Notations and Formulation for a Single Neuron

Harry Li ‡, Ph.D.
Department of Computer Engineering, College of Engineering
San Jose State University, San Jose, CA 95192, USA

***Abstract*—This note is for notations and mathematical formulation for a single neuron.**

I. Introduction

In this note, we describe the notations and mathematical formulation for a single neuron. A single neuron is illustrated in the figure below.

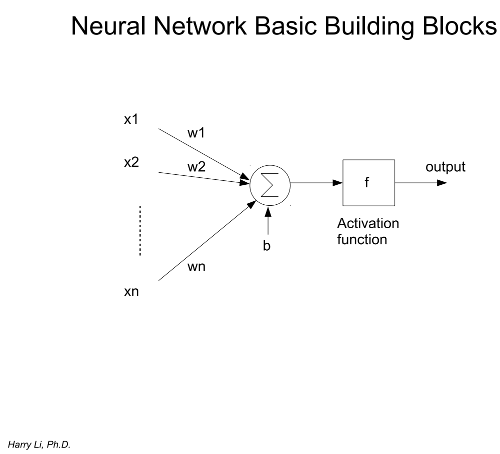

Fig. 1. A single neuron.

II. Notation for Inputs

Notation for a neuron input x_i, $i = 1, 2, ..., N$ is written as

$$\{x_i | i = 1, 2, ..., N\} \tag{1}$$

and its vector form is

$$(x_1, x_2, ..., x_N) \tag{2}$$

or simply denoted as X.

Now, introduce a superscript j for experiment j. The input is x_i^j, and $i = 1, 2, ..., N$ and $j = 1, 2, ..., P$.

$$\{x_i^j | i = 1, 2, ..., N; j = 1, 2, ..., P\} \tag{3}$$

III. Notation for Weights

Notation for a weight w_i, $i = 1, 2, ..., N$ is written as

$$\{w_i | i = 1, 2, ..., N\} \tag{4}$$

and its vector form is

$$(w_1, w_2, ..., w_N) \tag{5}$$

or simply denoted as W.

Now, introduce a superscript j for experiment j. The input is w_i^j, and $i = 1, 2, ..., N$ and $j = 1, 2, ..., P$.

$$\{w_i^j | i = 1, 2, ..., N; j = 1, 2, ..., P\} \tag{6}$$

IV. Inputs and Weights

For a single input x_i, it reaches the neuron with weight (synapse) w_i, e.g.,$w_i x_i$ for $i = 1, 2, ..., N$. So, for all the inputs, we have

$$(w_1, w_2, ..., w_N) \cdot (x_1, x_2, ..., x_N) = w_1 x_1 + w_2 x_2 + ...w_N x_N \tag{7}$$

Or equivalently

$$(x_1, x_2, ..., x_N) \cdot (w_1, w_2, ..., w_N) = w_1 x_1 + w_2 x_2 + ...w_N x_N \tag{8}$$

And

$$\sum_{i=1}^{N} w_i x_i = w_1 x_1 + w_2 x_2 + ...w_N x_N \tag{9}$$

Or simply in a short hand vector form notation:

$$\sum_{i=1}^{N} w_i x_i = W \cdot X. \tag{10}$$

V. A Transfer Function

A transfer function is defined as

$$h = \sum_{i=1}^{N} w_i x_i = W \cdot X + b \tag{11}$$

and it is denoted as h, or $h(\cdot)$, or $h(w_i, b)$ depending on the context of discussion. See the illustration below for the transfer function.

We define a transfer function to denote the inputs and weights interaction, e.g.,

$$(w_1, w_2, ..., w_N) \cdot (x_1, x_2, ..., x_N) = w_1 x_1 + w_2 x_2 + ...w_N x_N \tag{12}$$

Or equivalently

$$(x_1, x_2, ..., x_N) \cdot (w_1, w_2, ..., w_N) = w_1 x_1 + w_2 x_2 + ...w_N x_N \tag{13}$$

