College of Computer Science and Engineering

Department of Computer Science and Artificial Intelligence

CCAI-321: Artificial Neural Networks

Lab#2 Building a multiple input Neuron using Python

PLO = S1 - AI

Marks:

|  |  |  |  |
| --- | --- | --- | --- |
|  | Q1 | Q 2 | Total |
| Allocated | 6.5 | 3.5 | 10 |
| Obtained | 6.5 | 3.5 | 10 |
|  |  |  |  |
| Allocated |  |  |  |
| Marks |  |  |  |

Weighted Marks:

|  |  |  |  |
| --- | --- | --- | --- |
| Allocated |  |  |  |
| Obtained |  |  |  |

Objectives

* Implement a single input neuron in python
* Implement a multiple input neuron in python

Lab Tool(s)

[Download Python | Python.org](https://www.python.org/downloads/)

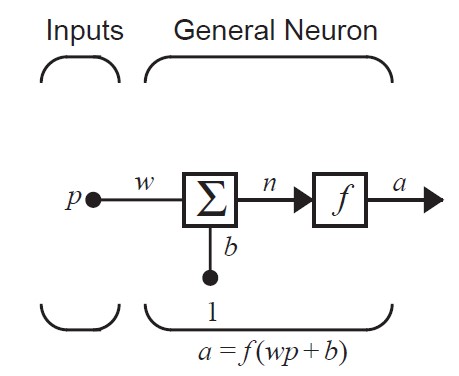
[Anaconda | Individual Edition](https://www.anaconda.com/products/individual)

Lab Deliverables

Submit a pdf document on Blackboard containing your solution to the lab assessment at the end of this document.

What is a Neuron?

A simple Artificial Neuron has a summer and an activation (transfer) function connected in series as shown in the figure below. For a single input neuron, shown in the figure, there is a single input ‘p’ weighted by a weight ‘w’ added to a bias value ‘b’. The result of summing these quantities is given as ‘n=wp+b’. This value is passed to a transfer function ‘f’ to obtain an output ‘a’ of the neuron in a predefined desired range.



We discussed various transfer functions in the course and implemented them in the previous lab. In this lab we shall use those functions to develop a working artificial neuron. The different transfer functions are presented below for reference.

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Description | Shape | Comments |
| Hardlimit | 1. 𝑓𝑜𝑟 𝑛 < 0 ℎ𝑎𝑟𝑑𝑙𝑖𝑚(𝑛) = { 2. 𝑒𝑙𝑠𝑒 |  | The function is not continuous as it jumps from zero to one. |
| Linear | 𝑝𝑢𝑟𝑒𝑙𝑖𝑛(𝑛) = 𝑛 |  | The output of the function keeps on increasing and never  clips to any value |
| Logsigmoid | 1  𝑙𝑜𝑔𝑠𝑖𝑔(𝑛) = |  | Continuous and clipped between 0 and 1. |
| Hyperbolic Tangent | 𝑒𝑛 − 𝑒−𝑛  ℎ𝑡𝑎𝑛(𝑛) = 𝑛 + 𝑒−𝑛  𝑒 |  | Continuous and clipped between -0.5 and 0.5 |
| Positive Linear | 0 𝑓𝑜𝑟 𝑛 < 0  𝑝𝑜𝑠𝑙𝑖𝑛(𝑛) = {  𝑛 𝑒𝑙𝑠𝑒 |  | Continuous yet clipped at 0 and linear for +ive values a = max(n,0) |

Q1. Develop a single input neuron. Assuming that it has an input p, weight w, bias b, summation output

n, transfer function f and output a. [2+3+1.5=6.5 marks]

The steps for implementation require you to first calculate wp

Two numbers can be multiplied in python using the following command 4\*6 or for variables x\*y Next calculate the sum of wp and b and this is the output of the summation i.e. n=wp+b Finally use this as input to the transfer function to get your result.

Please create a function titled single\_input\_neuron and pass the parameters i.e. values of ‘p’, ‘w’, ‘b’ and ‘f’ i.e. name of transfer function and return the output ‘a’ from that function. [2 marks]

Complete the table given below by passing the function values of p, w, b and name of transfer function

and sharing the result a. [3 marks]

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Input ‘p’ | Weight ‘w’ | Bias ‘b’ | Transfer Function ‘f’ | Output ‘a’ |
| 2 | -1.5 | 3 | hardlim |  |
| 2 | -1.5 | 3 | logsigmoid |  |
| 2 | -1.5 | 3 | poslin |  |

Check if each of the three answers that you have given above are correct? [1.5 marks]

Q2. Develop a multiple input neuron. Assuming that it can take as input a column vector p , weight matrix w, bias vector b and produces a summation output n which when passed to the transfer function f

generates the output a. [2+1+0.5=3.5 marks]

The steps for implementation require you to first calculate wp. This time both w and p are either vectors

𝟏

or matrices i.e. 𝐩 = [𝟐] 𝐰 = [𝟒 𝟓 𝟔] assuming that p and w are vectors

𝟑

Matrix or vector multiplication can be done in python using numpy library function dot i.e. np.dot(w,p)

(Note: Inputs are provided as arrays and the transpose is taken by python itself)

Next calculate the sum of wp and b and this is the output of the summation i.e. n=np.dot(w,p)+b Finally use this as input to the transfer function to get your result.

Please create a function titled neuron and pass the parameters i.e. values of ‘p’, ‘w’, ‘b’ and ‘f’ i.e. name of transfer function and return the output ‘a’ from that function. [2 marks]

What the output ‘a’ of the multiple input neuron if p=[1 2 3], w=[4 5 6], b=-1.5 and transfer function is

‘poslin’? [1 marks]

Check if the answer that you have given above is correct? [0.5 marks]