Electrical Engineering Department
Sharif University of Technology

Title:

Homework 2

Neural Networks

Author:
Mohammad Hossein Najafi (97103938)

Supervisor:
Dr. S. Bagheri

Spring 2024

Part One

1

It's correct because in the perceptron algorithm for classification, it creates a line between the minuses and pluses solely for classification purposes, without minimizing the error like in Adeline.

2

No, there is no guarantee for a perceptron to perform excellently in all cases. Its performance depends on the characteristics of the data and the complexity of the problem. For instance, factors such as linear separability of data and appropriate distribution of samples with a suitable ratio between classes are among the factors that can contribute to better performance of the perceptron.

3

As we are in one-dimensional space, we can separate the classes with a point. For State One, any point within the range (-8, 1) can serve as a separator and classifier. However, for State Two, it cannot be separated by a single point; it requires two points. Therefore, it couldn't.

4

No, a perceptron cannot solve the XOR problem because XOR is a nonlinear and non-linearly separable problem. To illustrate this, consider that XOR has two inputs (e.g., x_1 and x_2) and its output is zero if the inputs are the same, and one if the inputs are different. In other words, the truth table for XOR is as follows:

x_1	x_2	Output
0	0	0
0	1	1
1	0	1
1	1	0

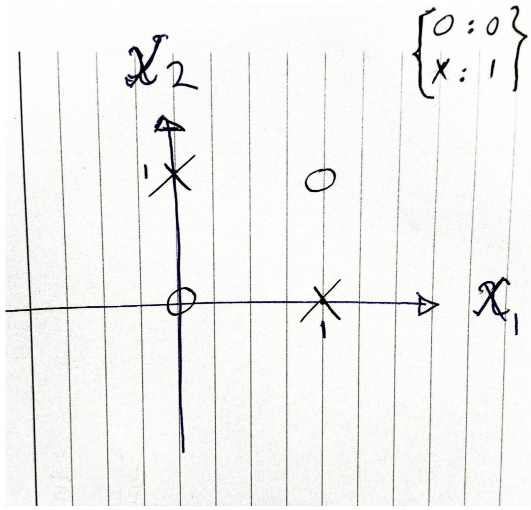

Figure 1: XOR Chart

Given this table and chart, it's not possible to separate the two different classes (output 0 and 1) with a single line in two-dimensional space. To solve this problem, we need to use multilayer neural networks or other nonlinear methods such as nonlinear activation functions like hyperbolic tangent or a combination of multilayer perceptions. Alternatively, we can add another dimension, and then we can separate it with a plane.

5

This sentence is correct. The perceptron and its learning rule are inspired by the functioning of biological neurons. In fact, the perceptron attempts to represent the functioning of biological neurons in a simplified manner; this means that instead of complex neural networks in the brain, we present a simpler model using perceptrons, which consist of simpler elements and are more understandable.

Part Two

1

We present a dataset depicted in this figure. The x-axis represents X1, and the y-axis represents X2. The violet points correspond to 'y' equaling -1, while the yellow points correspond to 'y' equaling +1.

