Machine Learning Course Project

Daya Mariam Alex
M1 AI Roll No 09
College of Engineering Trivandrum Kerala

***Abstract*—The course project aimed to study Support Vector Machines(SVM) in classification tasks for three different datasets. Experiments on the first data set involved varying the value of regulariser C and studying the changes in classification patterns. It was found that when C is small, it will permit misclassification but as C becomes large it will penalize the misclassification and reduce misclassifications to as if it was a hard margin SVM(C=0). The second dataset was a binary image file of handwritten digits 0 and 1 labeled as -1 and 1. After processing the dataset as required, SVM was used to classify the two different digits. Hard margin SVM was able to achieve 99.91% accuracy with only 2 misclassifications in the test set of 2115 images. Additionally SVM fitted with different C's was used to assess the changes in accuracy in the train and test set. The third dataset contained features where no clear linear decision boundary could separate the classes. The effectiveness of Radial-basis-Kernel-SVM for classifying this dataset was demonstrated. Different values of γ were tried and γ= 1 was found best for classifying the dataset.**

Keywords: Support Vector Machine(SVM), regularizer C, Handwritten Digit recognition, Radial Basis Kernel(RBK)

I. Introduction

The learning task involves implementing Support Vector Machines (SVMs) for both linear and non-linear classification and applying it to solve two different problems: soft margin SVM on two given data sets for handwritten digit recognition, and kernel SVM on a two-dimensional classification problem.

For the first part, the task involves plotting the decision boundary of the SVM with the training data, using the test data to evaluate the SVM classifier and showing the fraction of test examples that were misclassified. The task also requires trying different values of the regularization term c and reporting observations.

The second part of the task involves applying the SVM classifier to recognize handwritten digits, specifically distinguishing between 0's and 1's. The task requires training a linear SVM on the given training data and computing the training error. The task also involves experimenting with different values of the regularization term c, plotting the corresponding error, and answering questions related to the performance of the SVM.

The third part of the task involves applying an RBF kernel to a two-dimensional classification problem, plotting the positives and negatives using different colors, and determining whether a linear decision boundary exists for this dataset. The task also requires training an SVM model on an RBF kernel with different values of gamma and plotting the decision boundary for each model while observing how the plot of the boundary changes with gamma.

II. Description of the experiments

A. Soft margin SVM

The dataset was a .txt file, it first had to be parsed and converted to a .csv file. After converting both train and test sets, the features and their respective labels were extracted. SVM classifier was implemented on the train and test sets using the sklearn library and plots were rendered. Both test and train sets were drawn in the same plot using different color gradients available in the matplot library. Support vectors were encircled in black. Different SVM linear kernels with C values 2, 0, 0.01, 0.001, 0.0001 were drawn on dataset 1 and C values 1, 0, 0.005, 0.0005 on dataset 2 respectively.

B. Handwritten Digit Recognition

To perform SVM on the images, the binary image file needs to be preprocessed so that each row of data contains the 784 gray scale pixel values for that image. To get such a list, from the original file we must extract the pixel values and their respective indices to two separate lists and then arrange them into a 784-dimensional list for each image. SVM is trained on the changed dataset and labels. The misclassified instances are appended to an array. 10 different values for C are implemented on the SVM to analyze the effect of C on accuracy for both train and test datasets.

C. Radial basis kernel SVM

The dataset is first scatter plotted to see if a linear separation is possible. As such a separation is not possible, SVM with Radial basis kernel function is implemented. Scatterplots are drawn for γ= 1, 10, 100, 1000.

III. Results

A. Soft margin SVM

The train set was plot in autumn gradient and test set was plot in cool gradient.Hyperplanes marked in yellow and support vectors encircled in black.

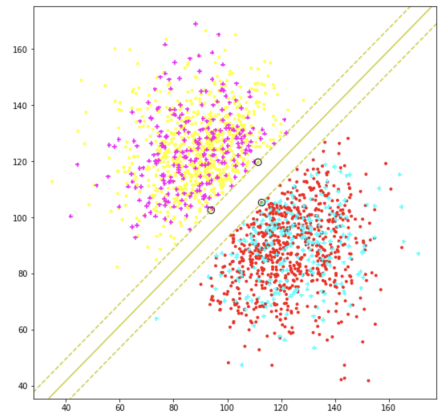

Fig. 1. Dataset 1, C=0,2 both yields same plot, misclassification is zero

[?]

