Work Integrated Learning Programmes

BITS Pilani
Pilani | Dubai | Goa | Hyderabad

Computer Science & Information Systems

Machine Learning - Lab sheet - Module 7

Exercise 3 - Feed Forward Neural Network

1 Objective

The objective is to

- implement a feed forward neural network.
- train the feed forward neural network for OR and XOR gates.

2 Steps to be performed

Tool Python3

Libraries required numpy, matplotlib

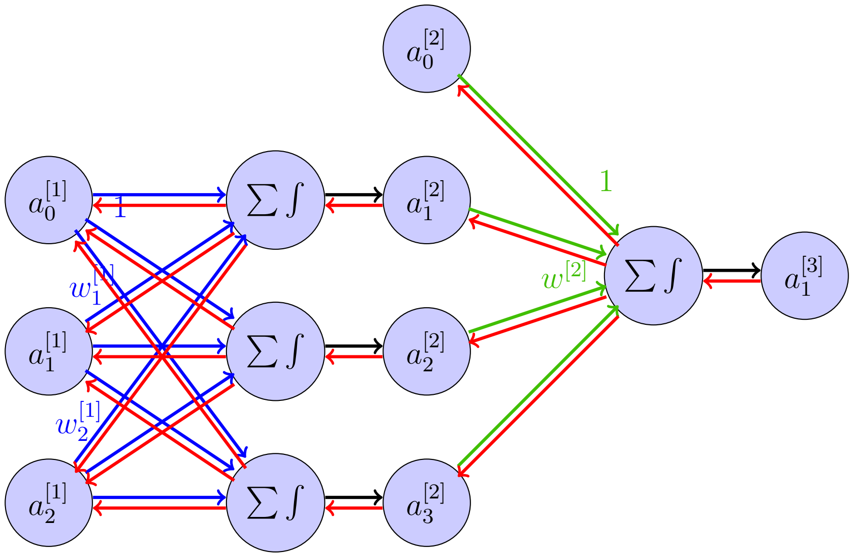
Input OR and XOR gate boolean data

Machine Learning Model Feed forward Neural Network

Implementation ML_Lab 16 Neural Network - Feed forward.ipynb

Steps .

- Import required Python libraries.
- Create the dataset as numpy arrays.
- Visualize the dataset.
- Define the neural network architecture as shown below.

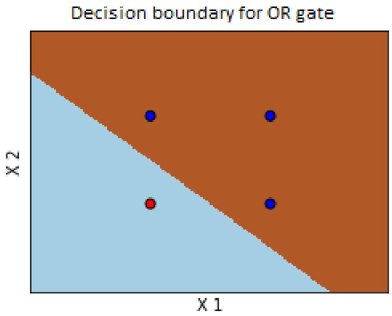


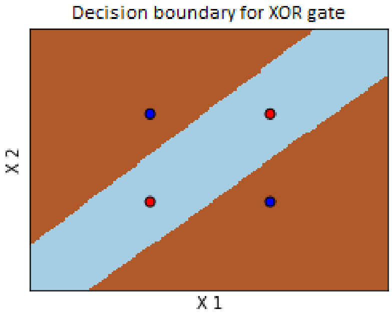
- Initialize the parameters, weights and bias, of the network.
- Sigmoid activation function
$$\sigma(z) = \frac{1}{1+e^{-z}}$$
- Derivative of sigmoid function
$$\sigma'(z) = z * (1 - z)$$
- Implement forward propagation.
  For one example x:
  For layer 1:
  - Compute the hypothesis.
  $$z^{[1]} = W^{[1]}x + b^{[1]} \tag{1}$$
  - Compute the activation
  $$a^{[1]} = \sigma(z^{[1]}) \tag{2}$$

  For layer 2:
  - Compute the hypothesis.
  $$z^{[2]} = W^{[2]}a^{[2]} + b^{[2]} \tag{3}$$
  - Compute the activation
  $$a^{[2]} = \sigma(z^{[2]}) \tag{4}$$
- Compute cost of the network.
$$J = -\frac{1}{m}\sum_{i=0}^{m}\left(y\log\left(a^{[2]}\right) + (1-y)\log\left(1-a^{[2]}\right)\right) \tag{5}$$
- Predicted values for the output layer
$$\hat{y} = \begin{cases} 1 & \text{if } a^{[2]} > 0.5 \\ 0 & \text{otherwise} \end{cases} \tag{6}$$
- Back propagation algorithm add equations here
- Update the parameters of the network using the learning rule. add equations here
$$W = W + \alpha * J \tag{4}$$
$$b = b + \alpha * J \tag{5}$$
- Visualize the decision boundary and cost function.
- Measure the performance of the model.

3 Results

- The feed forward neural network was implemented.
- The parameters that will predict the desired output for OR and XOR gates were learned.
- The decision boundary and cost were plotted for OR and XOR gates.





4 Observation

- The trained feed forward neural network predicted the OR gate outputs with 100% accuracy.
- The trained feed forward neural network predicted the XOR gate outputs with 100% accuracy.