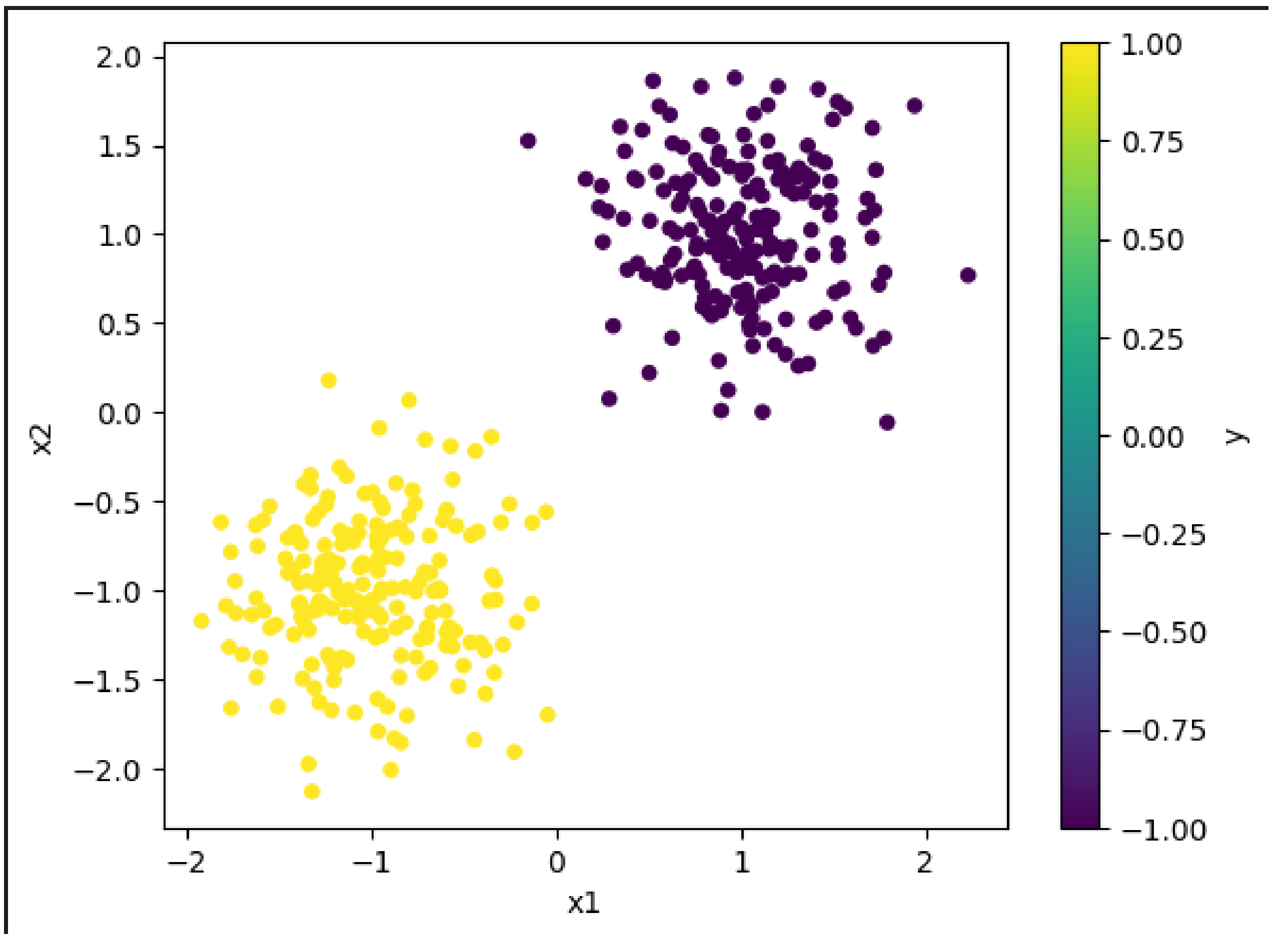

Figure 2: Dataset

2

Now, we define a perceptron neuron model and apply it to the shuffled dataset. After training, its accuracy with a learning rate of 0.01 and a threshold of 0.1 over 100 epochs is 97.5 percent. This implies that only two points are misclassified. Additionally, we can observe in this figure that these two points lie within the margin area.

