SVM-Based Binary Classifier (Challenge 5)

-Venkata Koustubha Krishna Pydimarri

***Abstract*- The primary objective addressed in this challenge is using SVM for classifying a two-dimensional dataset. In this challenge, different types of SVMs like linear, RBF and polynomial are used to achieve the objective of binary classification for two-dimensional dataset. After the required analysis on my dataset, it can be concluded that RBF kernel SVM gives the best classification accuracy when compared to other SVM approaches. After knowing the best SVM, fine-tuning of necessary parameters is done in order to increase the prediction accuracy. This fine-tuning is carried out for training dataset and then done for the test dataset to give the predictions.**

Keywords— Radial Basis Function(RBF), SVM, Polynomial, Decision Boundary

I. DATASET

The training dataset has two features namely 'Feature 0' and 'Feature 1' and a binary dependent variable (0s and 1s) which specified the class labels. Similarly, a test dataset is provided and the objective is to make the best prediction of "Class Labels" using one of the SVM methods. Both the training and testing dataset consisted of 200 data points each. The scatter plot of training data is indicated in Fig 1.

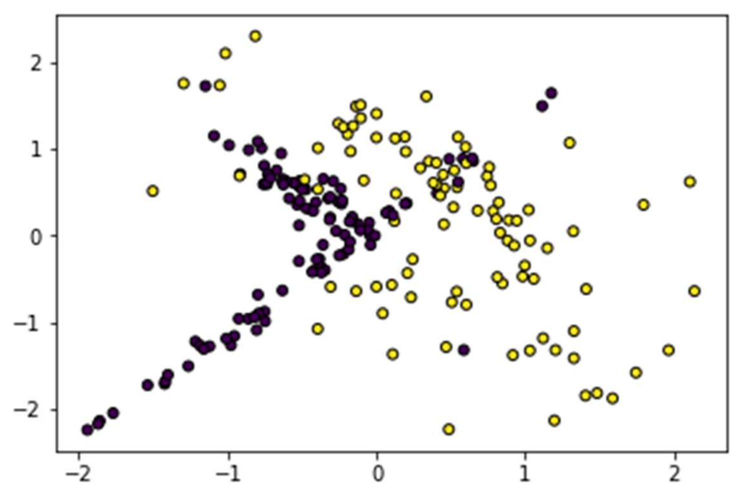

Fig-1: Scatter plot of training data

II MODEL SELECTION AND DECISION BOUNDARY ANALYSIS

The three Support Vector Machine (SVM) classifiers that I am using for this challenge are linear, Radial Basis Function (RBF) and polynomial kernels. I have set the SVM Regularization parameter C=3 and keeping other default parameter specifications as in the Scikit Learn library in python [1]. It is easier to visualize figure 1 and reach to a conclusion that a 'linear boundary' cannot separate the two classes optimally, so we need to look for some classifiers that can separate the data using some non-linear decision boundaries.

The problem statement, asks us to use a SVM classifier; and so we can easily test and train different kernel functions(linear and non-linear decision boundary estimators) and see which kernel gives the highest training accuracy. Sklearn's SVM API gives us three kernels essentially, viz RBF, Linear, Polynomial.

III. SCORES OF DIFFERENT KERNELS

I trained the SVM classifier using the above three kernel functions through the default parameters, and tried to deduce which kernel can give the highest accuracy. The scatter plots of those with the respective decision boundaries are shown below. Linear kernel achieved a training accuracy of 82% whereas Polynomial kernel showed only 76.5 % training accuracy. The highest accuracy was achieved by the RBF kernel function with 89 %; which no doubt generates a non-linear decision boundary with ease for complex datasets.

From our initial analysis above we see that RBF and linear kernels have the highest and second highest classification accuracies respectively. In the next section, we will optimize and the tune the hyper-parameter of these two classifiers and measure their accuracy to design the best classifier.

