

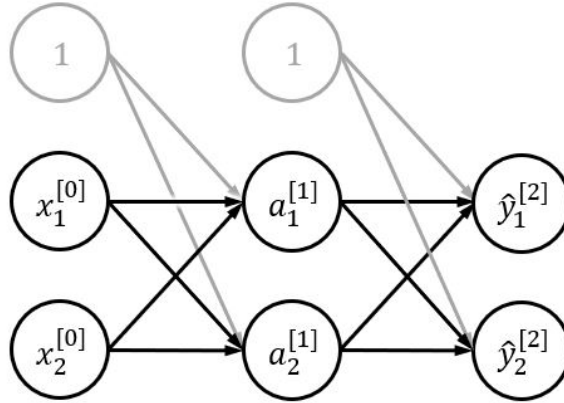
National University of Singapore
 School of Computing
 CS3244: Machine Learning
 Solution to Tutorial 07

Perceptrons and Neural Networks

Colab Notebook : Perceptrons and Neural Networks

1. **Backpropagation algorithm.** In this question, we're going to use a neural network with a **2-d input, one hidden layer with two neurons and two output neurons**. Additionally, the hidden neurons and the input will **include a bias**. We use **ReLU function** as the nonlinear activation function.

Here's the basic structure:



- (a) Suppose there is a data input $\mathbf{x} = (2, 3)^\top$ and the actual output label is $\mathbf{y} = (0.1, 0.9)^\top$. The weights for the network are

$$\mathbf{W}^{[1]} = \begin{bmatrix} 0.1 & 0.1 \\ -0.1 & 0.2 \\ 0.3 & -0.4 \end{bmatrix}, \mathbf{W}^{[2]} = \begin{bmatrix} 0.1 & 0.1 \\ 0.5 & -0.6 \\ 0.7 & -0.8 \end{bmatrix},$$

Calculate the following values after forward propagation:

$\mathbf{a}^{[1]}$, $\hat{\mathbf{y}}^{[2]}$ and $L(\hat{\mathbf{y}}^{[2]}, \mathbf{y})$.

- (b) Suppose we already know that $\frac{\partial L(\hat{\mathbf{y}}^{[2]}, \mathbf{y})}{\partial y_1^{[2]}} = 0.5$, $\frac{\partial L(\hat{\mathbf{y}}^{[2]}, \mathbf{y})}{\partial y_2^{[2]}} = 0.3$, $a_1^{[1]} = 0.5$, $a_2^{[1]} = 0.4$, $\hat{y}_1^{[2]} > 0$, $\hat{y}_2^{[2]} > 0$. Calculate the following gradient (partial derivative):
 $L(\hat{\mathbf{y}}^{[2]}, \mathbf{y})$ with respect to $W_{21}^{[2]}$ and $L(\hat{\mathbf{y}}^{[2]}, \mathbf{y})$ with respect to $W_{12}^{[2]}$.

2. Perceptrons

- (a) Model AND, OR, and NOT logic functions using a perceptron. Assume AND, and OR functions take 2 inputs while the NOT function takes a single input. Additionally, is it possible to model XOR function using a single Perceptron? Comment on your answer.
- (b) Model XOR function (takes 2 inputs) using a number of perceptrons which implement AND, OR, and NOT functions. Show the diagram of the final Perceptron network. Clearly specify the weights of your network.
- (c) Can the following function in Figure 1 be expressed with a 3-layer perceptron?

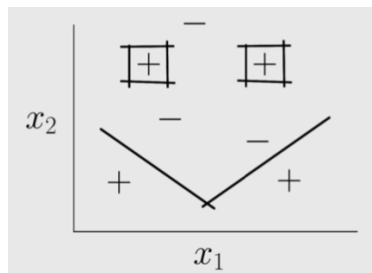


Figure 1: Function