And

$$\sum_{i=1}^{N} w_i x_i = w_1 x_1 + w_2 x_2 + ...w_N x_N \tag{14}$$

Or simply in a short hand vector form notation:

$$\sum_{i=1}^{N} w_i x_i = W \cdot X. \tag{15}$$

which are now simply denoted as a transfer function

$$h = \sum_{i=1}^{N} w_i x_i = W \cdot X + b. \tag{16}$$

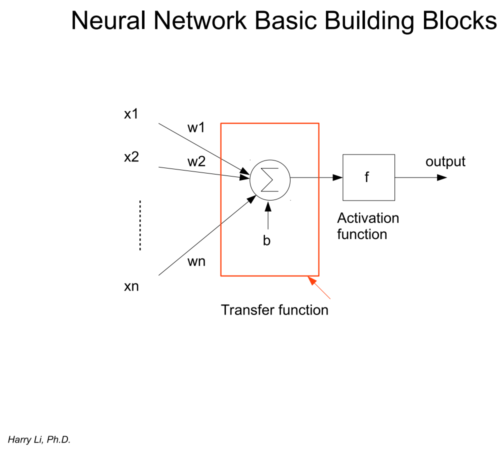

Fig. 2. A transfer function.

VI. Activation Function

An activation function denoted as $f(\cdot)$, which acts like a "switch" to turn on or off the neuron output, or it can "attenuate" the neuron output in case not totally turn the neuron on or off.

The activation function is illustrated in the figure above, and it is defined as

$$y = f(\sum_{i=1}^{N} w_i x_i = W \cdot X + b). \tag{17}$$

where y is the output of the neuron, and the activation function can be rewritten as

$$y = f(\sum_{i=1}^{N} w_i x_i = W \cdot X + b) = f(h(w_i, b)). \tag{18}$$

Or simply written as

$$y = f(h(\cdot)) = f(h(w_i, b)). \tag{19}$$

which can be used in the analysis and derivation of multi-layer behavior of a neural networks.

For more than one output of a neural network with more than one neurons put together as shown in the figure below (multi-layer feed forward neural networks), we have

$$y_i = f_i(h(\cdot)) = f_i(h(w_i, b)) \tag{20}$$

where $i = 1, 2, ..., M$.

Now for experiment j, $j = 1, 2, ..., P$, we denote the output as

$$y_i^j \tag{21}$$

where $i = 1, 2, ..., M$ and $j = 1, 2, ..., P$.

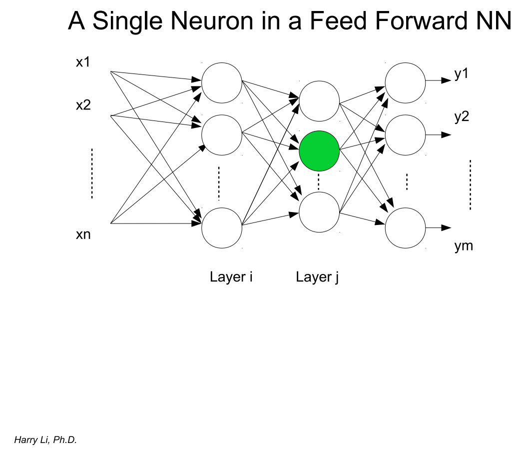

Fig. 3. Feed forward Neural Network.

VII. Loss Function

When experiment is performed for training purpose, input $(x_1, x_2, ..., x_N)$ and ground truth y are all provided. Let's now take a look at an output of a neuron at experiment j, we denote this experiment output as $\tilde{y}^j$, so with the ground truth output y^j, we can define a loss function L as follows

$$L = \tilde{y}^j - y^j. \tag{22}$$

The acumulated loss functions for all experiments j for $j = 1, 2, ..., P$ is

$$L_{total} = \frac{1}{2} \sum_{j=1}^{P} (\tilde{y}^j - y^j)^2. \tag{23}$$

Very often we just drop the subscript $total$ of the loss function with the understanding of the total accumulative loss.

VIII. Minimize the Loss Function

$$\frac{\partial L}{\partial w_{i,k}} = \frac{\partial}{\partial w_{i,k}} \frac{1}{2} \sum_{j=1}^{P} \sum_{i=1}^{M} (\tilde{y}_i^j - {y_i}^j)^2 \tag{24}$$

References

[1] *Harry Li, Lecture Notes on Deep Learning*, 2021.