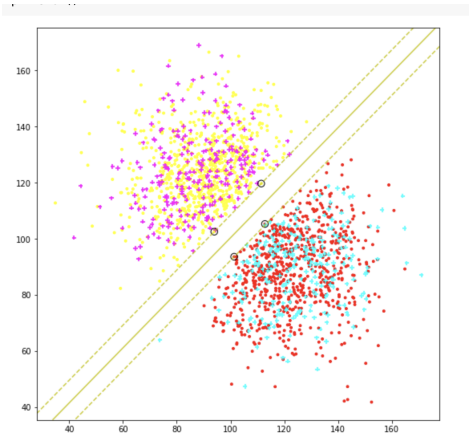

Fig. 2. Dataset 1, C=0.01, number of support vectors increase, misclassification is still zero

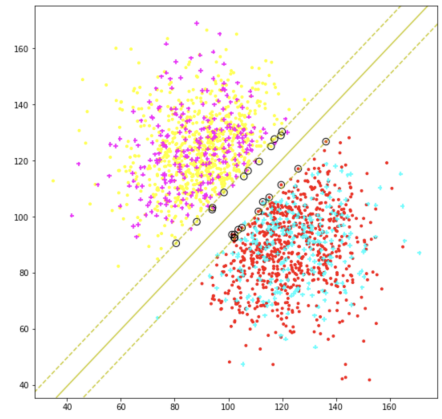

Fig. 3. Dataset 1, C=0.001, misclassification permitted

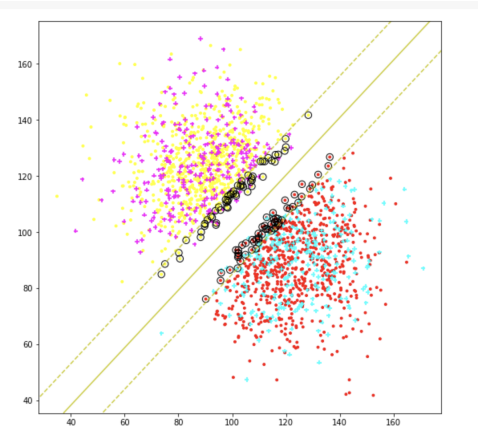

Fig. 4. Dataset 1, C=0.0001, more misclassification permitted

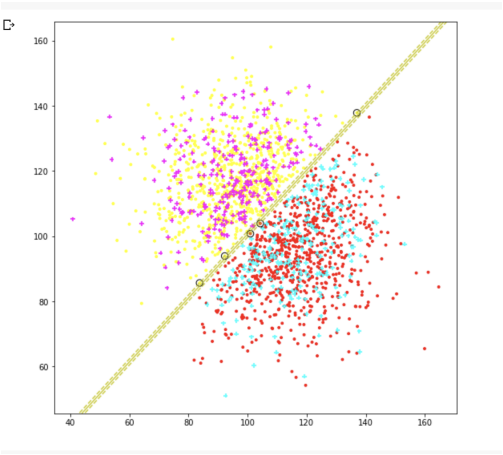

Fig. 5. Dataset 2, C=0, 1 , No misclassification permitted

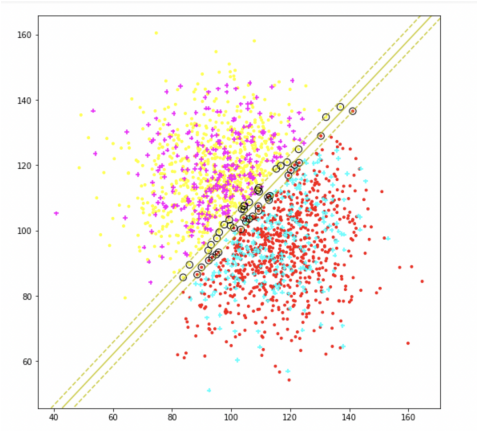

Fig. 6. Dataset 2, C=0.05, Misclassification permitted

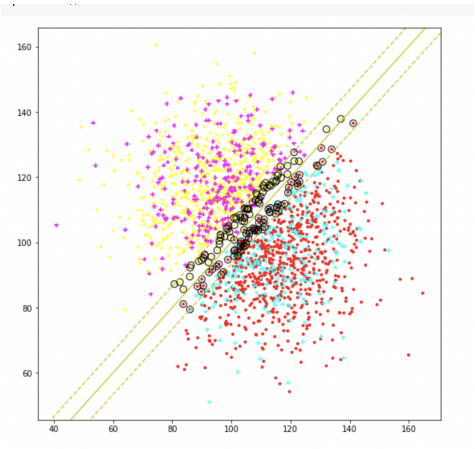

Fig. 7. Dataset 2, C=0.005, More misclassification permitted

B. Handwritten Digit Recognition

SVM is able to classify the two handwritten digits well, the misclassification samples reveal that the models performance is satifactory.

```
[29] model.score(x_trainim, y_trainim)
     1.0

[30] model.score(x_testim, y_testim)
     0.9990543735224586
```

Fig. 8. Train and Test accuracy of hard margin SVM in Handwritten Digit recognition

```
C=  5e-12 - Accuracy:  53.233320173707064
C=  4.99999999999995e-11 - Accuracy:  87.49309119621003
C=  4.9999999999999e-10 - Accuracy:  99.60521121200158
C=  4.999999999999e-09 - Accuracy:  99.8026056060008
C=  4.99999999999999e-08 - Accuracy:  99.8973549151204
C=  4.99999999999999e-07 - Accuracy:  99.9526253454402
C=  4.9999999999999e-06 - Accuracy:  100.0
C=  4.999999999999e-05 - Accuracy:  100.0
C=  0.0004999999999999999 - Accuracy:  100.0
