404	
4599	0.000	0.000	1.147	
4600	0.000	0.000	1.250	
	capital_run_length_longest	capital_run_length_total		
0	61	278		
1	101	1028		
2	485	2259		
3	40	191		
4	40	191		
...		
4596	3	88		
4597	4	14		
4598	6	118		
4599	5	78		
4600	5	40		

[4601 rows x 57 columns]

14 Show output y

```
[14]: y
```

```
[14]: 0      1
1      1
2      1
3      1
4      1
..
4596    0
4597    0
4598    0
4599    0
4600    0
Name: class, Length: 4601, dtype: int64
```

15 MinMaxScaler for Data

```
[15]: scaler = MinMaxScaler(copy=True, feature_range=(0, 1))
X = scaler.fit_transform(X)
print('X \n' , X[:10])
print('y \n' , y[:10])
```

```
X
[[0.0000000e+00 4.48179272e-02 1.25490196e-01 0.00000000e+00
 3.2000000e-02 0.00000000e+00 0.00000000e+00 0.00000000e+00
 0.00000000e+00 0.00000000e+00 0.00000000e+00 6.61840745e-02
 0.00000000e+00 0.00000000e+00 0.00000000e+00 1.60000000e-02
 0.00000000e+00 1.41914191e-01 1.02933333e-01 0.00000000e+00
 8.64086409e-02 0.00000000e+00 0.00000000e+00 0.00000000e+00
 0.00000000e+00 0.00000000e+00 0.00000000e+00 2.39546770e-02
 0.00000000e+00 0.00000000e+00 2.50204267e-03 6.00720865e-03
 1.74873737e-02]
[4.62555066e-02 1.96078431e-02 9.80392157e-02 0.00000000e+00
 1.40000000e-02 4.76190476e-02 2.88858322e-02 6.30063006e-03
 0.00000000e+00 5.17051705e-02 8.04597701e-02 8.16959669e-02
 1.17117117e-01 2.10000000e-02 3.17460317e-02 7.00000000e-03
 9.80392157e-03 3.08030803e-02 1.85066667e-01 0.00000000e+00
 1.43114311e-01 0.00000000e+00 7.88990826e-02 3.44000000e-02
 0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00]
```

0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00
 0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00
 1.01596517e-02 0.00000000e+00 0.00000000e+00 0.00000000e+00
 0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00
 0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00
 0.00000000e+00 1.35356850e-02 0.00000000e+00 1.14539073e-02
 2.99850075e-02 2.42069696e-03 3.73490695e-03 1.00120144e-02
 6.48358586e-02]
[1.32158590e-02 0.00000000e+00 1.39215686e-01 0.00000000e+00
 1.23000000e-01 3.23129252e-02 2.61348006e-02 1.08010801e-02
 1.21673004e-01 1.37513751e-02 1.45593870e-01 4.65356774e-02
 2.16216216e-02 0.00000000e+00 3.96825397e-01 3.00000000e-03
 8.40336134e-03 1.13311331e-01 7.25333333e-02 1.76017602e-02
 4.59045905e-02 0.00000000e+00 2.12844037e-01 4.80000000e-03
 0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00
 0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00
 0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00
 0.00000000e+00 0.00000000e+00 0.00000000e+00 1.26050420e-02
 0.00000000e+00 0.00000000e+00 3.36134454e-02 0.00000000e+00
 2.80112045e-03 2.72108844e-03 0.00000000e+00 0.00000000e+00
 2.28050171e-03 1.46636587e-02 0.00000000e+00 8.49806023e-03
 3.06513410e-02 5.04311866e-04 8.00817068e-03 4.84581498e-02
 1.42550505e-01]
[0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00
 6.30000000e-02 0.00000000e+00 4.26409904e-02 5.67056706e-02
 5.89353612e-02 3.46534653e-02 1.18773946e-01 3.20579111e-02
 5.58558559e-02 0.00000000e+00 0.00000000e+00 1.55000000e-02
 0.00000000e+00 0.00000000e+00 1.69600000e-01 0.00000000e+00
 2.79027903e-02 0.00000000e+00 0.00000000e+00 0.00000000e+00
 0.00000000e+00 1.40484003e-02 0.00000000e+00 4.21824004e-03
 0.00000000e+00 0.00000000e+00 2.30322288e-03 3.90468562e-03
 1.19949495e-02]
[0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00
 6.30000000e-02 0.00000000e+00 4.26409904e-02 5.67056706e-02
 5.89353612e-02 3.46534653e-02 1.18773946e-01 3.20579111e-02
 5.58558559e-02 0.00000000e+00 0.00000000e+00 1.55000000e-02
 0.00000000e+00 0.00000000e+00 1.69600000e-01 0.00000000e+00
 2.79027903e-02 0.00000000e+00 0.00000000e+00 0.00000000e+00
 0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00
 0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00
 0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00
 0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00

