Artificial Neural Networks
A Simple Perceptron Implementation in Python

Kevin Costa Scaccia
Federal University of São Paulo - UNIFESP

I. Implementation

The core of the code consists of the **Perceptron** class that encapsulates all the functionality of the model. Following is a description of its operation and the implemented methods.

A. Constructor

The constructor specifies the data that will feed the model: $data$ as the input data and $target$ their respective classes. Some variables and the weight matrix of the model are initialized, including $bias$.

B. Train

This method implements the network training on the input data.

Each epoch iterates over all examples, passes through the net and readjusts the weights according to the error relative to the current example.

C. Predict

This method passes the x parameter through the network, feeding it and returning the output value already activated by the linear activation function.

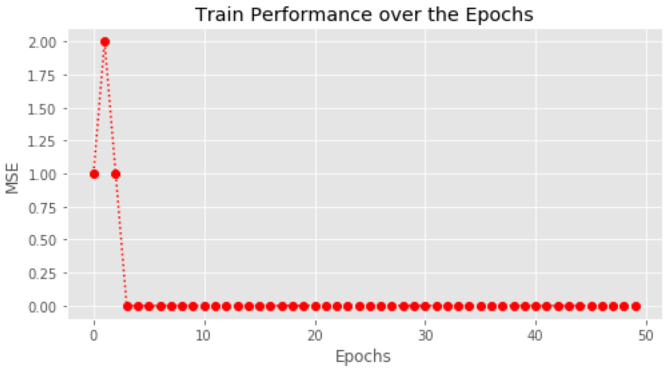
D. Accuracy

This method evaluates the accuracy of the model in the test set passed as parameter.

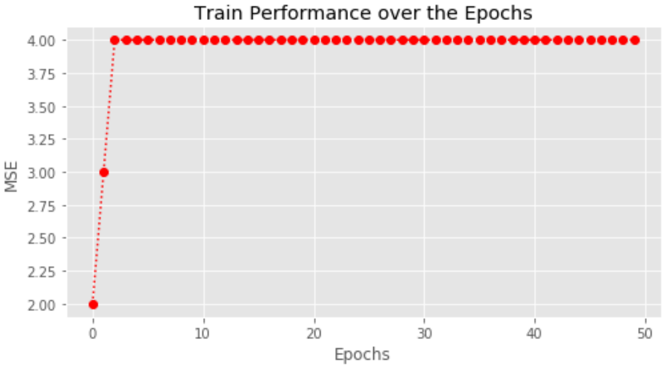
II. Experiments and Evaluation

The image below shows some measurements of **mean_squared_error** during each epoch of the model training in three different data sets.

A. OR Truth Table



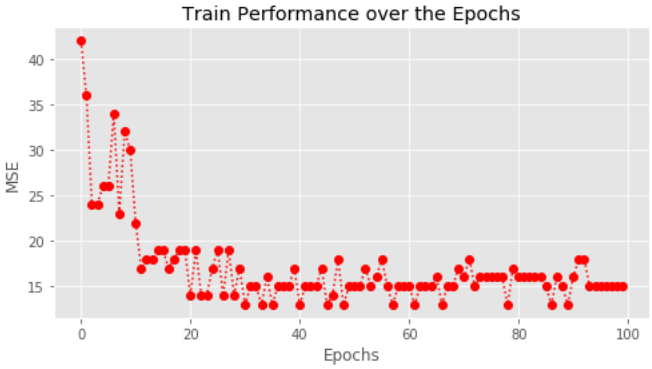
B. XOR Truth Table



C. Sklearn Breast Cancer Dataset

This is a classic dataset used to classify cells into cancerigenic or not, whether it is a problem of some complexity and clearly nonlinearly separable. Each example consists of 30 features related to the cell, followed by its class.

From the figure below we can see that the model learned and converged to a local minimum, reaching an accuracy of 64.18% in the test set (20% of the original data).


Train Performance over the Epochs
MSE
Epochs