



Banknote Authentication Problem

Learning from Data Lab Task 3



Name	ID
Mayar Ahmed Ibrahim	20200572
Sara Salah Mohamed	20200216

Supervised by:
DR.El-Shimaa Elgendi.

Pre-Processing

First of all we cleaned the data by removing duplicates row from data frame and removes the rows that contains Null values.

Modeling

divide data into input "x" and output "y" to make x train, y train, x test, and y test, size on 30 percentage.

Using x and y train to make my model using machine learning, comparing the output of the model to my y test calculating accuracy and error percentage based on predicted values.

In x train: we need to compute the mean and std dev for a given feature to be used further for scaling perform scaling using mean and std dev calculated using the .fit() method. So, we used (fit_tranform(x_train)).

In X test: we need to only perform scaling using mean and std dev calculated using the .fit() method. So, we used (transform(x_test)).

Determining the suitable batch size for our data set

Then implement the SVM Soft and hard with updating batch size every iterator.

SVM (Support Vector Machines)

It is one of the classification techniques that aim to minimize the number of misclassification errors directly.

- **SVM Soft**

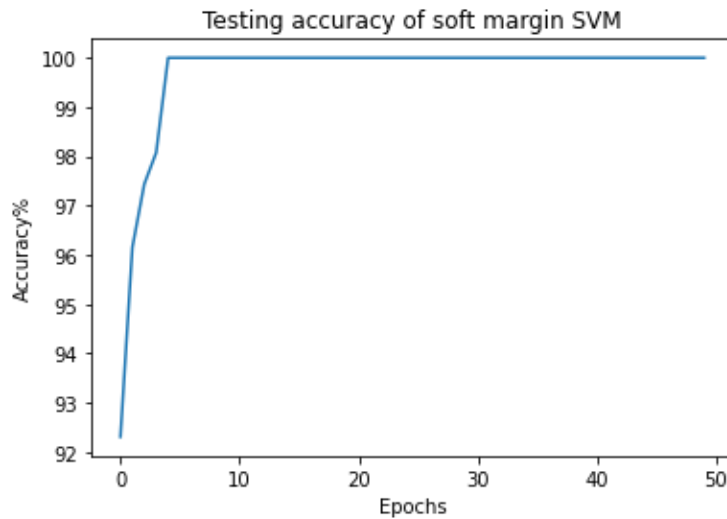
This idea is based on a simple premise: allow SVM to make a certain number of mistakes and keep margin as wide as possible so that other points can still be classified correctly. This can be done simply by modifying the objective of SVM.

- **SVM Hard**

If the training data is linearly separable, we can select two parallel hyperplanes that separate the two classes of data, so that the distance between them is as large as possible

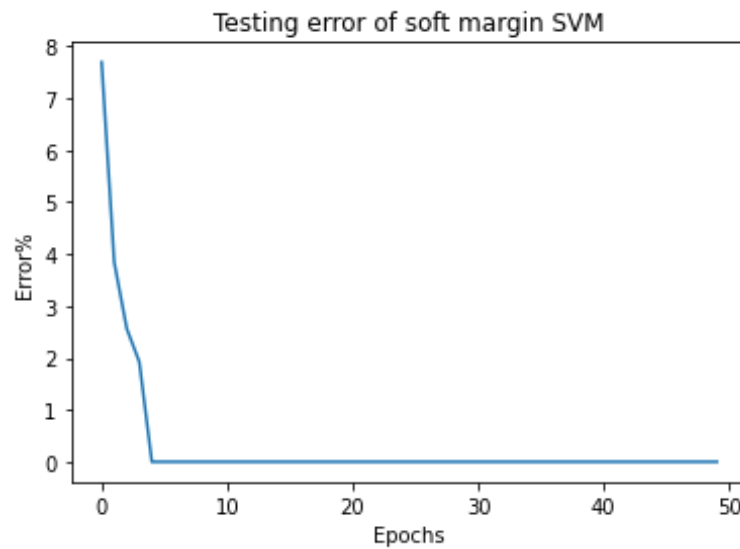
Experimental Design and analysis of the results

