

# Neural networks and Deep Learning <sup>★</sup>

Mohamed Hesham Ibrahim Abdalla<sup>1[0000–1111–2222–3333]</sup>, Second Author<sup>1[1111–2222–3333–4444]</sup>, and Third Author<sup>1[2222–3333–4444–5555]</sup>

German University in Cairo, Cairo, Egypt

**Abstract.** The abstract should briefly summarize the contents of the paper in 15–250 words.

**Keywords:** Neural Networks · Deep Learning · Machine Learning

## 1 Introduction

Artificial Neural Networks (ANNs) and perceptrons are intelligent units that has taken inspiration from biology, especially the brain (cite). ANNs work by taking labeled inputs and then trying to find a mathematical rule or function to systematically answer the question of which label belongs to which input, and later identify labels of new inputs that have never been seen before by the network. For example, inputs could be human images and the labels are the gender of the human in that particular image.

The history of ANNs and perceptrons goes back to the 50's and the 60's when the first known perceptron was created. The first perceptron was simulated on an IBM 704 computer at Cornell Aeronautical Laboratory in 1957 (cite). It works by giving(cont)

## 2 Mathematical Background

### 2.1 Forward Propagation

ANNs consist of neurons which are connected via links. Each neuron gets an input and produces an output as a return. Each input is given a certain weight which makes that specific input a more priority in controlling the output of the neuron. Neurons calculate their output by multiplying their inputs by their weights and applying a bias to the multiplication. Equation 1 shows a linear mapping of a single input.

$$z^i = w^T * x^i + b \quad (1)$$

where  $z^i$  is the linear mapping of the  $i$ th example,  $w^T$  in  $\mathbb{R}^{1 \times N}$  is the weight vector of the form  $[w_1, w_2, \dots, w_N]$  and  $x^i$  in  $\mathbb{R}^{N \times 1}$  is the input vector of the form  $[x_1, x_2, \dots, x_N]$  of the  $i$ th example.

---

<sup>★</sup> Supported by organization x.

This mapping is then forwarded to an activation function (Equation 2).

$$a^i = g(z^i) \quad (2)$$

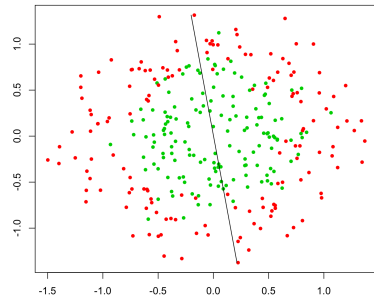
where  $a^i$  is the output of the activation function on the linear mapping of the  $i$ th example.

The choice of the activation function vary depending on the given data (inputs). In general, there are three main categories of the activation functions: binary, linear and non-linear. Binary or threshold functions output a binary value depending on the input. For example, the step function produces a +1 in case  $z^i$  is greater than or equal to 0 and -1 otherwise (Equation 3). Binary functions can not deal with categorical data, therefore they are not widely used.

$$f(x) = \begin{cases} +1 & z^i \leq 0 \\ -1 & \text{otherwise} \end{cases} \quad (3)$$

Another type of activation functions is linear. Linear functions forward the input directly to the output without any transformation. This is useful in problems where the output is continuous, for example, predicting house prices.

Although all of the previous functions are useful for some situations, they fail to find a pattern if the data is non-linearly separable since the function will only be able to draw a linear decision boundary that can divide the data into two groups. Therefore, other functions were used to find non-linear separations between the data such as sigmoid, ReLU and tanh/ Hyperbolic Tangent.



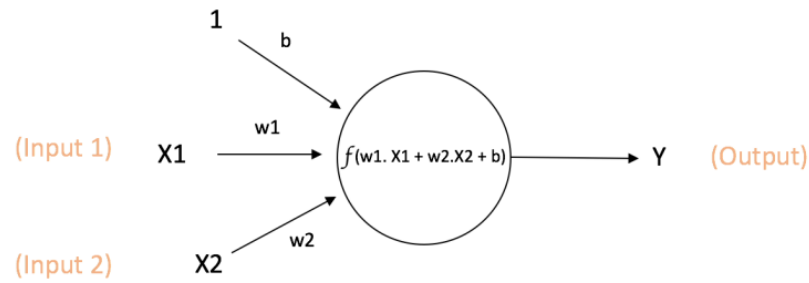
**Fig. 1.** Example of a linear activation function on non linearly separable data <https://www.r-bloggers.com/interactive-visualization-of-non-linear-logistic-regression-decision-boundaries-with-shiny/>

## References

1. Author, F.: Article title. Journal **2**(5), 99–110 (2016)

**Table 1.** Table captions should be placed above the tables.

Function name	Formule	Graph
Title (centered)	<b>Lecture Notes</b>	lol
1st-level heading	<b>1 Introduction</b>	12 point, bold
2nd-level heading	<b>2.1 Printing Area</b>	10 point, bold
3rd-level heading	<b>Run-in Heading in Bold.</b> Text follows	10 point, bold
4th-level heading	<i>Lowest Level Heading.</i> Text follows	10 point, italic



$$\text{Output of neuron} = Y = f(w1.X1 + w2.X2 + b)$$

**Fig. 2.** <https://ujjwalkarn.me/2016/08/09/quick-intro-neural-networks/>

2. Author, F., Author, S.: Title of a proceedings paper. In: Editor, F., Editor, S. (eds.) CONFERENCE 2016, LNCS, vol. 9999, pp. 1–13. Springer, Heidelberg (2016). <https://doi.org/10.1007/1234567890>
3. Author, F., Author, S., Author, T.: Book title. 2nd edn. Publisher, Location (1999)
4. Author, A.-B.: Contribution title. In: 9th International Proceedings on Proceedings, pp. 1–2. Publisher, Location (2010)
5. LNCS Homepage, <http://www.springer.com/lncs>. Last accessed 4 Oct 2017