

Problem 1

Method

We use a perceptron to separate out two sets of data: one linearly separable, and the other non-linearly separable. Linearly separable data was created by randomly initializing weights for 50 samples, and then doing class assignments along a predefined threshold boundary. Similarly, the non-linearly separable data was constructed with 50 samples, with randomly initialized weights. However, each feature was squared before class assignment in order to make the data unseparable in the original space.

We run the perceptron learning algorithm for both datasets for 1000 iterations or until the error is 0. The data and decision boundaries were plotted, along with the classification error over the number of iterations.

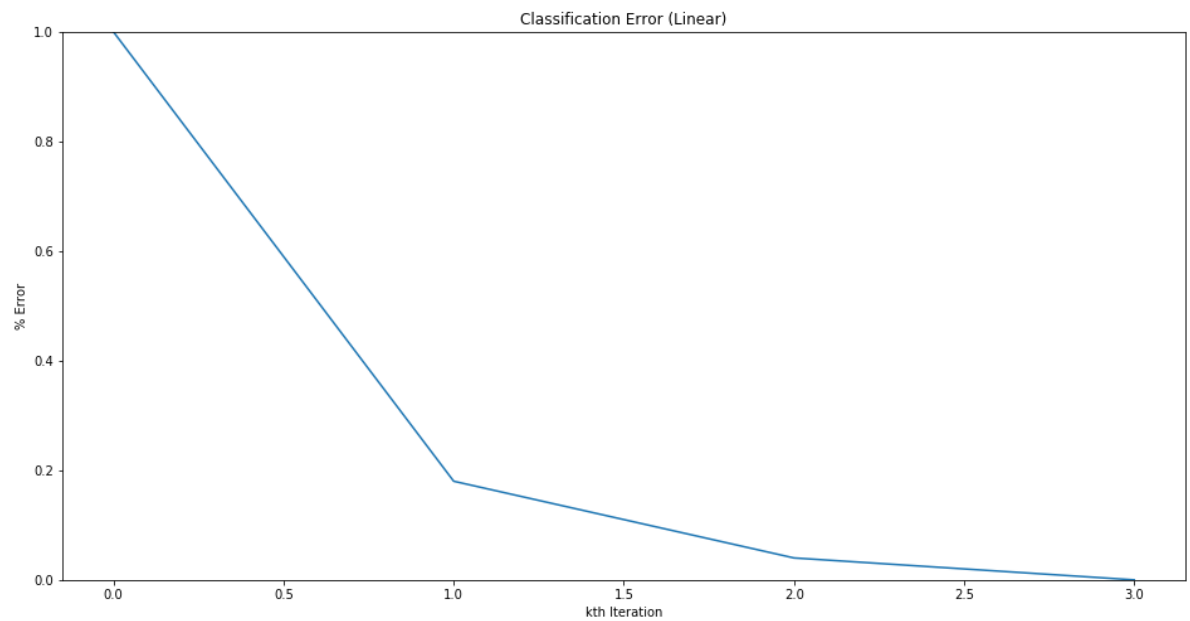
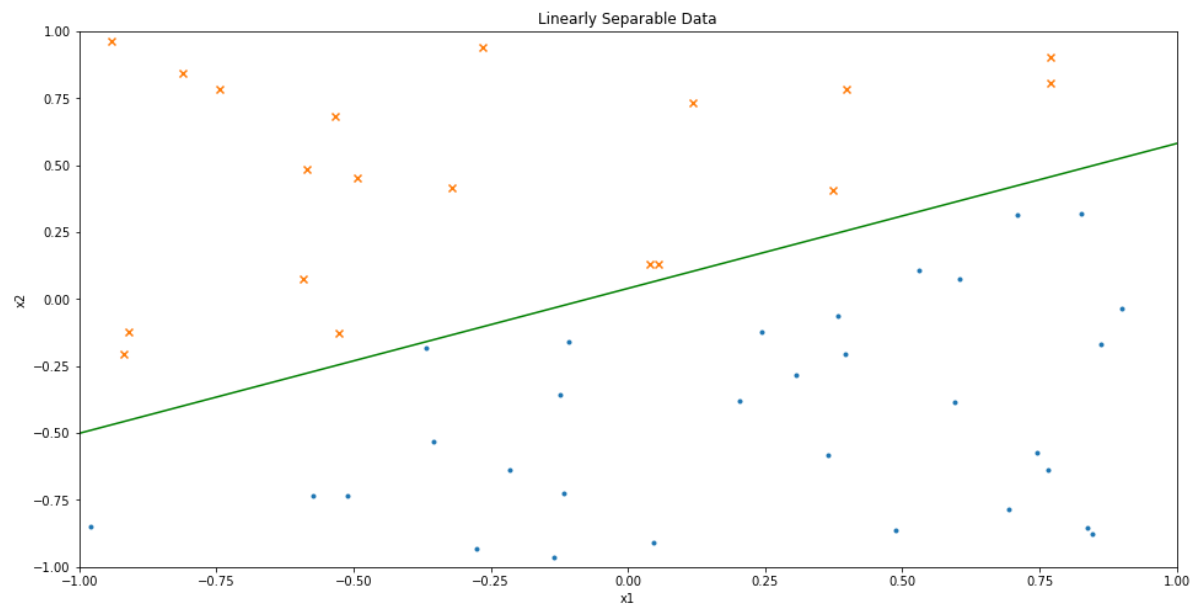
Out[1]: [Toggle Code](#)

