Classification with Support Vector Machines for double moon

Manjunath Dharshan Shanthigrama Rangaswamy
Ira A Fulton Engineering
Arizona State University
Tempe, USA
mdshanth@asu.edu

***Abstract*—This paper presents the use of Support Vector Machines to make classification determinations for double moon problem. In this work, a novel approach is proposed to make classification determination for the data which is not linearly separable. SVM with 3 different kernel and regularized parameter are used to determine the non-linear decision boundary.**

Keywords—Support Vector Machines, classification, MATLAB

I. Introduction

Support vector machine and its extensions have been widely applied to pattern recognition, microarray classification, system identification and so on. Support Vector Machines (SVM) are specifically used for classification problems. In the simplest scenario (linearly separable data), an SVM divides the data set into two classifications, or groups, using a straight line, as seen in Fig. 1, where the red dots correspond to one classification, and the green dots to another.

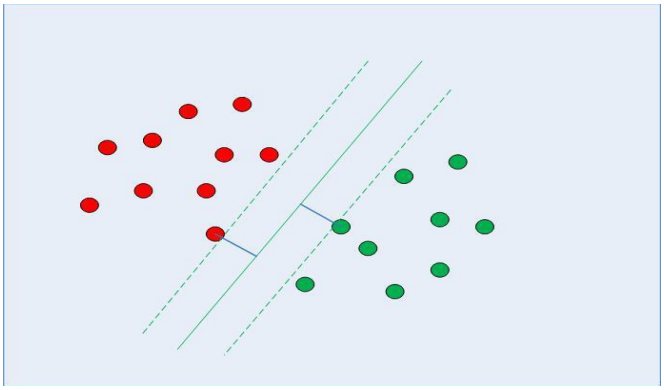

Figure 1: Simple SVM Classification [1]

The location of the classification dividing line is determined by identifying the line location that leads to the greatest overall distance between the line and the nearest points of each classification. The lines that run tangential to the closest points are known as support vectors and are represented by the dotted green lines in Fig. 3. The plane formed by the area between the two support vectors is known as the hyperplane.

For more complicated data sets with data that is not linearly separable (Fig 2), the classifications may not be identified by a simple straight line. In this case a kernel SVM may be used. A simplified explanation of a kernel SVM is that it mathematically projects the data to higher dimensions.

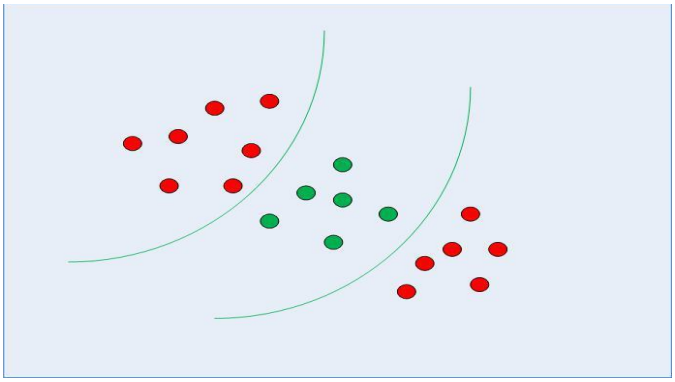

Figure 2: Data Separated after using Kernel SVM [1]

II. Methodology

Support Vector Machines with 3 different kernel and regularization parameters were applied to the created dataset double moon to make classification determinations. Specifically, 1000 data points 500 each in Region A and B respectively were used for training purpose and 400 pairs from training samples were used to make classification determinations. Regularization strategies avoid overfitting by choosing the function that fits the data which is shown in eq 1. The available kernel functions are shown in eq 2.

$$f = \underset{f \in \mathcal{H}}{\operatorname{argmin}} \left\{ \frac{1}{n} \sum_{i=1}^{n} \left(1 - y f(x)\right)_+ + \lambda \|f\|_{\mathcal{H}}^2 \right\} \tag{1}$$

$$K(\mathbf{X_i}, \mathbf{X_j}) = \begin{cases} \mathbf{X_i} \cdot \mathbf{X_j} & \text{Linear} \\ (\gamma \mathbf{X_i} \cdot \mathbf{X_j} + \mathbf{C})^d & \text{Polynomial} \\ \exp\left(-\gamma |\mathbf{X_i} - \mathbf{X_j}|^2\right) & \text{RBF} \\ \tanh\left(\gamma \mathbf{X_i} \cdot \mathbf{X_j} + \mathbf{C}\right) & \text{Sigmoid} \end{cases}$$

$$\text{where } K(\mathbf{X_i}, \mathbf{X_j}) = \phi(\mathbf{X_i}) \bullet \phi(\mathbf{X_j}) \tag{2}$$

III. Implementation

Support Vector Machines with 3 different kernel and regularization parameters were applied to the created dataset double moon to make classification determinations. Specifically, 1000 data pairs 500 each in Region A and B

respectively were used for training purpose and 400 pairs from the training samples were used for verification of the trained neural network. The radius of the inner and the outer ring is 10 and 16 respectively with both rings separated at a distance of -12. The data points were created using MATLAB inbuilt function rand within the specified limits. The created data samples is as seen in Fig. 3.

