Implementation of Logic Gates by Single Layer Perceptron

Majid Nasiri Manjili
Shahid Rajaee teacher training University, Tehran, Iran
majid.nasiri@srttu.edu

Abstract

In this report we have implemented basic logic gates (AND, OR, NAND, NOR, XOR and XNOR) only using one perceptron neuron. We also explain the classification of linear separable data using perceptron model and how single perceptron neuron cannot classify XOR/XNOR logic gates. This report is based on Machine Learning course.

Keywords: classification, perceptron neuron, XOR/XNOR problem.

1 Introduction

Perceptron is type of linear classifier that separate feature space using a straight line. When the classes are linearly separable we can draw a line to divide feature space to two isolated parts. For finding a line to separate binary and separable classes, perceptron neuron can be very useful.

2 Model

As shown in Figure 1, a single layer perceptron can be modeled by inputs(X_i), weights(W_i), summation module and unipolar step function as activation function.

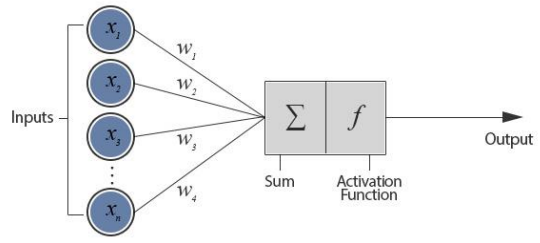

Figure 1: Single Layer Perceptron

In this single perceptron layer we should learn the model by showing the correct answers we want to generate and propagate error back in order to adjusting weights. Actually regulating weights means moving separator line in feature space in a way that finally classify samples.

3 Implementation & results

After implementation of single layer perceptron, we have trained our model to learn *AND, OR, NAND, NOR, XOR and XNOR* logic gates using their true tables. The results including separated classes using red line and error for training in each epoch are shown in Figure 2-11. In this figures sequence of light to dark blue lines and final red separator line representing the process of changes in weights to lead the model to local minimum error.

4 Conclusion

As shown in Figure 2-11 it seems that for AND, OR, NAND and NOR logic gates which we can draw a line to separate 1's and 0's output classes, the weights converges and error goes to zero, but for those XOR and XNOR logic gates that we cannot separate 1's and 0's binary classes, the weights doses not converges and minimum error between desire output and generated output will not be zero. And this misclassification in XOR/XNOR called XOR problem [1], and by this single layer perceptron is not able to classify it. The solution is using more perceptron layer to build a more complex model.