Graph showing the Testing accuracy of soft margin SVM



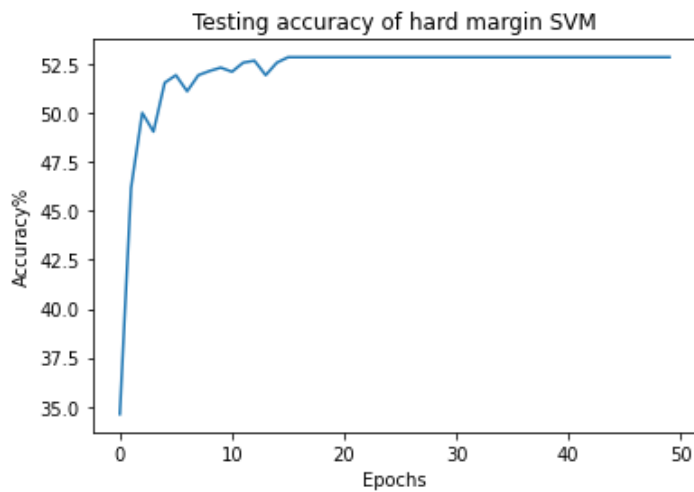
line graph showing the network accuracy for the test set across epochs 1 to 50 which equal to 100 %

Graph showing the Testing error of soft margin SVM



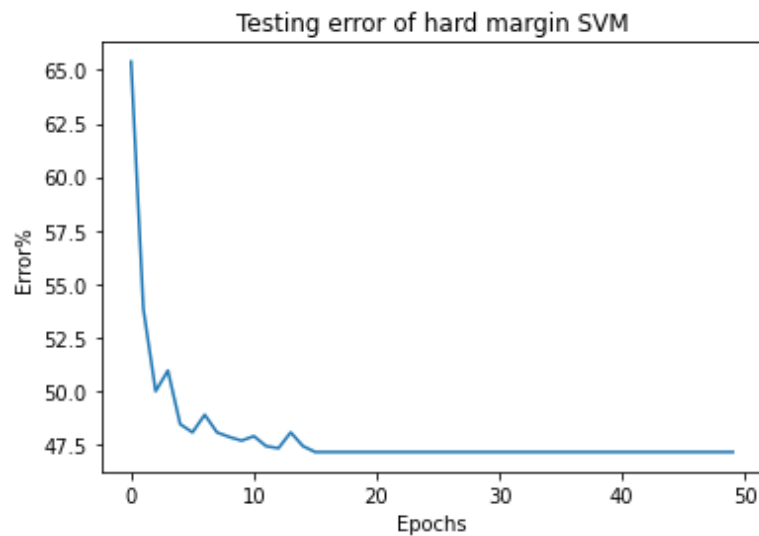
line graph showing the network error for the test set across epochs 1 to 50 which equal to zero.

Graph showing the Testing accuracy of hard margin SVM



line graph showing the network accuracy for the test set across epochs 1 to 50 which equal to 52.8395061728395%

Graph showing the Testing error of hard margin SVM



line graph showing the network error for the test set across epochs 1 to 50 which equal to 47.1604938271605

ROC architecture

ROC or Receiver Operating Characteristic plot is used to visualize the performance of a binary classifier. It gives us the trade-off between the **True Positive Rate (TPR)** and the **False Positive Rate (FPR)** at different classification thresholds

True Positive Rate:

The true Positive Rate is the proportion of observations that are correctly predicted to be positive.

$$\text{TPR} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}}$$

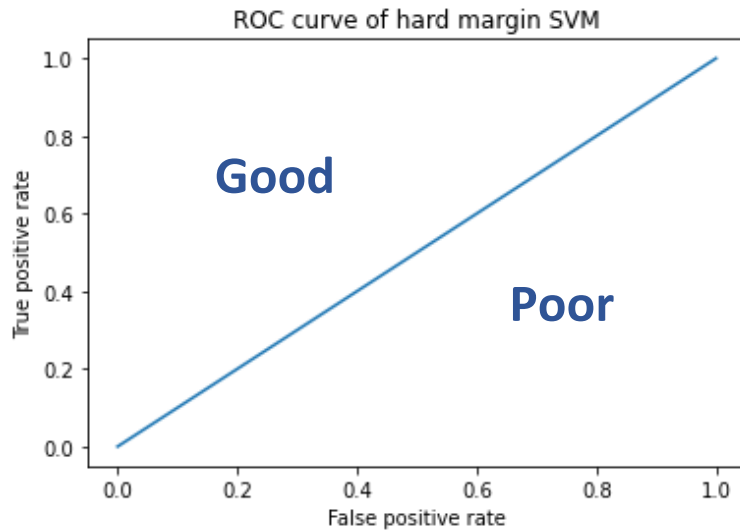
False Positive Rate:

The false Positive Rate is the proportion of observations that are incorrectly predicted to be positive.