0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00
 0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00
 0.00000000e+00 1.38433142e-02 0.00000000e+00 4.15665989e-03
 0.00000000e+00 0.00000000e+00 2.30322288e-03 3.90468562e-03
 1.19949495e-02]
[0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00
 1.85000000e-01 0.00000000e+00 0.00000000e+00 1.66516652e-01
 0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00
 0.00000000e+00 2.28671042e-02 0.00000000e+00 0.00000000e+00
 0.00000000e+00 0.00000000e+00 1.81570586e-03 1.40168202e-03
 3.34595960e-03]
[0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00
 1.92000000e-01 0.00000000e+00 0.00000000e+00 0.00000000e+00
 0.00000000e+00 3.52035204e-02 3.67816092e-01 1.32368149e-01
 0.00000000e+00 0.00000000e+00 0.00000000e+00 4.80000000e-02
 0.00000000e+00 3.52035204e-02 2.05333333e-01 0.00000000e+00
 5.76057606e-02 0.00000000e+00 0.00000000e+00 0.00000000e+00
 0.00000000e+00 5.53732568e-03 0.00000000e+00 5.04957202e-03
 8.99550225e-03 0.00000000e+00 6.09169315e-04 3.00360433e-04
 7.00757576e-03]
[0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00
 1.88000000e-01 0.00000000e+00 0.00000000e+00 1.69216922e-01
 0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00
 0.00000000e+00 2.11238720e-02 0.00000000e+00 0.00000000e+00

```

0.0000000e+00 0.0000000e+00 1.31638675e-03 1.00120144e-03
3.03030303e-03]
[3.30396476e-02 0.0000000e+00 9.01960784e-02 0.0000000e+00
6.1000000e-02 0.0000000e+00 4.12654746e-02 0.0000000e+00
1.74904943e-01 4.18041804e-02 2.91187739e-01 9.51396070e-02
0.0000000e+00 0.0000000e+00 0.0000000e+00 0.0000000e+00
0.0000000e+00 1.65016502e-02 6.56000000e-02 1.94169417e-01
1.80018002e-01 0.0000000e+00 0.0000000e+00 1.2000000e-02
0.0000000e+00 0.0000000e+00 0.0000000e+00 0.0000000e+00
0.0000000e+00 0.0000000e+00 0.0000000e+00 0.0000000e+00
8.25082508e-03 0.0000000e+00 0.0000000e+00 0.0000000e+00
0.0000000e+00 0.0000000e+00 0.0000000e+00 0.0000000e+00
0.0000000e+00 0.0000000e+00 8.40336134e-02 0.0000000e+00
0.0000000e+00 0.0000000e+00 0.0000000e+00 0.0000000e+00
0.0000000e+00 2.77891715e-02 0.0000000e+00 5.57300326e-03
3.38164251e-02 1.10948611e-03 7.93826600e-03 4.44533440e-02
7.92929293e-02]
[1.32158590e-02 8.40336134e-03 1.50980392e-01 0.0000000e+00
1.9000000e-02 5.44217687e-02 5.22696011e-02 0.0000000e+00
1.14068441e-02 0.0000000e+00 0.0000000e+00 6.61840745e-02
4.50450450e-02 0.0000000e+00 2.72108844e-02 0.0000000e+00
0.0000000e+00 1.32013201e-02 8.90666667e-02 3.30033003e-03
6.39063906e-02 0.0000000e+00 3.48623853e-02 0.0000000e+00
0.0000000e+00 0.0000000e+00 0.0000000e+00 0.0000000e+00
0.0000000e+00 0.0000000e+00 0.0000000e+00 0.0000000e+00
0.0000000e+00 0.0000000e+00 0.0000000e+00 0.0000000e+00
0.0000000e+00 0.0000000e+00 0.0000000e+00 3.0000000e-03
0.0000000e+00 0.0000000e+00 0.0000000e+00 0.0000000e+00
9.12200684e-03 3.07629204e-03 0.0000000e+00 7.51277788e-03
1.34932534e-02 0.0000000e+00 6.61824784e-04 4.20504606e-03
4.72222222e-02]]
y
0      1
1      1
2      1
3      1
4      1
5      1
6      1
7      1
8      1
9      1
Name: class, dtype: int64