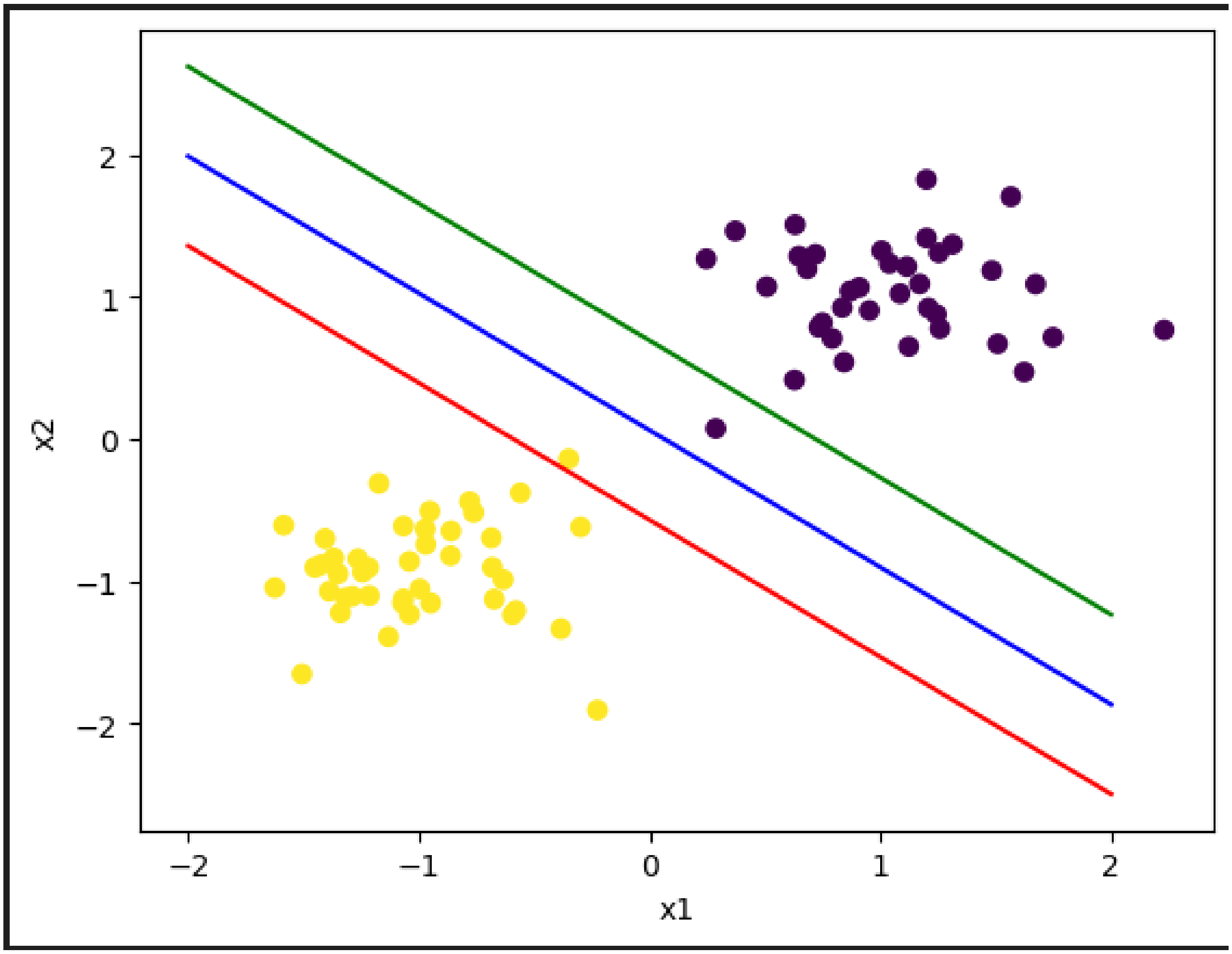

Figure 3: Result

3

Next, we repeat the previous process with a threshold of 0.001 and a threshold of 100. For the threshold of 0.001, it achieves 100 percent accuracy. However, with a threshold of 100, it yields an accuracy of 92.5 percent, indicating that six points are detected incorrectly. Therefore, we conclude that increasing the threshold decreases accuracy because the chance of points being located in the margin area increases, leading to incorrect outputs.

