Assignment 6 lab report

Om Patil (200010036)
Hrishikesh Pable (200010037)

March 2022

1 Library Used

We used the scikit learn (sklearn) library for this assignment. The svm module of this library was imported and used.

2 Methodology

Initially the samples are parsed from the input data file; 70% of the sample are randomly selected to form the training set and the remainder form the testing set. This is done by first shuffling all the samples and then selecting the first 70% samples for the training set.

We then create a SVM model object with the specified C and kernel. This is done by calling the command `svm.SVC(kernel=kernel, degree=2, C=c)` which returns a SVM object with the kernels and C as passed. The optional parameter `degree` is passed as 2 so that when the kernel is passed as polynomial, the degree is taken as two, effectively creating a quadratic kernel. Note that we also change kernel passed as "quad" to "polynomial" to facilitate the same.

This is then trained on the training set samples using the method `fit(x,y)` on the SVM object where x is an array of arrays of the sample's features and y is an array of the corresponding labels.

Following this, samples in the testing set are used to find the accuracy of the model. An unseen sample can be classified using the model by the object's `predict(x)` method where x is an array of the features of the sample which is to be classified. This classification is compared with the label from the dataset, doing this on all samples in the testing set allows us to find an test accuracy for the trained model. We can do the same on the training set to obtain the training accuracy

We do this 3 times, once for each of the kernel types i.e. linear, quadratic and rbf with the passed optional C parameter (if not passed, C=1 is assumed). Following this, the accuracy values are printed out.

3 Experimental Details

We implemented SVM on the emails-dataset provided to us, using different kernels and different values of the regularisation constant C. The kernels that we used were linear, quadratic and rbf kernels. The value of C was varied from C=1 to C=30,000 . The quadratic kernel showed steady increase in test accuracy (with one case of negligible decrease) as the value of c was increased from 1 to 30,000. Its test accuracy increased from 67.7045 % to 86.0246 %. The test accuracy of rbf kernel also increased, which was 70.7458 % at C=1, and became 93.5553 % at C=30,000. However, in case of linear kernel, the test accuracy was nearly the same, with little bit of deviations due to random selection of training and testing data. Its test accuracy was 92.3968 % for C=1 and 91.7451 % for C=30000.

Among all the kernels used, the rbf kernel was most sensitive to changes in values of c, followed by quadratic kernel, and the linear kernel (which was very slightly affected for even large variations in C).The best test accuracy achieved was for the rbf kernel at C=30000, i.e. 93.5553 %.

Following is a detailed overview of the experiment:

C	Linear	Quadratic	RBF
1	92.3968	67.7046	70.7458
28	90.3693	67.9218	75.0905
50	92.5416	67.1252	77.2628
1000	92.3968	76.2490	89.7175
5000	91.9623	82.476	91.6003
15000	91.3831	85.5177	93.2657
30000	91.7451	86.0246	93.5553

Table 1: Test accuracy across various kernels and values of C

Kernel	Best C	Test Accuracy	Training Accuracy
linear	50	92.0348	91.6149
quad	30000	87.6901	85.7764
rbf	30000	92.7589	94.0062

Table 2: Test and training accuracy across kernels and their respective best value of C