```

16 Splitting data

```
[16]: X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.33,random_state=44, shuffle =True)
print('X_train shape is ' , X_train.shape)
print('X_test shape is ' , X_test.shape)
print('y_train shape is ' , y_train.shape)
print('y_test shape is ' , y_test.shape)
```

```
X_train shape is (3082, 57)
X_test shape is (1519, 57)
y_train shape is (3082,)
y_test shape is (1519,)
```

17 Applying LogisticRegression Model

```
[17]: LogisticRegressionModel = LogisticRegression(penalty='l2',solver='sag',C=1.
    ↪0,random_state=33)
LogisticRegressionModel.fit(X_train, y_train)
```

```
[17]: LogisticRegression(random_state=33, solver='sag')
```

18 SVM Model

```
[18]: svm_model = svm.SVC(kernel='linear')
svm_model.fit(X_train, y_train)
```

```
[18]: SVC(kernel='linear')
```

19 Calculating Details

```
[19]: print('LogisticRegressionModel Train Score is : ' , LogisticRegressionModel.
    ↪score(X_train, y_train))
print('LogisticRegressionModel Test Score is : ' , LogisticRegressionModel.
    ↪score(X_test, y_test))
print('LogisticRegressionModel Classes are : ' , LogisticRegressionModel.
    ↪classes_)
print('LogisticRegressionModel No. of iteratios is : ' , ↪
    LogisticRegressionModel.n_iter_)
```

```
LogisticRegressionModel Train Score is : 0.8822193380921479
LogisticRegressionModel Test Score is : 0.8933508887425938
LogisticRegressionModel Classes are : [0 1]
LogisticRegressionModel No. of iteratios is : [19]
```

```
[20]: MLPClassifierModel = MLPClassifier(activation='relu', # can be also identity ,  

    ↪logistic , relu  

    solver='lbfgs', # can be also sgd , adam  

    learning_rate='constant', # can be also  

    ↪in uscaling , adaptive  

    early_stopping= False,  

    alpha=0.0001 ,hidden_layer_sizes=(100,  

    ↪10),random_state=33)  

MLPClassifierModel.fit(X_train, y_train)
```

```
C:\Users\ashwi\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.10_qbz5n  

2kfra8p0\LocalCache\local-packages\Python310\site-  

packages\sklearn\neural_network\_multilayer_perceptron.py:546:  

ConvergenceWarning: lbfgs failed to converge (status=1):  

STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.
```

Increase the number of iterations (max_iter) or scale the data as shown in:
<https://scikit-learn.org/stable/modules/preprocessing.html>
self.n_iter_ = _check_optimize_result("lbfgs", opt_res, self.max_iter)

```
[20]: MLPClassifier(hidden_layer_sizes=(100, 10), random_state=33, solver='lbfgs')
```

