Akshay Kenchappa Manjunath EE 559
Homework 4
February 21, 2020

Problem 1

Part(a)

Synthetic 1 dataset

The final weight vector is [-56.82901 62.31412 28.1]

The number of misclassified samples for the training data is 2. The error-rate for training data is 2%.

This was obtained after carrying out multiple trials where the consistent value obtained was a misclassification of 2. The code put up shows for instance a misclassification of 4. This is because reshuffling the data leads to different results due to the fact that datasets 1 and 2 are not linearly separable, so the results will depend on where you start as the algorithm does not converge, and you will end up halting at the second condition.

The number of misclassified samples for the training data is 3.The error-rate for test data is 3%.

This was obtained after carrying out multiple trials where the consistent value obtained was a misclassification of 3.

Henceforth the results put up are the most consistent values obtained over many random executions.

Synthetic 2 dataset

The final weight vector is [-2.033105 18.741684 6.1]

The number of misclassified samples for the training data is 1. The error-rate for training data is 1%.

The number of misclassified samples for the training data is 3.The error-rate for test data is 3%.

Synthetic 3 dataset

The final weight vector is [-8.78156 7.8604 1.1]

The number of misclassified samples for the training data is 0. The error-rate for training data is 0%.

The number of misclassified samples for the training data is 1.The error-rate for test data is 1%.

Part (b)

2D Plots showing decision boundary and regions, as well as training data points.

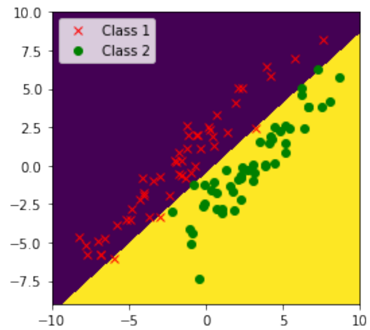

Figure 1: Plot showing classification of Synthetic1 training dataset

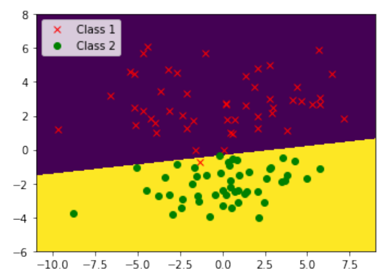

Figure 2: Plot showing classification of Synthetic2 training dataset

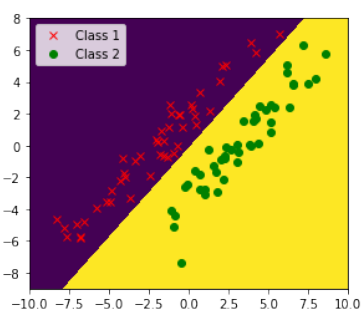

Figure 3: Plot showing classification of Synthetic3 training dataset

Part (c)

Synthetic 1 dataset

The number of misclassified samples for the training data is 21 and the number of misclassified samples for the test data is 24. Consequently,the error-rate for training data is 21% and the error-rate for test data is 24% respectively in HW1(a).

Comparing the error rates obtained in part(a) here, we can conclude that 2-class perceptron classifier works much better and gives a very low error rate.

Synthetic 2 dataset

The number of misclassified samples for the training data is 3 and the number of misclassified samples for the test data is 4. Consequently,the error-rate for training data is 3% and the error-rate for test data is 4% respectively in HW1(a).

Comparing the error rates obtained in part(a) here, we can conclude that 2-class perceptron classifier works considerably better and gives a lesser error.

Thus we can say that the perceptron is a much better classifier algorithm than the nearest means algorithm.

Differences between perceptron and nearest means looking at the results

Looking at the results obtained we can see that nearest means classifier is very sensitive to outliers. Without appropriate data processing, it can produce wrong predictions.

This is because Nearest means is the simplest type of classifier that assigns to observations the label of the class of training samples whose mean is closest to the observation. The main disadvantage of this approach is that the algorithm must compute the distance and sort all the training data at each prediction, which can be slow if there are a large number of training examples. Another disadvantage of this approach is that the algorithm does not learn anything from the training data, which can result in the algorithm not generalizing well and also not being robust to noisy data.

In the perceptron algorithm random weights get generated, which means that random simulation is happening in every iteration. When we have one desired output that we show to the model, the machine has to come up with an output similar to our expectation. Initially, it may not be as accurate. But, as the “training” continues the machine becomes more accurate. With more and more iterations, the error minimizes.

In synthetic dataset 1, the data has more ouliers .Employing nearest means classifirer on this data will lead to high error rates and more misclaffications. The Perceptron algorithm learns the weights for the input signals in order to draw a linear decision boundary. It gives a reduced error rate, thus showing that it can better adjust the weight vectors for the features, resulting in better decision boundary.Perceptron we are minimizing the error by employing criterion function and if the training set is linearly separable, then the perceptron is guaranteed to converge.

In synthetic dataset 1, the data has very few outliers. Hence here the error rates in both classification algorithms are relatively very low and quite similar.