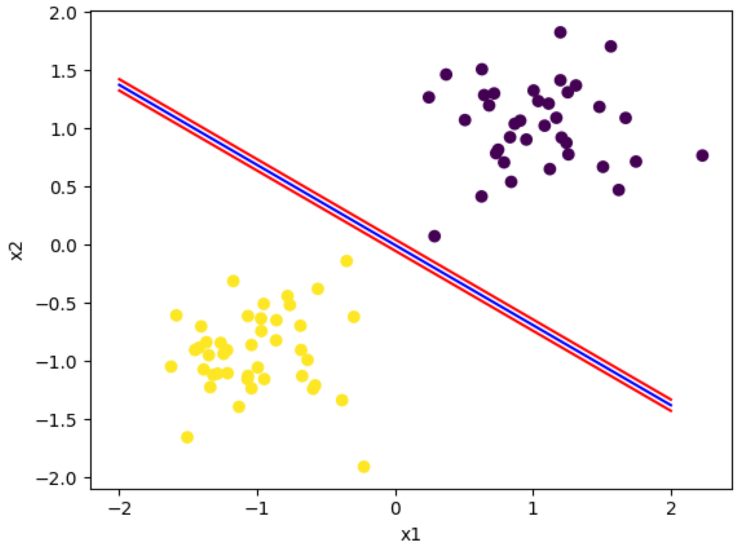

Figure 4: small threshold

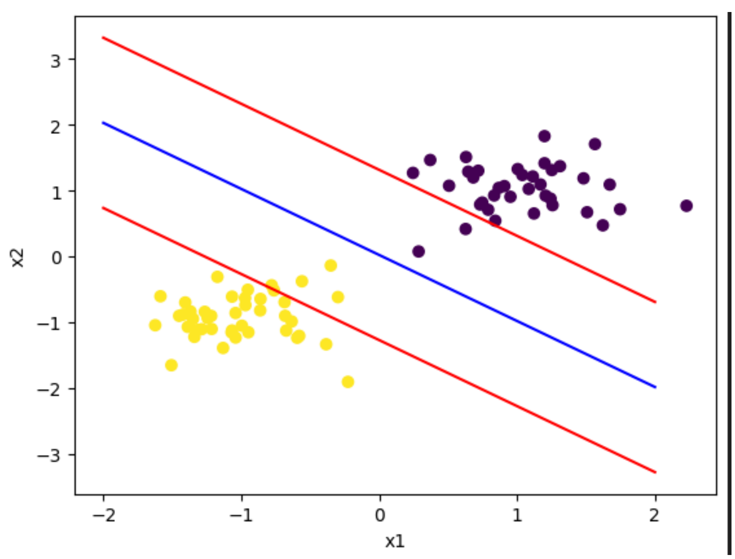

Figure 5: big threshold

Part Three

1

Firstly, we define and display the fonts. Then, we assign targets for each neuron, define the model, and create five neurons to detect each character. Subsequently, we train all the neurons using all of the fonts, followed by testing the neurons with all the fonts. The results are as follows:

Accuracy of B: 100.00%
Accuracy of C: 100.00%
Accuracy of D: 100.00%
Accuracy of L: 100.00%
Accuracy of N: 100.00%

Total accuracy: 100.00%

2

Using the given formula for calculating MSE, we obtained the following results:

Mean Squared Error of B: 1.73
Mean Squared Error of C: 1.64
Mean Squared Error of D: 2.76
Mean Squared Error of L: 2.69
Mean Squared Error of N: 1.44

We observe that for complicated characters like 'B' and 'N', the MSE is smaller than for simple characters like 'C' and 'L'. This discrepancy can be attributed to the structural complexity of the characters. Simple characters have fewer intricate details, resulting in smaller data variations, hence the error is comparatively higher.

3

Learning Rate 0.99:

- Advantages:
    1. Fast Learning Speed: A high learning rate increases the speed of learning for the algorithm as weights quickly move towards optimal values.
    2. Quick Convergence: With a high learning rate, the algorithm rapidly approaches or reaches approximately optimal weight and bias values, producing output quickly.
- Disadvantages:
    1. Far from Optimization: Often, with a high learning rate, the algorithm quickly approaches final values but may stray away from optimization and fail to reach the optimal point.

2. Large Jumps in Various Directions: A high learning rate can lead to large, discontinuous jumps in the optimization search.

Learning Rate 0.99 Result:

- Mean Squared Error of B: 30777.90
- Mean Squared Error of C: 23996.80
- Mean Squared Error of D: 33451.02
- Mean Squared Error of L: 44340.51
- Mean Squared Error of N: 33537.43

Learning Rate 0.001:

- Advantages:
  1. More Precise Optimization: A small learning rate can help the algorithm perform more precise optimization and get closer to the optimal point.
  2. Stability Across Problems: With a small learning rate, the algorithm is generally more stable and less sensitive to data fluctuations.
- Disadvantages:
  1. Slow Learning Speed: The algorithm with a small learning rate may generally converge to final values more slowly and require more time for training.
  2. Getting Stuck in Local Minima: A small learning rate may cause the algorithm to get stuck in local minima and lose more comprehensive information.