20 Calculating Details

```
[21]: print('MLPClassifierModel Train Score is : ' , MLPClassifierModel.  

    ↪score(X_train, y_train))  

print('MLPClassifierModel Test Score is : ' , MLPClassifierModel.score(X_test,  

    ↪y_test))  

print('MLPClassifierModel loss is : ' , MLPClassifierModel.loss_)  

print('MLPClassifierModel No. of iterations is : ' , MLPClassifierModel.n_iter_)  

print('MLPClassifierModel No. of layers is : ' , MLPClassifierModel.n_layers_)  

print('MLPClassifierModel last activation is : ' , MLPClassifierModel.  

    ↪out_activation_)
```

```
MLPClassifierModel Train Score is : 0.95425048669695  

MLPClassifierModel Test Score is : 0.945358788676761  

MLPClassifierModel loss is : 0.13134318136698722  

MLPClassifierModel No. of iterations is : 200  

MLPClassifierModel No. of layers is : 4  

MLPClassifierModel last activation is : logistic
```

21 Applying RandomForestClassifier Model

```
[22]: RandomForestClassifierModel = RandomForestClassifier(criterion =  
    ↪'gini',n_estimators=100,max_depth=20,random_state=33) #criterion can be also  
    ↪: entropy  
RandomForestClassifierModel.fit(X_train, y_train)
```

```
[22]: RandomForestClassifier(max_depth=20, random_state=33)
```

22 Calculating Details

```
[23]: print('RandomForestClassifier Model Train Score is : ',  
    ↪RandomForestClassifierModel.score(X_train, y_train))  
print('RandomForestClassifier Model Test Score is : ',  
    ↪RandomForestClassifierModel.score(X_test, y_test))  
print('RandomForestClassifier Model features importances are : ',  
    ↪RandomForestClassifierModel.feature_importances_)
```

```
RandomForestClassifier Model Train Score is : 0.9935107073329007  
RandomForestClassifier Model Test Score is : 0.9611586570111915  
RandomForestClassifier Model features importances are : [0.00316473 0.00619621  
0.00957861 0.00101638 0.03838965 0.00700016  
0.08086117 0.0159258 0.00388911 0.00738561 0.01095491 0.01146843  
0.00352738 0.00208826 0.0008272 0.07380709 0.01315063 0.00887973  
0.02773934 0.00525226 0.05326055 0.00195407 0.02750044 0.03211117  
0.05073361 0.01783291 0.0189023 0.00423941 0.00189117 0.0039426  
0.00180011 0.00095643 0.00210106 0.00065366 0.00317525 0.00313297  
0.01528722 0.00027693 0.00324306 0.00122623 0.00098861 0.00675916  
0.0018326 0.00281864 0.00960934 0.01883362 0.00027381 0.00105319  
0.00691406 0.0128784 0.00266575 0.09993113 0.09867403 0.00372857  
0.06208269 0.0572962 0.03833638]
```

23 Applying DecisionTreeClassifier Model

```
[24]: DecisionTreeClassifierModel =  
    ↪DecisionTreeClassifier(criterion='gini',max_depth=20,random_state=33)  
    ↪#criterion can be entropy  
DecisionTreeClassifierModel.fit(X_train, y_train)
```

```
[24]: DecisionTreeClassifier(max_depth=20, random_state=33)
```

