Assignment-6,Report

R.Chidaksh(200010046), R.Sai Krupakar(200010047)

March 16, 2022

1 Problem Statement :

Spam email classification using Support Vector Machine: In this assignment you will use a SVM to classify emails into spam or non-spam categories. And report the classification accuracy for various SVM parameters and kernel functions

2 Dataset Description :

An email is represented by various features like frequency of occurrences of certain keywords, length of capitalized words etc. A data set containing about 4601 instances are available in this link (data folder): Link The data format is also described in the above link. You have to randomly pick 70% as training data and the remaining as test data.

3 Libraries Used :

1.Numpy : NumPy is a general-purpose array-processing package. It provides a high-performance multidimensional array object, and tools for working with these arrays.

2.Pandas : Used for reading data in the files and storing in data frames.

3.Matplotlib : Matplotlib is a low level graph plotting library in python that serves as a visualization utility.

4.Seaborn : Seaborn is a library for making statistical graphics in Python. It builds on top of matplotlib and integrates closely with pandas data structures. Seaborn helps you explore and understand your data.

5.Scikit Learn : It provides a selection of efficient tools for machine learning and statistical modelling including classification, regression, clustering and dimensionality reduction via a consistence interface in Python. This library, which is largely written in Python, is built upon NumPy, SciPy and Matplotlib.

4 Methodology: Details of the SVM package used :

This interface makes implementing SVM's very quick and simple. It also facilitates probabilistic classification by using the kernel trick. It provides the most common kernels like linear, RBF, sigmoid, and polynomial.

- **svm.SVC()** : SVM model is initialized by this command.

- **clf.fit()** : We use this command for fitting the data into the model. We fit the train data into the model by using this command.

- **clf.predict()** : This command is helpful for predicting the output from the test input data.

- **f1 _score()** : This command is used to calculate the score or accuracy of the model.

5 Experimental Results :

5.1 Linear :

Linear Kernel is used when the data is Linearly separable, that is, it can be separated using a single Line. It is one of the most common kernels to be used. It is mostly used when there are a Large number of Features in a particular Data Set.

C values Vs **Accuracy of test data plot**:

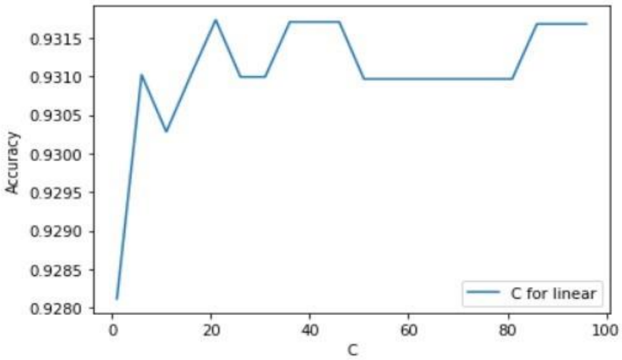

Figure 1: Linear

5.2 Quadratic :

The polynomial kernel with degree 2 is called Quadratic kernel.In machine learning, the polynomial kernel is a kernel function commonly used with support vector machines (SVMs) and other kernelized models, that represents the similarity of vectors (training samples) in a feature space over polynomials of the original variables, allowing learning of non-linear models.

<u>C values</u> Vs **<u>Accuracy of test data plot</u>**:

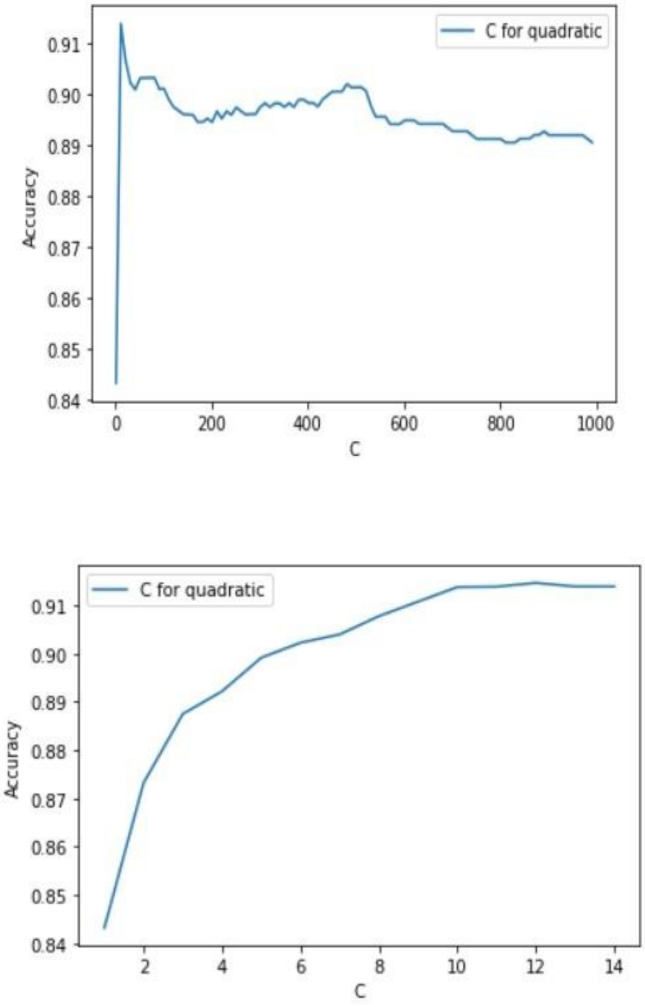

Figure 2,3: Quadratic

5.3 <u>RBF</u> :

RBF is the default kernel used within the sklearn's SVM classification algorithm. In machine learning, the radial basis function kernel, or RBF kernel, is a popular kernel function used in various kernelized learning algorithms.