Learning Rate 0.001 Result:

- Mean Squared Error of B: 5.21
- Mean Squared Error of C: 5.66
- Mean Squared Error of D: 5.56
- Mean Squared Error of L: 4.84
- Mean Squared Error of N: 5.05

4

Finally, we train the model for 5, 10, 20, 50, 100 epochs, and for each epoch, the total error is shown in this figure. Based on the results, initially, the error is higher, but then it decreases and stabilizes. The number of epochs (training iterations) directly influences the learning capability of

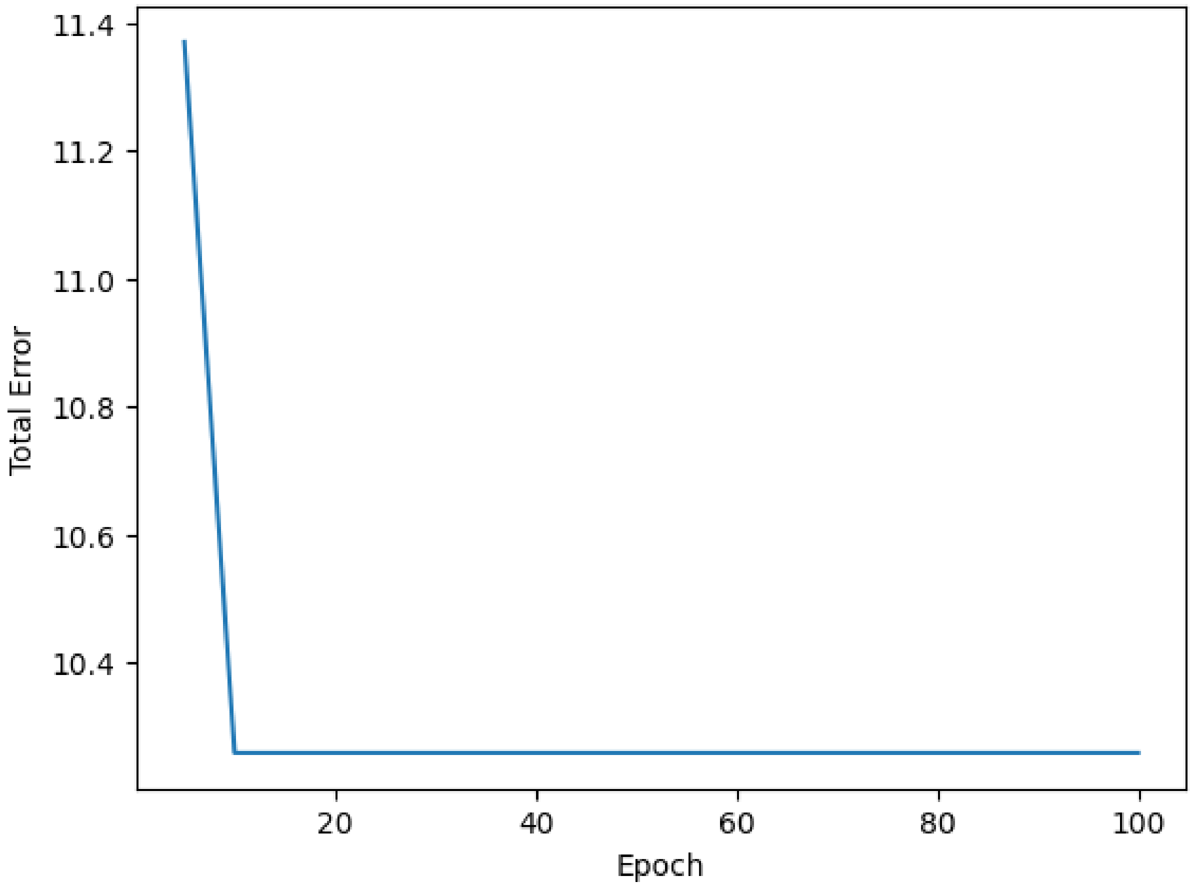

Figure 6: MSE vs Epoch

a neural network. Increasing the number of epochs may lead to improved accuracy as the network has more opportunities to learn from the training data, potentially reducing convergence delay and preventing overfitting. Conversely, reducing the number of epochs can decrease training time but may limit the network's learning potential and its ability to avoid overfitting. Ultimately, determining the optimal number of epochs requires careful experimentation and evaluation based on the dataset characteristics and network architecture.