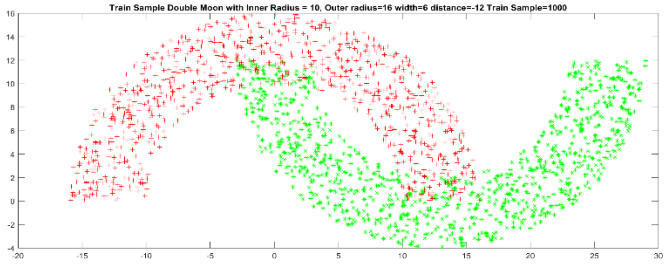

Figure 3: Training set with 1000 pairs of training samples

Support Vector Machine algorithm trained with 3 different kernel rbf, and 2 custom kernel TanH and GHI. The model was trained for 3 different regularization parameters with C1 = 0.01, C2 = 1, C3=100.

IV. Results

From the observation we can see SVM rbf kernel performed well around 94% accuracy on test data points. TanH and GHI kernel performed well for higher regularized parameter with C1 = 0.01 with test accuracy 70.4% and 88.5% respectively.

A. *Tables and Figures*

Table 1 shows the SVM model performance with 3 different kernel and varying regularized parameters.

TABLE I. Test Accuracy

Kernel	Regularized Parameter	Accuracy (%)
RBF	0.01	94.6
	1	94
	100	94
GHI	0.01	88.5
	1	87.5
	100	87.3
Sigmoid TanH	0.01	70.4
	1	49.6
	100	49.6

The Decision Boundary and the Prediction on Test Sample for 3 different kernel and C1=0.01 and confusion matrix for the same is seen in Fig. 4.1 and Fig 4.2 respectively. The confusion matrix and DB for other parameter is saved in the zipped folder.

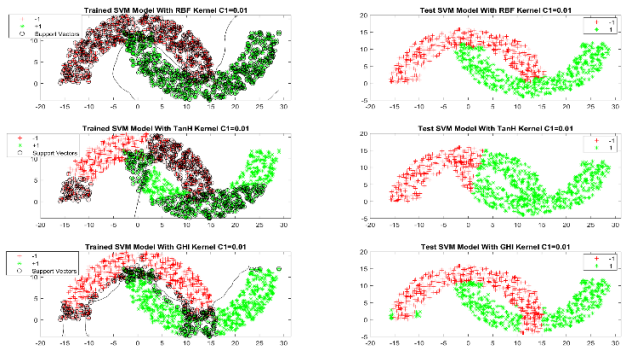

Figure 4.1: DB, Support Vectors and Prediction for C1=0.01

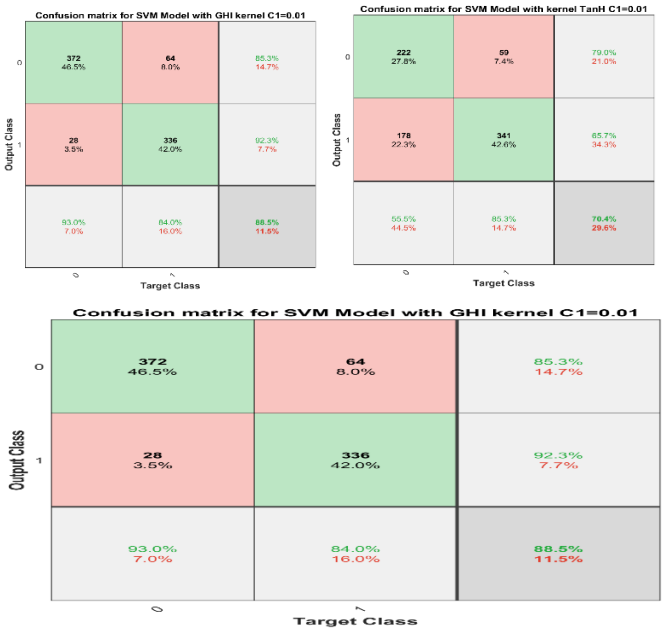

Figure 4.2: Confusion Matrix with C1=0.01

Figures (4.2) and (4.3) shows the Decision boundary of the test data and the Performance plot for the trained neural network with 3 hidden layer, Epoch 40 and learning rate 0.01

V. Conclusion

The Support Vector Machine has been shown to be a useful tool for classification for non-linear data. Based on the observation we can see as the regularized parameter value increased the accuracy dropped in all three kernels but with lower value the model was overfitting. Regularization strategies avoided overfitting and paved way for optimal solution. RBF kernel performed better for non-linear data points with prediction accuracy of 94.6% for lower regularized parameter.

References

[1] U. Malik, "Implementing SVM and Kernel SVM with Python's Scikit-Learn," *Stack Abuse*, 18-Apr-2018. [Online]. Available: https://stackabuse.com/implementing-svm-and-kernel-svm-with-pythons-scikit-learn/. [Accessed: 13-March-2019].

[2] STATSOFT:http://www.statsoft.com/textbook/support-vector-machines

[3] MATLAB: "https://www.mathworks.com/examples/statistics/mw/stats-ex32265893-train-svm-classifier-using-custom-kernel.