TREVOR ANDRUS

Neural Networks

Outline

HISTORY

-A brief discussion of the history of neural newtorks and how they have progressed

THEORY

-The ideology behind neural networks: why they were created and how they work

ELEMENT BREAKDOWN

-A detailed look at each element of neural nets

FUNCTIONAL EXMAPLE

-A walkthrough of a simple neural network in Jupyter Notebooks

History

INITIAL PROPOSAL
CYCLICAL RESURGENCE

Initial Proposal

- The initial concepts of neural networks were proposed by Alexander Bain in 1873 and William James in 1890. As philosopher and a psychologist respectively, they were trying to understand brain process (independent of anything computational)

- Computational Neurons based on mathematics weren't constructed for nearly 50 years. In 1944 two researchers for the University of Chicago, Warren McCullough and Walter Pitts, formed the first steps towards a computational neural network in 1944. (Perceptron)

-Despite their advances, Alan Turing is attributed to creating the first comprehensive view of computational neural networks in his 1948 paper *Intelligent Machinery.*

- However, the work done by McCullough and Pitts allowed a split in ideology with regards to neural nets - one focusing on the biological processes, and the other focusing on applications in artificial intelligence.

Cyclical Resurgence

- After McCullough and Pitts' research was published, neural networks remained a major area of research in both computer and neural science until about 1969. At that point, the conceptual value of the model was recognized, but the application wasn't wholly possible with the technology of the era.

-After a decade of near silence in mainstream research, Neural Networks returned to popularity in the 80's, only to fall back under the radar near the turn of the century.

- Recently, after nearly 70 years from it's initial emergence, it seems neural nets are again gaining popularity, spurred by the increased computational capability of current machines. It seems that the full potential of this model has yet to be realized even currently.

Theory

A CONCEPTUAL EXPLANATION
DESIGN STRUCTURE

Conceptual Explanation

- Neural nets were initially designed to mimic human brain architecture in an attempt to create models that could solve tasks which more conventional algorithms failed.

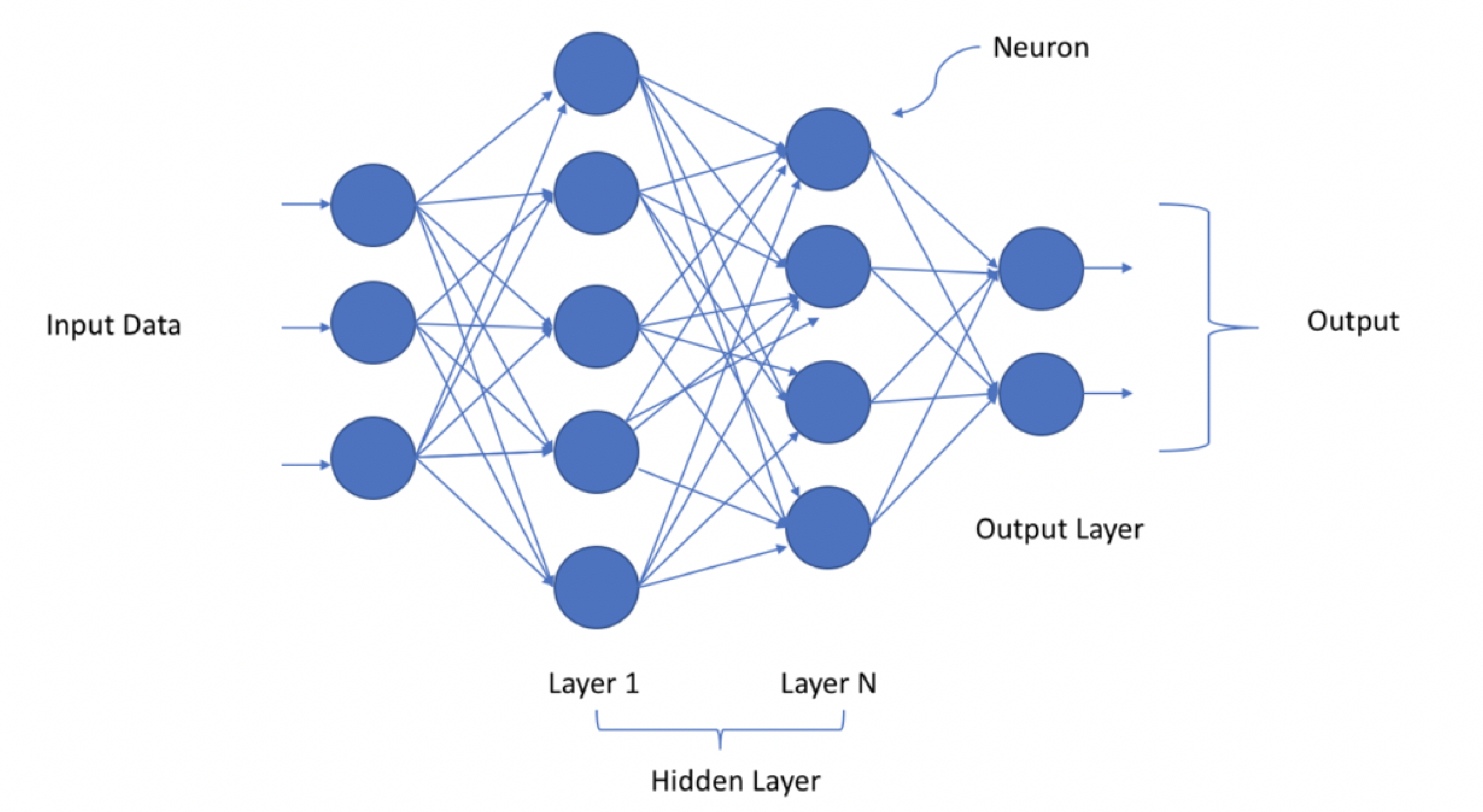
- As such, the model contains artificial neurons to simulate biological neurons, and connections between the neurons to simulate brain synapses.