C=  0.004999999999999999 - Accuracy:  100.0
C=  0.04999999999999999 - Accuracy:  100.0
C=  0.4999999999999999 - Accuracy:  100.0
```

Fig. 9. Changes in training accuracy scores for handwritten digit regonition with 10 values of C

```
C=  5e-12 - Accuracy:  53.664302600472816
C=  4.99999999999999995e-11 - Accuracy:  86.95035460992908
C=  4.9999999999999e-10 - Accuracy:  99.76359338061465
C=  4.999999999999999e-09 - Accuracy:  99.90543735224587
C=  4.999999999999999e-08 - Accuracy:  99.90543735224587
C=  4.999999999999999e-07 - Accuracy:  99.95271867612293
C=  4.999999999999999e-06 - Accuracy:  99.90543735224587
C=  4.999999999999999e-05 - Accuracy:  99.90543735224587
C=  0.0004999999999999999 - Accuracy:  99.90543735224587
C=  0.004999999999999999 - Accuracy:  99.90543735224587
C=  0.04999999999999999 - Accuracy:  99.90543735224587
C=  0.4999999999999999 - Accuracy:  99.90543735224587
```

Fig. 10. Changes in testing accuracy scores for handwritten digit regonition with 10 values of C

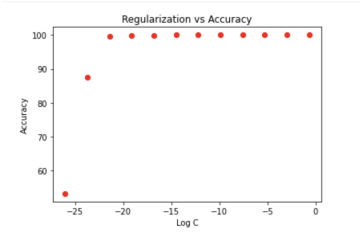

Fig. 11. Regularisation (log C) vs Accuracy

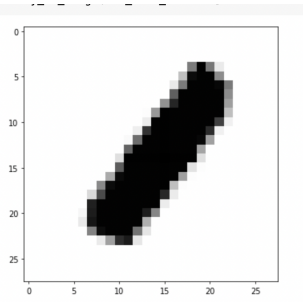

Fig. 12. Missclassified test sample 1

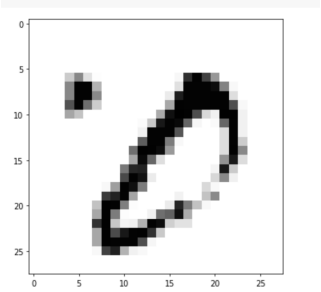

Fig. 13. Missclassified test sample 2

IV. Radial Basis Kernalised SVM

Radial Basis Kernal can be used for SVM's where non-linearity requires to be introduced. RBK-SVM with γ= 1, works best as a seperator that allows misclassifications without leading to overfitting.