24 Calculating Details

```
[25]: print('DecisionTreeClassifierModel Train Score is : ',  
      ↪DecisionTreeClassifierModel.score(X_train, y_train))  
print('DecisionTreeClassifierModel Test Score is : ',  
      ↪DecisionTreeClassifierModel.score(X_test, y_test))  
print('DecisionTreeClassifierModel Classes are : ',  
      ↪DecisionTreeClassifierModel.classes_)  
print('DecisionTreeClassifierModel feature importances are : ',  
      ↪DecisionTreeClassifierModel.feature_importances_)
```

```
DecisionTreeClassifierModel Train Score is : 0.9935107073329007  
DecisionTreeClassifierModel Test Score is : 0.9117840684660962  
DecisionTreeClassifierModel Classes are : [0 1]  
DecisionTreeClassifierModel feature importances are : [2.17307506e-03  
2.01897104e-03 3.55054373e-03 0.00000000e+00  
2.23174595e-02 1.79968416e-03 1.53150501e-01 8.07170832e-03  
2.14857669e-03 3.88566332e-03 5.72015974e-03 5.39027479e-03  
3.31000774e-03 1.99353559e-03 0.00000000e+00 2.77571297e-02  
1.32868868e-03 7.71873435e-03 1.93365699e-02 1.98166690e-03  
1.90999181e-02 2.80982131e-03 2.14372882e-02 6.26441059e-03  
5.29547580e-02 2.98919742e-03 1.36447641e-02 5.42661616e-03  
1.34165467e-03 0.00000000e+00 6.49257703e-04 0.00000000e+00  
6.89884684e-04 0.00000000e+00 5.52321466e-03 7.26554993e-03  
9.20211758e-04 0.00000000e+00 1.08639996e-02 0.00000000e+00  
0.00000000e+00 1.61254483e-02 1.95557190e-03 1.39117375e-03  
1.09234873e-02 1.13578104e-02 0.00000000e+00 0.00000000e+00  
1.26697397e-03 7.74145996e-03 2.48425997e-03 9.84607125e-02  
3.39814539e-01 7.20258733e-05 4.53599543e-02 1.87664934e-02  
1.87465924e-02]
```

25 Applying KNeighborsClassifier Model

```
[26]: KNNClassifierModel = KNeighborsClassifier(n_neighbors=15,weights ='distance', #  
      ↪it can be distance  
                                         algorithm='auto') # it can be  
      ↪ball_tree, kd_tree, brute  
KNNClassifierModel.fit(X_train, y_train)
```

```
[26]: KNeighborsClassifier(n_neighbors=15, weights='distance')
```

26 Calculating Details

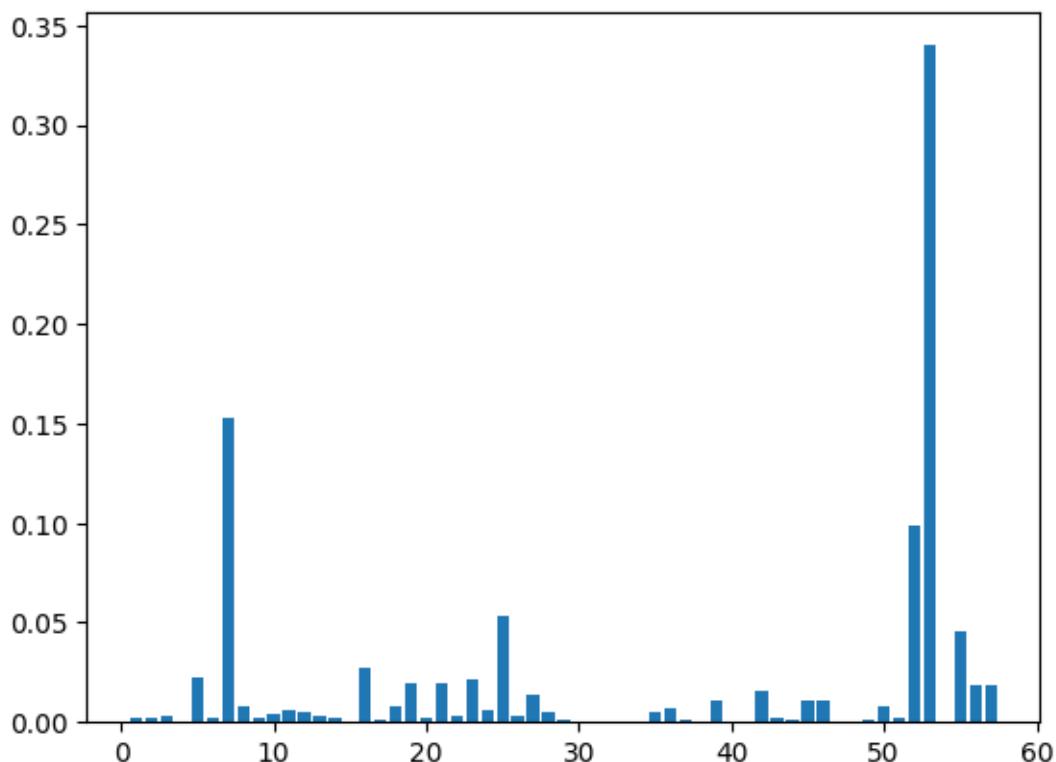
```
[27]: print('KNNClassifierModel Train Score is : ', KNNClassifierModel.  
         score(X_train, y_train))  
print('KNNClassifierModel Test Score is : ', KNNClassifierModel.score(X_test,  
                     y_test))
```

```
KNNClassifierModel Train Score is :  0.9993510707332901  
KNNClassifierModel Test Score is :  0.9144173798551679
```

