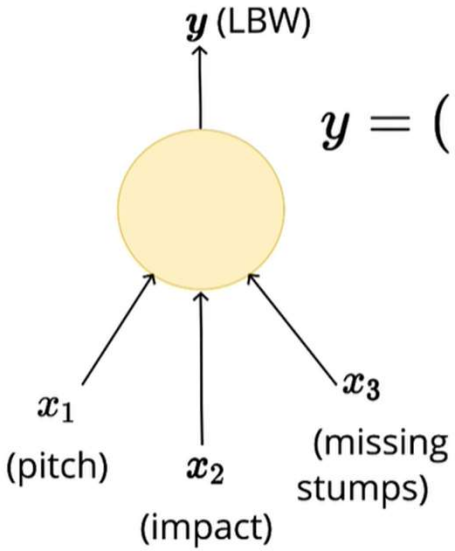
Deep Learning : McCulloch Pitts Neuron (Walter Pitts & Warren McCulloch)

राष्ट्रीय प्रौद्योगिकी संस्थान सिक्किम
NATIONAL INSTITUTE OF TECHNOLOGY SIKKIM

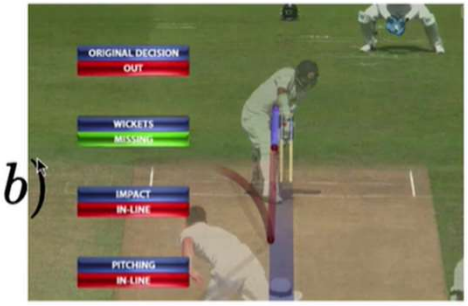
Course Instructor:
Dr. Bam Bahadur Sinha
Assistant Professor
Computer Science & Engineering
National Institute of Technology
Sikkim

MP Neuron a.k.a Linear Threshold Gate

Fundamental building block of deep learning.

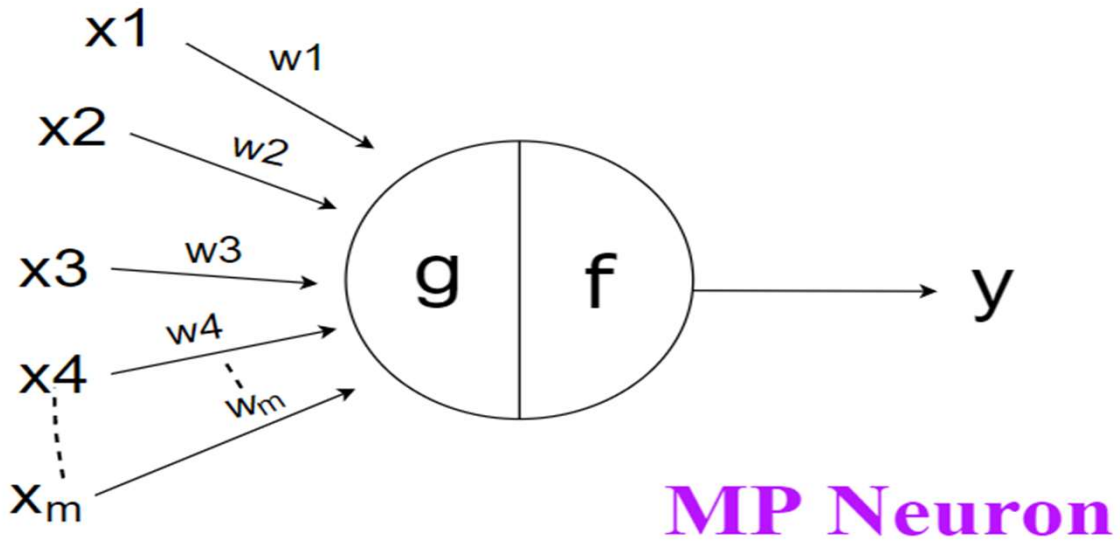


$$y = \left(\sum_{i=1}^{3} x_i \geq b\right)$$



- Boolean Input
- Boolean Output

The Model



$g(x1,x2,x3\ldots x_n) = g(x) = \sum_{i=1}^{n} x_i$

$Y = f(g(x))$

$Y = f(g(x)) = 1$, if $g(x) >= b$

$Y = f(g(x)) = 0$, if $g(x) < b$

Note: This is a general architecture of neurons. MP-Neuron doesn't have weights associated with it.

Data

Pitch In-Line	Impact	Missing Stumps	Is it LBW?
1	0	0	0
0	1	1	0
1	1	1	1
0	1	0	0

Input: Boolean
Output: Boolean

How to handle Non-Boolean Input ?

Eg:
Product launched 16 days back
Product launched 2 weeks back

Whether launched (within 6 months): Yes (1) || No (0)

	P1	P2	P3	P4
Launch (within 6 months)	0	1	1	0
Weight (<160 gm)	0	0	1	1
Screen Size (<5.9 inch)	1	1	0	0
Dual Sim	0	0	1	1
Price > 20K	0	0	1	1
Like Prediction (Y)	0	0	0	1

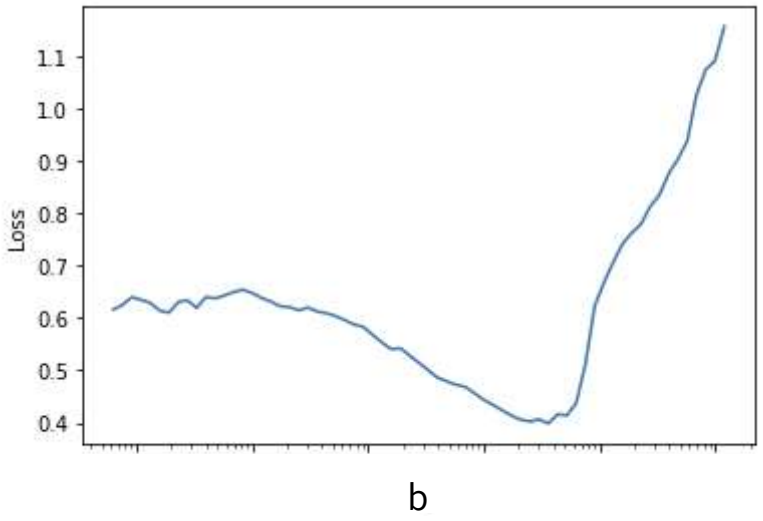
Loss Function

- Loss/ Error = True – Predicted : Basic Idea
- Loss/ Error = $\sum(True - Predicted)$: Loss for all 'n' points
- Loss/Error = $\sum(True - Predicted)$^2 : Square of difference
- Loss/ Error = |True - Predicted| : Absolute Difference

Learning Algorithm

- $\boldsymbol{Y} = \sum_{i=1}^{n} x_i \geq b$: *Model*
- $\mathbf{L} = \sum_i (A_i - Y_i)^2$: *Loss*

Can afford Brute Force: Since only parameter is there - b



Evaluation

- To determine the model performance

$$\text{Accuracy} = \frac{Number\ of\ Correct\ Predictions}{Total\ Number\ of\ Predictions}$$

Implementation of MP – Neuron using Python