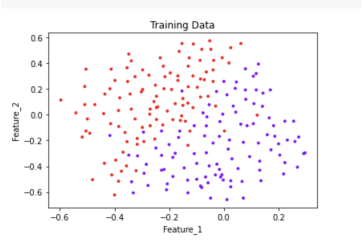

Fig. 14. Training data scatter plot of dataset 3, No linear seperator can be used for classification of this dataset

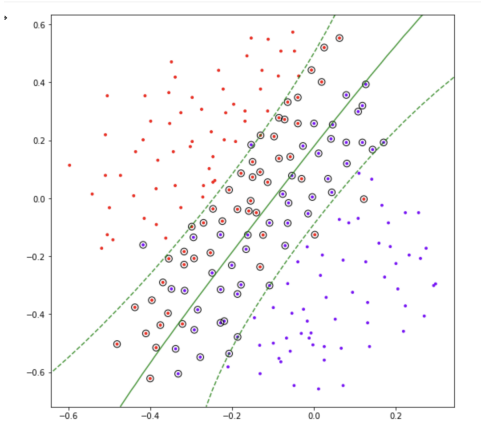

Fig. 15. RBK-SVM with γ= 1, linear seperation with misclassification created, good seperator function, support vectors located close to the decision plane

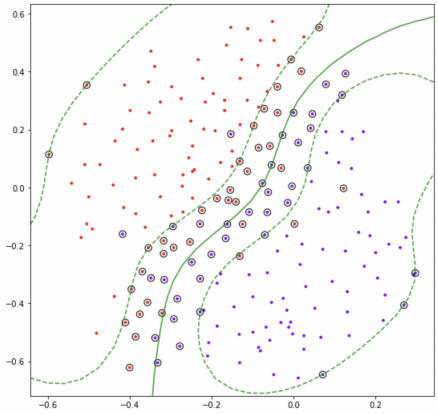

Fig. 16. RBK-SVM with γ= 10, non linearity of seperator increased, support vectors influence increased from decision plane

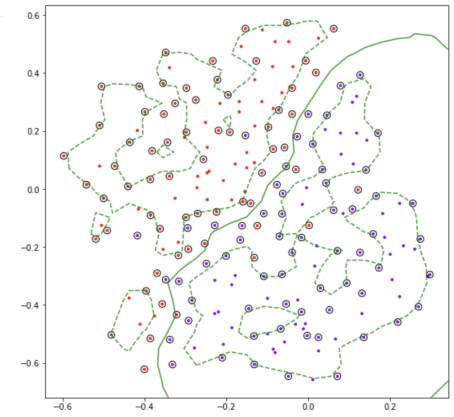

Fig. 17. RBK-SVM with γ= 100, non linearity of seperator increased to the extend of overfitting

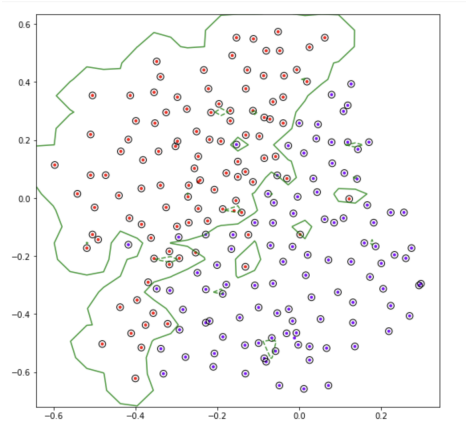

Fig. 18. RBK-SVM with γ= 1000, Highly overfitted, all datapoints become support vectors

V. Significane of results

The robustness of SVM machine learning on classification tasks was elucidated. SVM's can result in powerful predictive models because of distinct features like regularising parameter C and kernel functions. Regularising parameter C allows us to control the extend of misclassification by proving a mechanism to penalise errors from misclassification. Kernel functions allow as to introduce non-linear decision boundaries at a low computational cost.

VI. Conclusion

In this course project SVM properties of regularisation parameter C, kernel function- RBK was studied using different datasets. Additionally SVM was implemented to recognise two handwitten digits.