27 Display DecisionTreeClassifierModel.feature_importances_

```
[28]: x_bar=list(range(1,58))  
plt.bar(x_bar,DecisionTreeClassifierModel.feature_importances_)
```

```
[28]: <BarContainer object of 57 artists>
```



28 Calculating Prediction

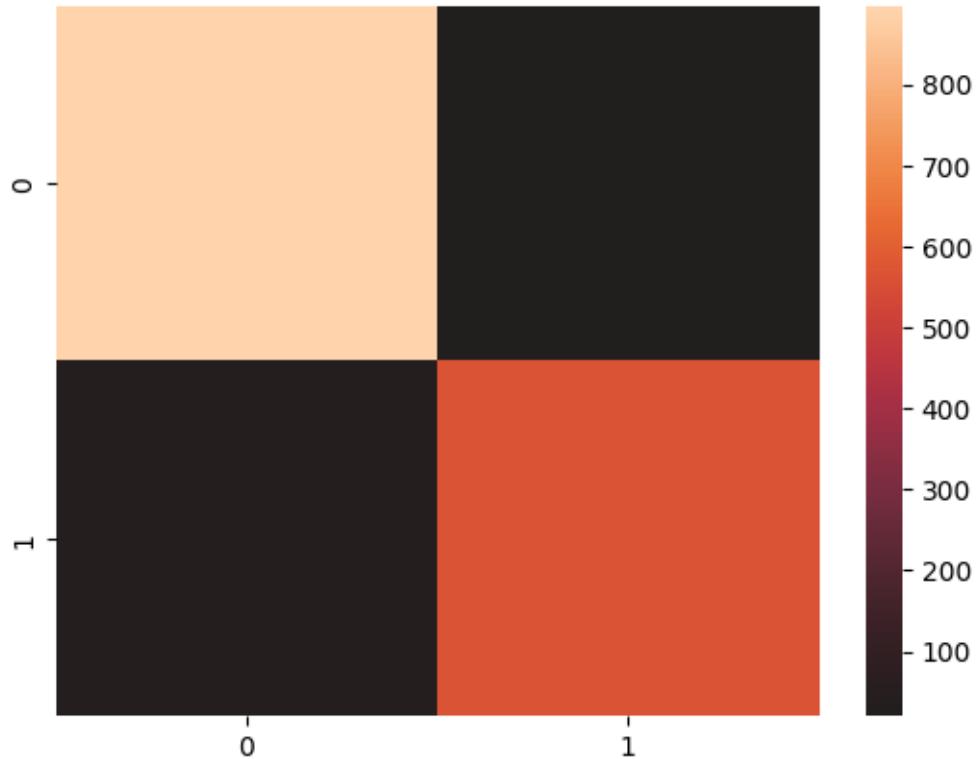
```
[29]: y_pred = RandomForestClassifierModel.predict(X_test)
y_pred_prob = RandomForestClassifierModel.predict_proba(X_test)
print('Predicted Value for RandomForestClassifierModel is : ', y_pred[:10])
print('Prediction Probabilities Value for RandomForestClassifierModel is : ', y_pred_prob[:10])
```

```
Predicted Value for RandomForestClassifierModel is : [0 1 0 0 1 0 0 0 1 0]
Prediction Probabilities Value for RandomForestClassifierModel is : [[0.952091
0.047909
[0.18      0.82      ]
[0.75585249 0.24414751]
[0.97780429 0.02219571]
[0.02      0.98      ]
[0.94738326 0.05261674]
[0.59907764 0.40092236]
[0.81903022 0.18096978]
[0.11168115 0.88831885]
[0.92      0.08      ]]
```