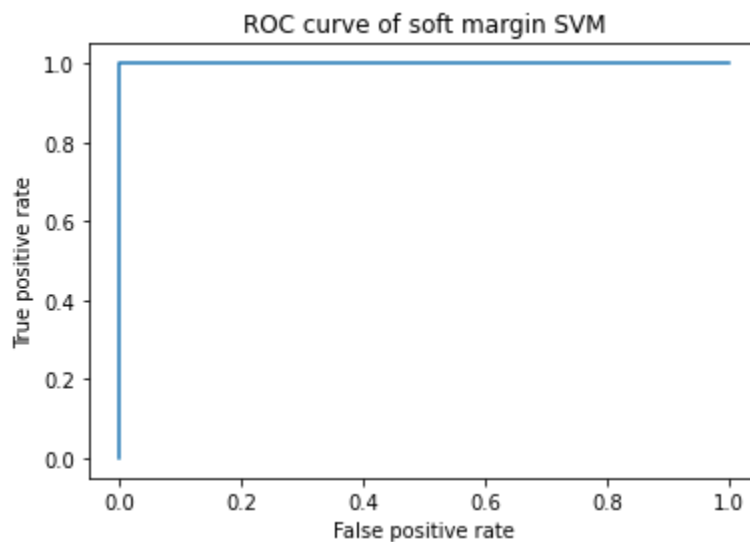
$$\text{FPR} = \frac{\text{False Positive}}{\text{True Negative} + \text{False Positive}}$$

ROC Curve

The ROC curve of a **random classifier** with a random performance level always shows a straight line. This random classifier ROC curve is considered to be the baseline for measuring the performance of a classifier” SVM Hard”. Two areas separated by this ROC curve indicate an estimation of the performance level which is good” SVM soft”.

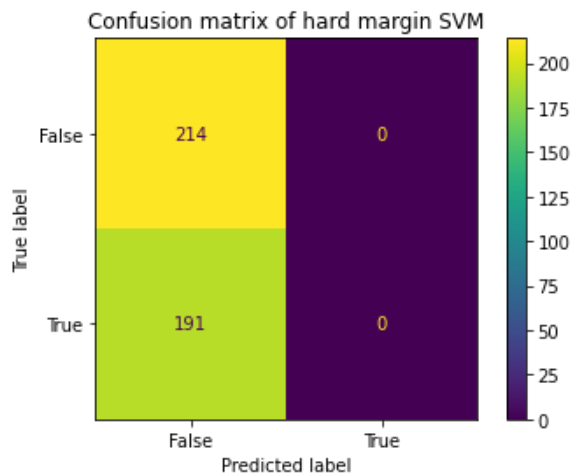


ROC curves that fall under the area at the top-left corner indicate good performance levels, whereas ROC curves fall in the other area at the bottom-right corner indicate poor performance levels



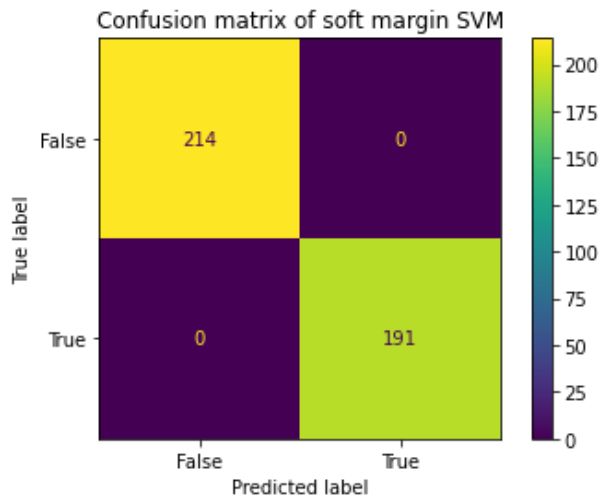
That represent a good classifier stays as far away from that line as possible (toward the top-left corner).

Confusion matrix for hard margin SVM



- True Negatives:214
- False Positives: 0
- False Negatives: 0
- True Positives:191

Confusion matrix for soft margin SVM



- True Negatives:214
- False Positives: 0
- False Negatives: 0
- True Positives:191

True categories explained above where the actuals and predictions are conforming, False categories indicate that prediction is not matching the ground truth

Conclusion

From the previous data, we get that SVM soft is a better implementation for our data than SVM hard.

As SVM soft has full accuracy with 100% and no error percentage otherwise, SVM hard implementation low accuracy = 52.8395061728395% and error = 47.1604938271605%