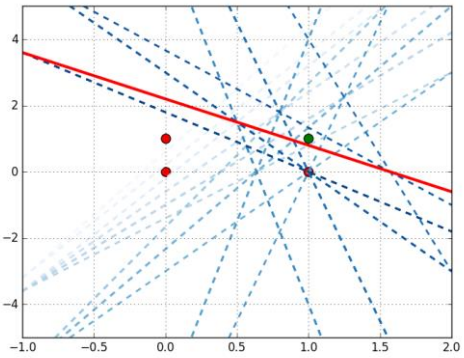

Figure 2: Classification AND logic gate by single layer perceptron

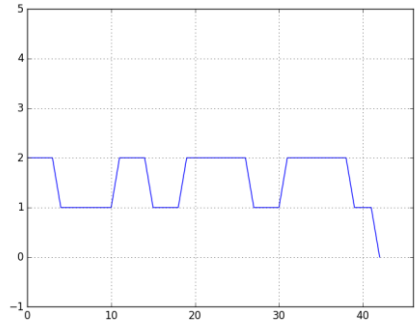

Figure 3: Classification error vs epoch number

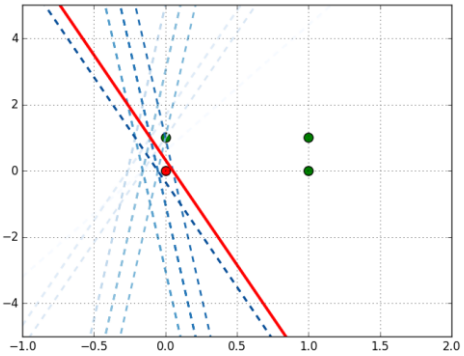

Figure 4: Classification OR logic gate by single layer perceptron

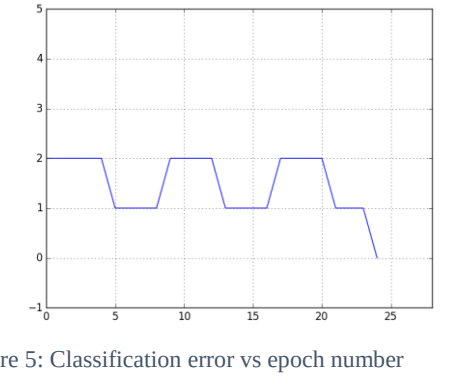

Figure 5: Classification error vs epoch number

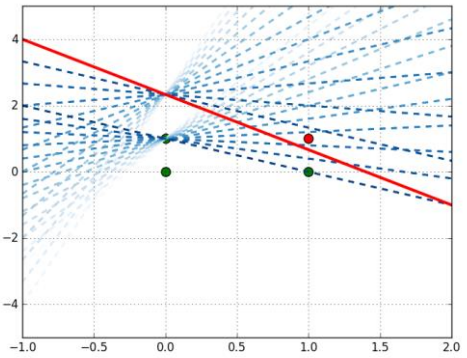

Figure 6: Classification NAND logic gate by single layer perceptron

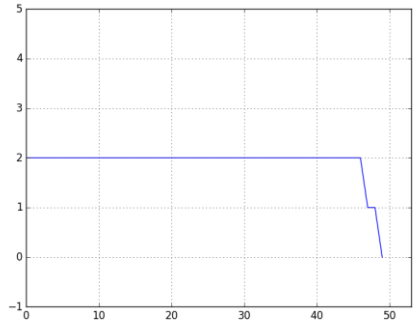

Figure 7: Classification error vs epoch number

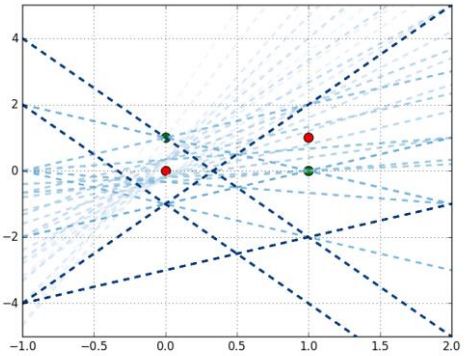

Figure 8: Misclassification XOR logic gate by single layer perceptron

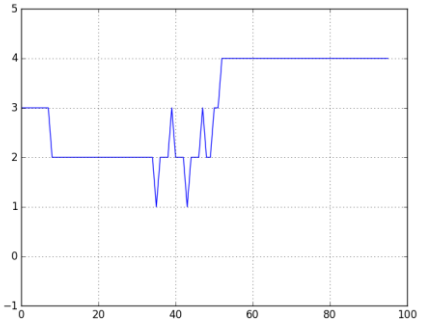

Figure 9: Classification error vs epoch number

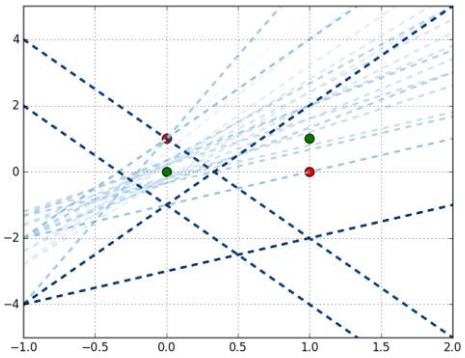

Figure 10: Misclassification XNOR logic gate by single layer perceptron

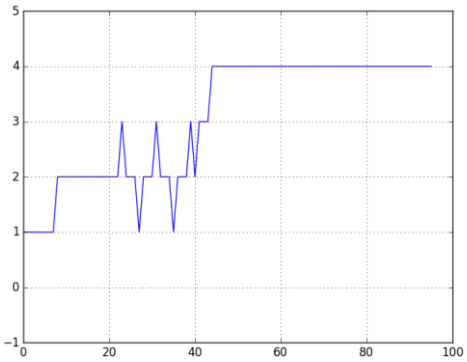

Figure 11: Classification error vs epoch number

References

[1] Nitta, Tohru. "Solving the XOR problem and the detection of symmetry using a single complex-valued neuron." *Neural Networks* 16.8 (2003): 1101-1105.

[2] Rosenblatt, Frank. "The perceptron: A probabilistic model for information storage and organization in the brain. *"Psychological review* 65.6 (1958): 386.