29 Calculating Confusion Matrix

```
[30]: CM = confusion_matrix(y_test, y_pred)
print('Confusion Matrix is : \n', CM)
sns.heatmap(CM, center = True)
plt.show()
```

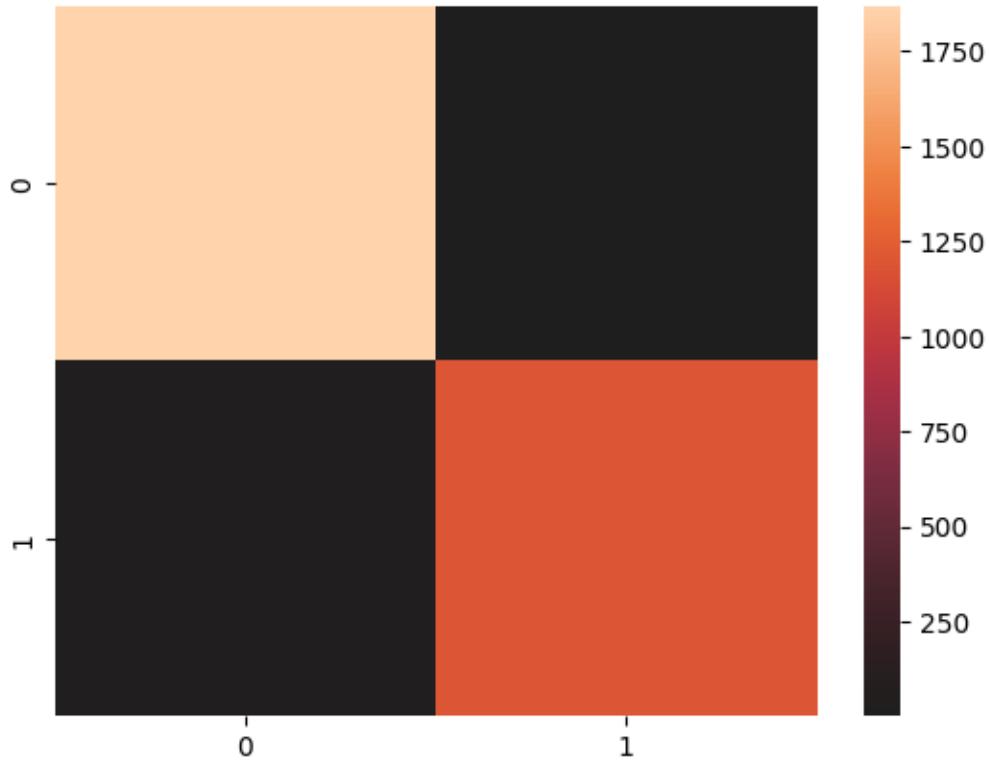
```
Confusion Matrix is :
[[897  20]
 [ 39 563]]
```



30 Calculating Confusion Matrix

```
[31]: CM = confusion_matrix(y_train,RandomForestClassifierModel.predict(X_train))
print('Confusion Matrix is : \n', CM)
sns.heatmap(CM, center = True)
plt.show()
```

```
Confusion Matrix is :
[[1870    1]
 [ 19 1192]]
```