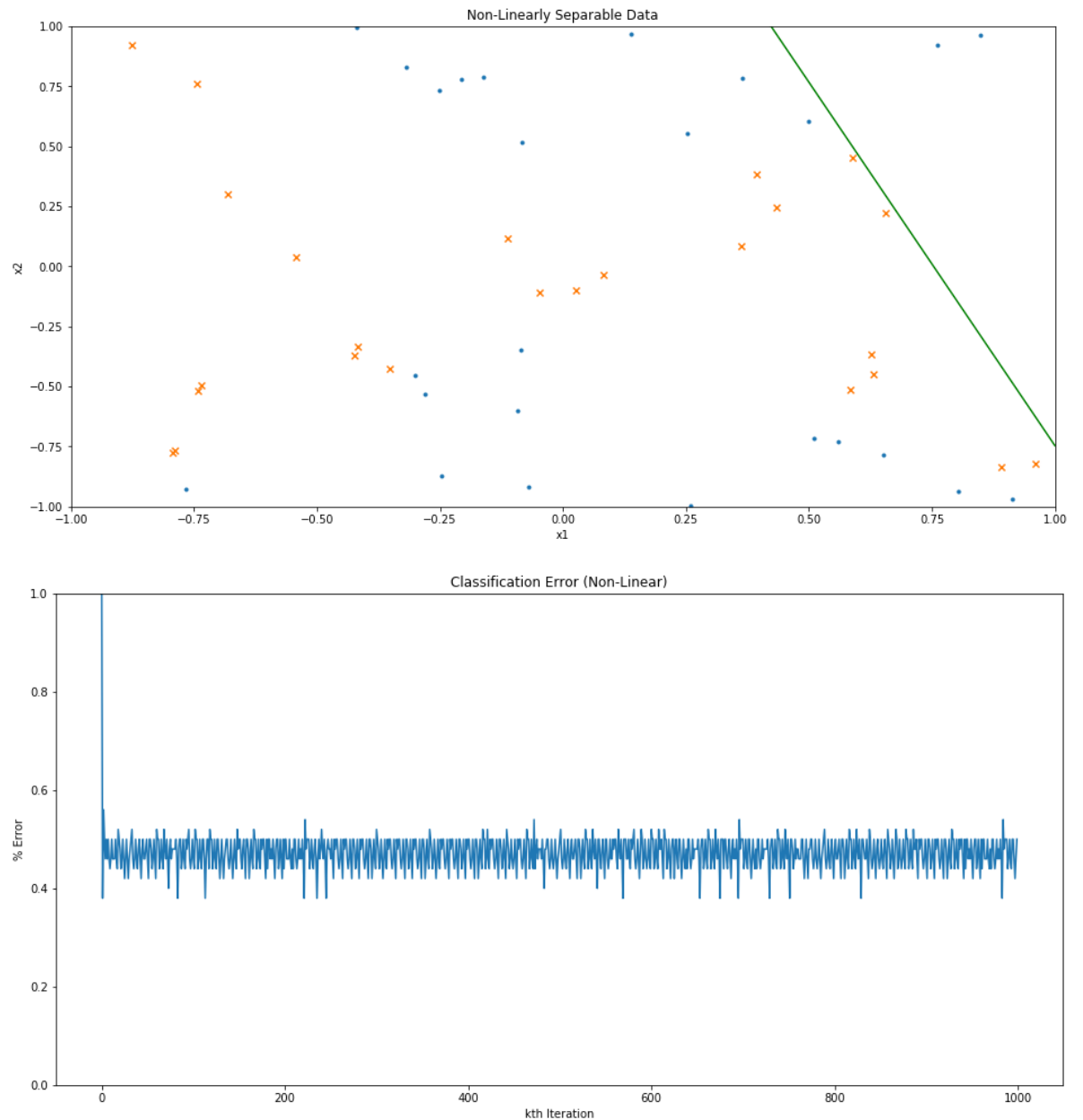
Linearly Separable Data:

Finished @ Iteration 3, Weights: [1. 13.54780038 -25.04214042], Error: 0.0

Non-linearly Separable Data:

Finished @ Iteration 999, Weights: [-11. 14.60004316 4.82544108], Error: 0.5





Discussion

The perceptron algorithm cleanly separates the linearly separable data in 3 iterations with 0 error. This is to be expected since perceptrons are linear models.

For non-linearly separable data, the perceptron fails and does not converge to a solution. Classification error for the non-linear data hovers around 50%, which is no better than random chance.