

# On the Functionality of the Multilayer Perceptron

Landon Buell

18 May 2020

## Introduction

In 1943, neurophysiologist Warren McCulloch and mathematician Walter Pitts published *A Logical Calculus of the Ideas Immanent in Nervous Activity* [6], which paved the way for the fields of computational neuroscience and early mathematical models of the brain. McCulloch and Pitts proposed that one could loosely mimic the behavior of a brain as a series of mathematical calculations. They realized that the smallest known element of the brain, a *neuron*, acts as a simple on/off switch, much like computers have been designed to. Furthermore, single neurons by themselves do very little, but instead it is the organization of these cells into collections that make up functional networks [1, 4, 6].

The mathematical model could be augmented by trading the simple Boolean (T/F) value, with a continuously ranging number referred to as an *activation*. By producing different types of artificial neurons and establishing a series of synapses, it then became apparent that different organization of these elements could produce networks of solving increasingly advanced problems [2]. To further expand upon this, in 1957, Frank Rosenblatt created a neural network architecture called a Multilayer Perceptron, which uses layers of interacting neurons to create a single mathematical model [1, 7]. This gives it a simple feed-forward design which has experimentally shown to perform well with tasks related to image and speech recognition [2, 5].

## Mathematical Treatment

To build a Multilayer Perceptron (MLP), artificial neurons need to be grouped into multiple structures, each called a *layer*. A single layer is modeled mathematically as a column vector,  $\vec{x}$  by convention:

$$\vec{x}^{(l)} = [x_0, x_1, x_2, \dots, x_{n-2}, x_{n-1}]^T \quad (1)$$

The superscript ( $l$ ) is used to indicate the layer number of which this vector represents. Each element inside this vector is a neuron in the MLP model, and the value of that element is the neuron's *activation*.

## Feed-Forward System

## Back Propagation System

## Conclusion

## References

- [1] Géron Aurélien. *Hands-on Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*. O'Reilly, 2017.
- [2] Goodfellow, Ian, et al. *Deep Learning*. MIT Press, 2017.
- [3] James, Gareth, et al. *An Introduction to Statistical Learning with Applications in R*. Springer, 2017.
- [4] Levine, Daniel S. *Introduction to Neural and Cognitive Modeling*. 3rd ed., Routledge, 2019.
- [5] Loy, James , *Neural Network Projects with Python*. Packt Publishing, 2019
- [6] McCulloch, Warren S., and Walter Pitts. “A Logical Calculus of the Ideas Immanent in Nervous Activity.” *The Bulletin of Mathematical Biophysics*, vol. 5, no. 4, 1943, pp. 115–133.
- [7] Petrik, Marek. “Introduction to Deep Learning.” Machine Learning. 20 April. 2020, Durham, New Hampshire.