31 Calculating Accuracy Score : $((\text{TP} + \text{TN}) / \text{float}(\text{TP} + \text{TN} + \text{FP} + \text{FN}))$

```
[32]: AccScore = accuracy_score(y_test, y_pred, normalize=True)
print('Accuracy Score is : ', AccScore)
```

Accuracy Score is : 0.9611586570111915

32 Calculating F1 Score : $2 (\text{precision} \cdot \text{recall}) / (\text{precision} + \text{recall})$

```
[33]: F1Score = f1_score(y_test, y_pred, average='micro')
print('F1 Score is : ', F1Score)
```

F1 Score is : 0.9611586570111915

33 Calculating Recall Score : (Sensitivity) ($TP / \text{float}(TP + FN)$) 1 / 1+2

```
[34]: RecallScore = recall_score(y_test, y_pred, average='micro') #it can be :  
      ↪binary,macro,weighted,samples  
print('Recall Score is : ', RecallScore)
```

Recall Score is : 0.9611586570111915

34 Calculating Precision Score : (Specificity) #($TP / \text{float}(TP + FP)$)

```
[35]: PrecisionScore = precision_score(y_test, y_pred, average='micro')  
print('Precision Score is : ', PrecisionScore)
```

Precision Score is : 0.9611586570111915

35 Calculating Precision recall Curve

```
[36]: PrecisionValue, RecallValue, ThresholdsValue =  
      ↪precision_recall_curve(y_test,y_pred)  
print('Precision Value is : ', PrecisionValue)  
print('Recall Value is : ', RecallValue)  
print('Thresholds Value is : ', ThresholdsValue)
```

Precision Value is : [0.39631336 0.96569468 1.]
Recall Value is : [1. 0.93521595 0.]
Thresholds Value is : [0 1]

36 Calculating classification Report

```
[37]: ClassificationReport = classification_report(y_test,y_pred)  
print('Classification Report is : ', ClassificationReport )
```

		precision	recall	f1-score
Classification Report is :	support			
0	0.96	0.98	0.97	917
1	0.97	0.94	0.95	602
accuracy			0.96	1519
macro avg	0.96	0.96	0.96	1519
weighted avg	0.96	0.96	0.96	1519

37 Calculating ROC AUC Score:

```
[38]: ROCAUCScore = roc_auc_score(y_test,y_pred, average='micro') #it can be :  
      ↪macro, weighted, samples  
print('ROCAUC Score : ', ROCAUCScore)
```

ROCAUC Score : 0.9567028480129847

38 Model training and prediction using SVM

```
[39]: y_pred = svm_model.predict(X_test)
```

```
[40]: accuracy = accuracy_score(y_test, y_pred)  
print("The reported accuracy:", accuracy, " (approximately ",accuracy * 100  
     ↪, "%)")
```

The reported accuracy: 0.9091507570770243 (approximately 90.91507570770243 %)

```
[41]: print(classification_report(y_test, y_pred))
```

	precision	recall	f1-score	support
0	0.90	0.96	0.93	917
1	0.93	0.83	0.88	602
accuracy			0.91	1519
macro avg	0.91	0.90	0.90	1519
weighted avg	0.91	0.91	0.91	1519

```
[42]: import joblib  
  
# Save the model to a file  
joblib.dump(svm_model, 'svm_model.pkl')
```

```
[42]: ['svm_model.pkl']
```

39 y_pred to csv

```
[43]: submision=pd.DataFrame(y_pred,columns=['prediction'])  
submision.to_csv('C:/Users/ashwi/Downloads/ML Lab/A4_MLLab_AshwinRavi/  
    ↪prediction.csv')  
submision
```

```
[43]: prediction
0          0
1          1
2          0
3          0
4          1
...
...      ...
1514        1
1515        1
1516        0
1517        0
1518        1

[1519 rows x 1 columns]
```

40 Inference and Learning Outcomes

1. Understand the working of SVM algorithm.
2. Trained a ML model using SVM alog using sklearn library.
3. Performed all ML steps such as pre-processing, encoding, data splitting, and model training etc