SVM Kernels	Mean Cross Validation Accuracies
Linear	82%
RBF	89%
Polynomial	76.5%

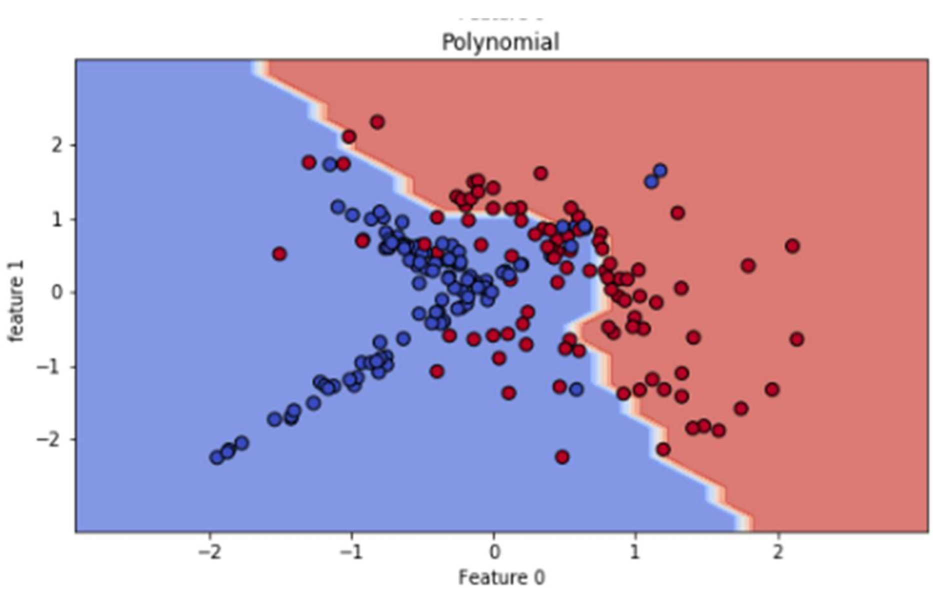

Fig 2. Polynomial Kernel

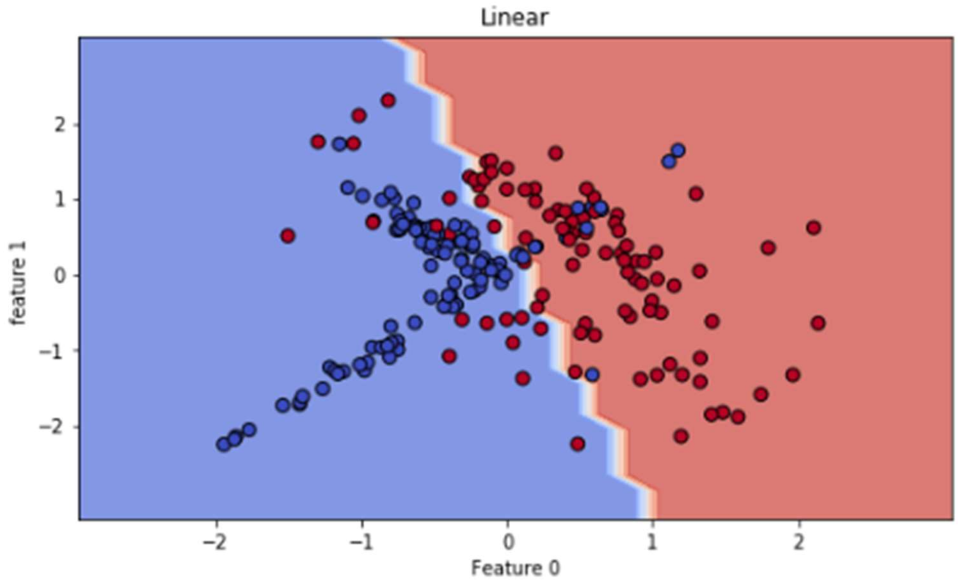

Fig 3. Linear separator Kernel

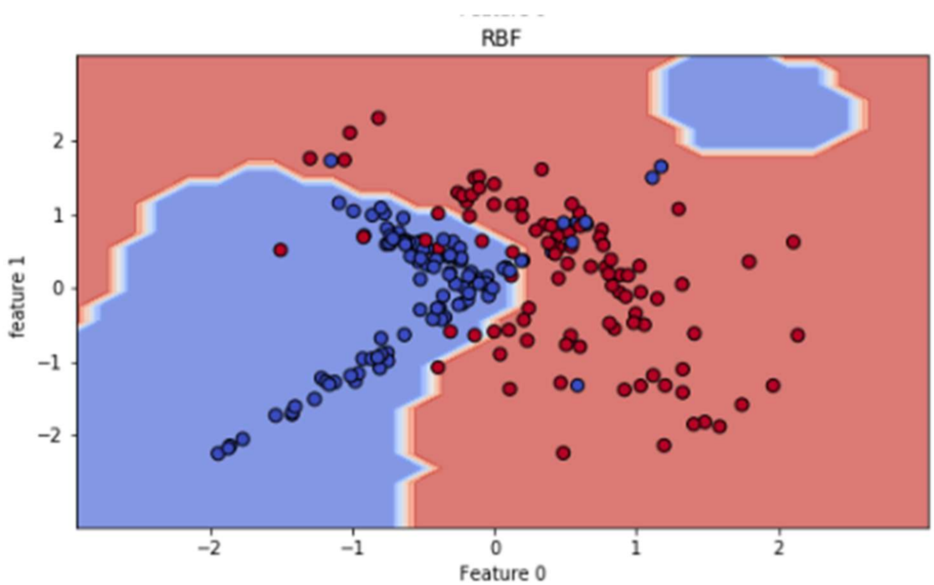

Fig 4. RBF Kernel

IV. Parameter tuning

Fine tuning the hyper parameters is an essential procedure in machine learning problems, and once I tuned the hyper parameters viz gamma and C(regularization constant), I was able to achieve a 3.5% increase in the accuracy. The scatter plot with the decision boundary of the final classifier is shown.

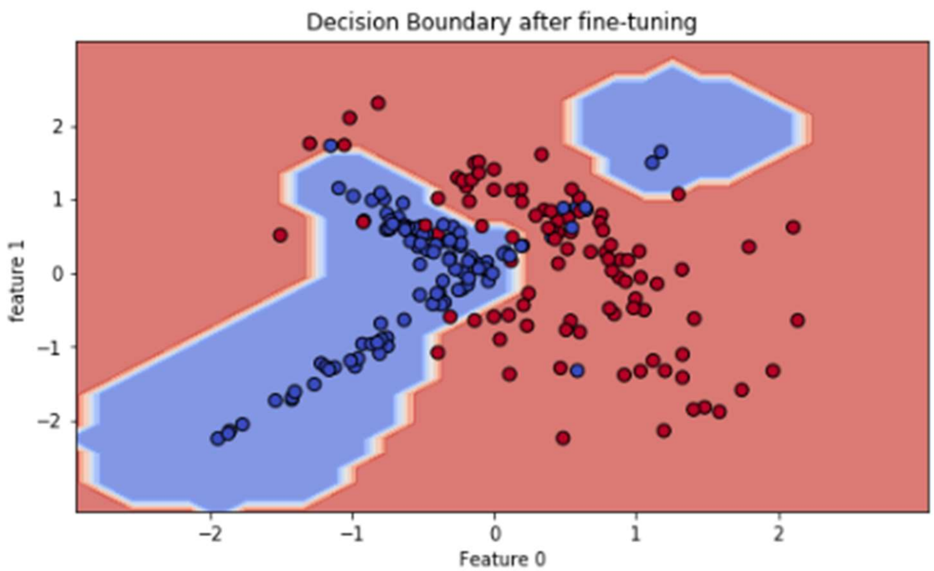

Figure 5. RBF Kernel after fine-tuning

V. CONCLUSION

In the given problem, I used a RBF kernel with gamma equal to 1.2 and regularization parameter as 1.8. The score that was achieved on the training data (92.5%) was much more than any other kernel and thus, the test data classes were predicted using the same.

VI. References

[1] Scikit-learn.org. (2018). RBF SVM parameters — scikit-learn 0.20.0 documentation. [online] Available at: http://scikit-learn.org/stable/auto_examples/svm/plot_rbf_parameters.html [Accessed 5 Nov. 2018].

[2] Scikit-learn.org.(2018). sklearn.model_selection.GridSearchCV—scikit-learn0.20.0 documentation.[online]Availableat:http://scikitlearn.org/stable/modules/generated/sklearn.model_selection.Gri d SearchCV.html [Accessed 7 Nov. 2018].