C values Vs **Accuracy of test data plot**:

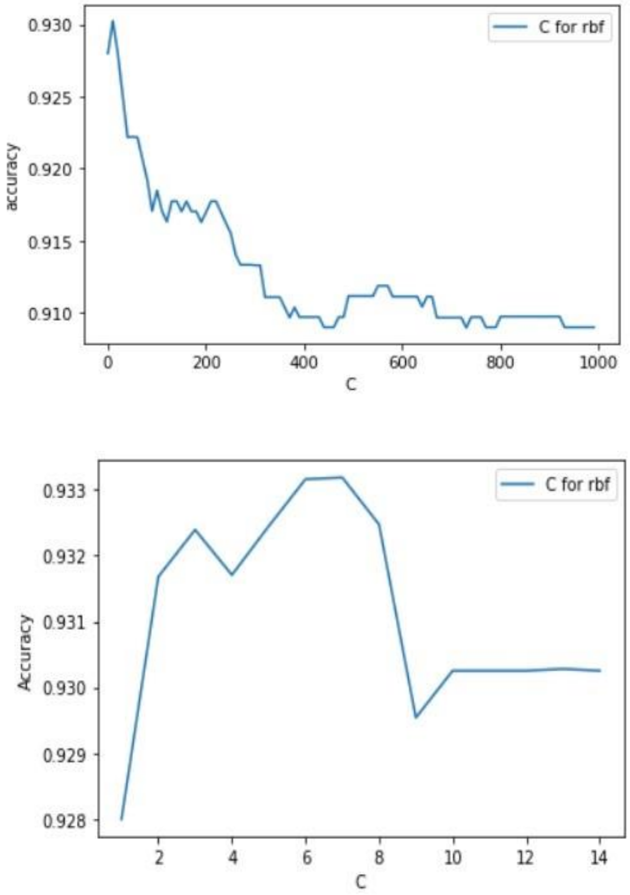

Figure 4, 5: RBF

6 Accuracy Tables of Train and Test Data :

Accuracy for different C values for the Train data

C values	Linear	Quadratic	RBF
0.005	0.918380780895517	0.4774190004054143	0.45265176095805554
0.05	0.9324333894658831	0.6381436978505738	0.9008128979485592
0.5	0.9374877994182427	0.80655706779999	0.9388835632482969
1	0.9374966172422412	0.8478052221040676	0.9489620850971479
5	0.9371812312381913	0.9382519360189834	0.9657771462290136
50	0.9402655643270055	0.9700654187338599	0.9844573184534755
500	0.9393190794258798	0.9869386718416041	0.9925384764639602
5000	0.9393190794258798	0.9937837855607204	0.996270782958689

Table 1: Accuracy values of Train dataset

Accuracy for different C values for the Test data

C values	Linear	Quadratic	RBF
0.005	0.9133100837271381	0.477606263802384	0.4681088312557028
0.05	0.9266619046732455	0.6320174553752312	0.8928874914756523
0.5	0.9281141441846663	0.8090848985003566	0.9226626902866654
1	0.9281141441846663	0.8432369310527914	0.9279999087976117
5	0.931018623207508	0.8991332296032617	0.932444546428645
50	0.9309645110270706	0.9024035389484224	0.9222131762877461
500	0.931677599351608	0.9013045094737268	0.9111809722044802
5000	0.931677599351608	0.8833972858389348	0.904608702745661

Table 2: Accuracy values of Testing dataset

7 Best C values for various kernels in the range 1 to 100 :

Best Value of C for maximum accuracy in rbf kernel is: 7
Best Accuracy achieved for rbf kernel: 0.9331840078381469

Best Value of C for maximum accuracy in linear kernel is: 21
Best Accuracy achieved for linear kernel: 0.931731486269411

Best Value of C for maximum accuracy in quadratic kernel is: 12
Best Accuracy achieved for quadratic kernel: 0.9146181157451513

8 Conclusions :

From the above results we can say that the RBF kernel function gives us the best results compared to other two kernel functions.

(i)**Linear**(in blue color): The accuracy value for the Test set is more for the value C=500.
Accuracy value for the Test set = 93.16%
Accuracy value for the Train set = 93.93%
(ii)**Quadratic**(in green color): The accuracy value for the Test set is more for the value C=50.
Accuracy value for the Test set = 90.24%
Accuracy value for the Train set = 97.00%
(iii)**RBF**(in red color): The accuracy value for the Test set is more for the value C=5.
Accuracy value for the Test set = 93.24%
Accuracy value for the Train set = 96.57%

In the above table we have calculated different accuracy values for the test data for three different kernel functions and we have observed that the accuracy for the Test set is maximum in RBF kernel function (i.e around 93.4%) compared to other two kernels. Also the training data is linear separable because for kernel other than linear model the f1 score or accuracy was very poor.