**CS312 - Artificial Intelligence Lab**

**Assignment 6**

Group 10

Rudraksh (190030038), Sumedhsingh (190030042)

Likhilesh (190030024)

**Introduction**

In this assignment we need to classify Spam emails using a Support Vector Machine, we will be using an SVM to classify emails into spam or non-spam categories. And report the classification accuracy for various SVM parameters and kernel functions.

**Libraries Used**

1) Scikit-Learn Package

Scikit-learn is a free machine learning package provided by Python. It includes a various number of things such as support vector machines, random forests, gradient boosting, k-means, and DBSCAN, among other classification, regression, and clustering techniques, and is designed to work with Python's numerical and scientific libraries.

2) Pandas Package

Pandas is data manipulation and analysis software package which is available with python language. It includes data structures and methods for manipulating numerical tables and time series, in particular.

**Details of SVM package used**

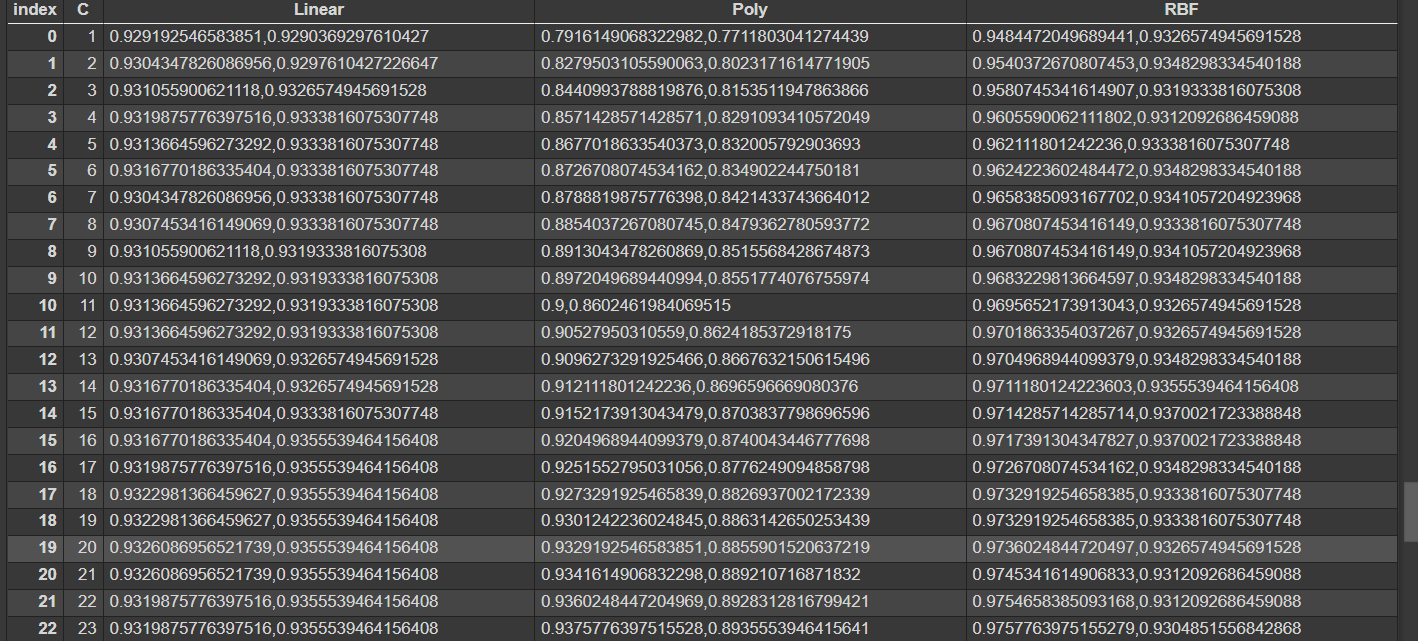
We have used various SVM modules for the implementation of our code, further details are mentioned below:

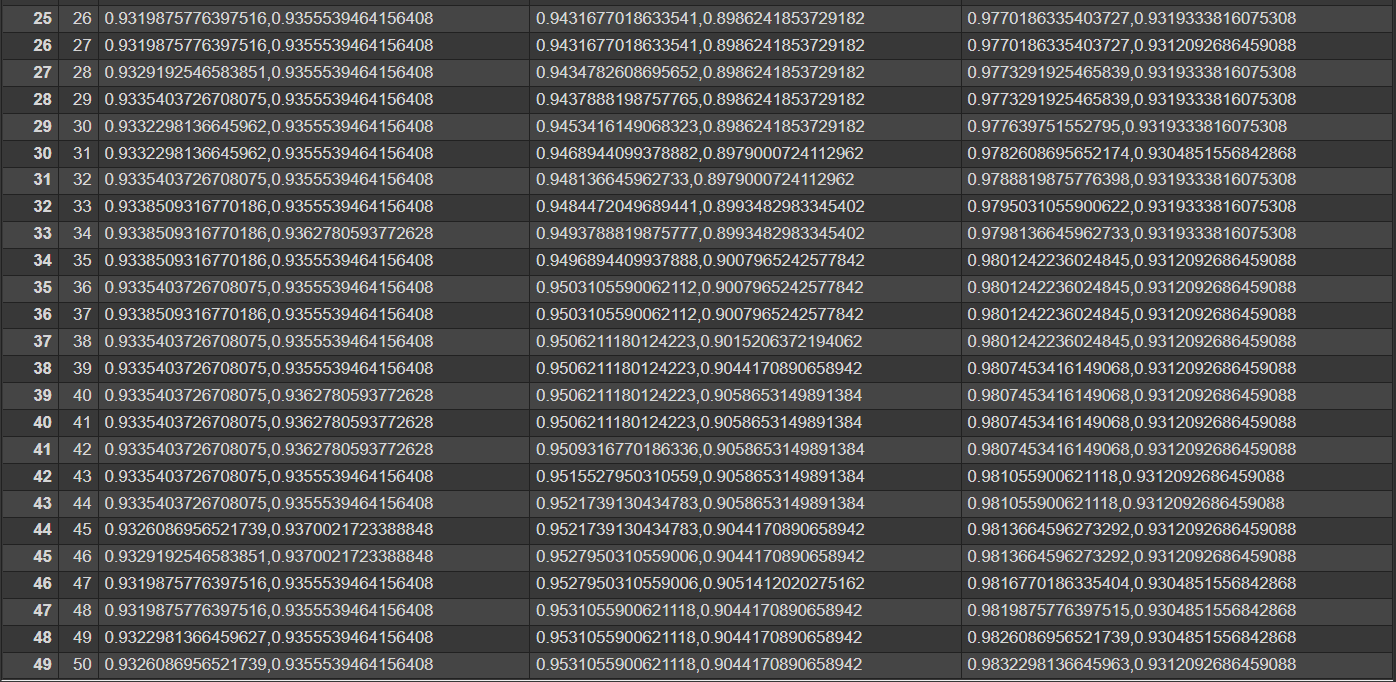
Some

* SVC(): We have used this function to initialize our model. We have also passed the hyperparameters of our algorithm this function.
* .fit(X\_train, y\_train): This function takes two arguments i.e. input dataset and output dataset, and then applies the SVM algorithm to them further creating a model that fits the given dataset.
* .score(X\_test, y\_test): This function takes two arguments i.e. test input and expected test output. It applies our generated model to the test input and compares the generated output with the given expected output. It also returns the accuracy of our model.

**Experimental Results:**

The training and testing accuracies observed for different values of C has been shown in the table below:





For different kernels, we were able to observe maximum accuracies for the following value of C

● Linear: 46

● Quadratic/Polynomial:44

● RBF: 16

**Conclusion**

Training data is linear separable because kernel other than linear model performs poorly with a low value of C. For very tiny values of C, we should get misclassified examples, often even if our training data is linearly separable.