- These connections are the paths through which data bases from one neuron to another, eventually exiting the model as output.

- However as time passed, the models became more geared toward empirical results, abandoning the strict adherence to biological structure.

08

Design Structure



NEURAL NETWORKS

Element Breakdown

NEURONS
ACTIVATION FUNCTIONS
CONNECTIONS / WEIGHTS
PROPAGATION FUNCTIONS
LAYERS
OVERVIEW
EXTRAS

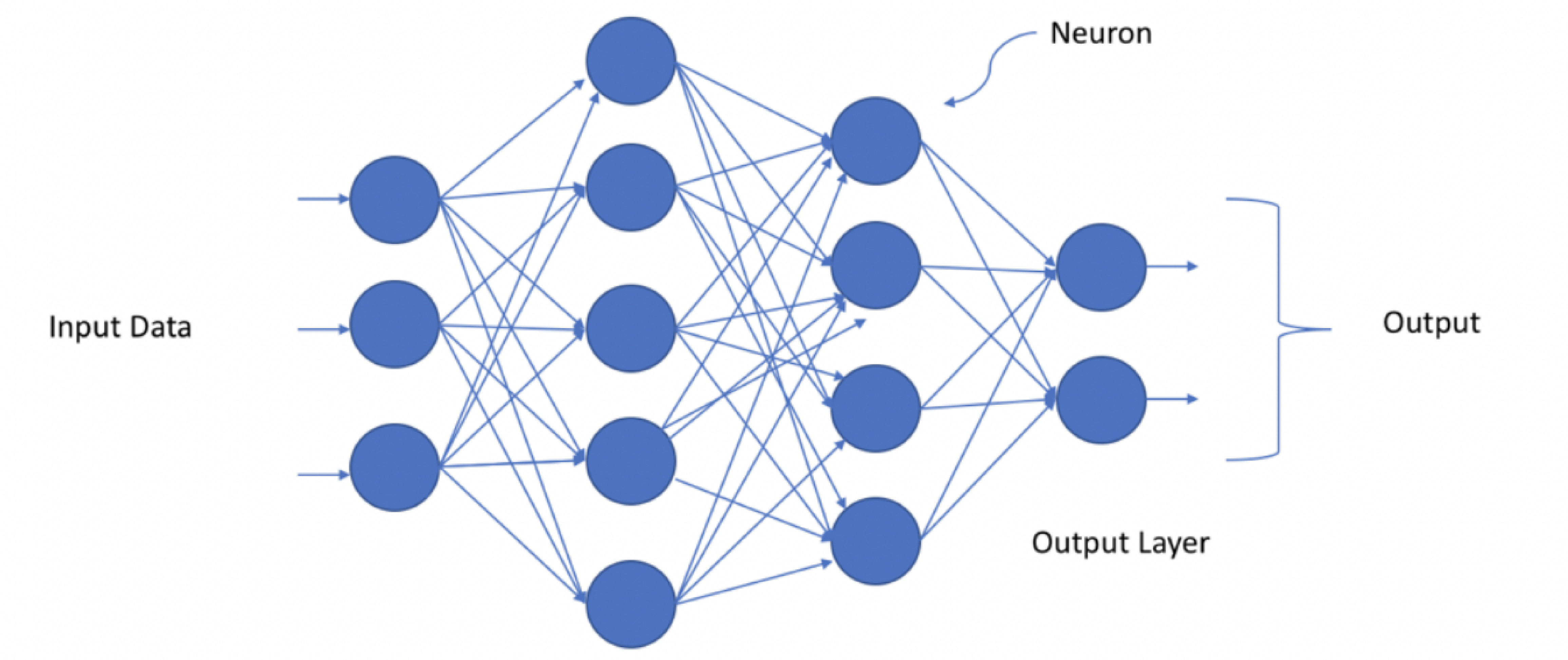
Neurons

- Neurons (or nodes) transmit and alter information.

- Each recieves one (or multiple) input(s), assesses their value based on an activation function, and can transmit a transformed version as a single output.

-Types of Nodes: Input, Output, Hidden

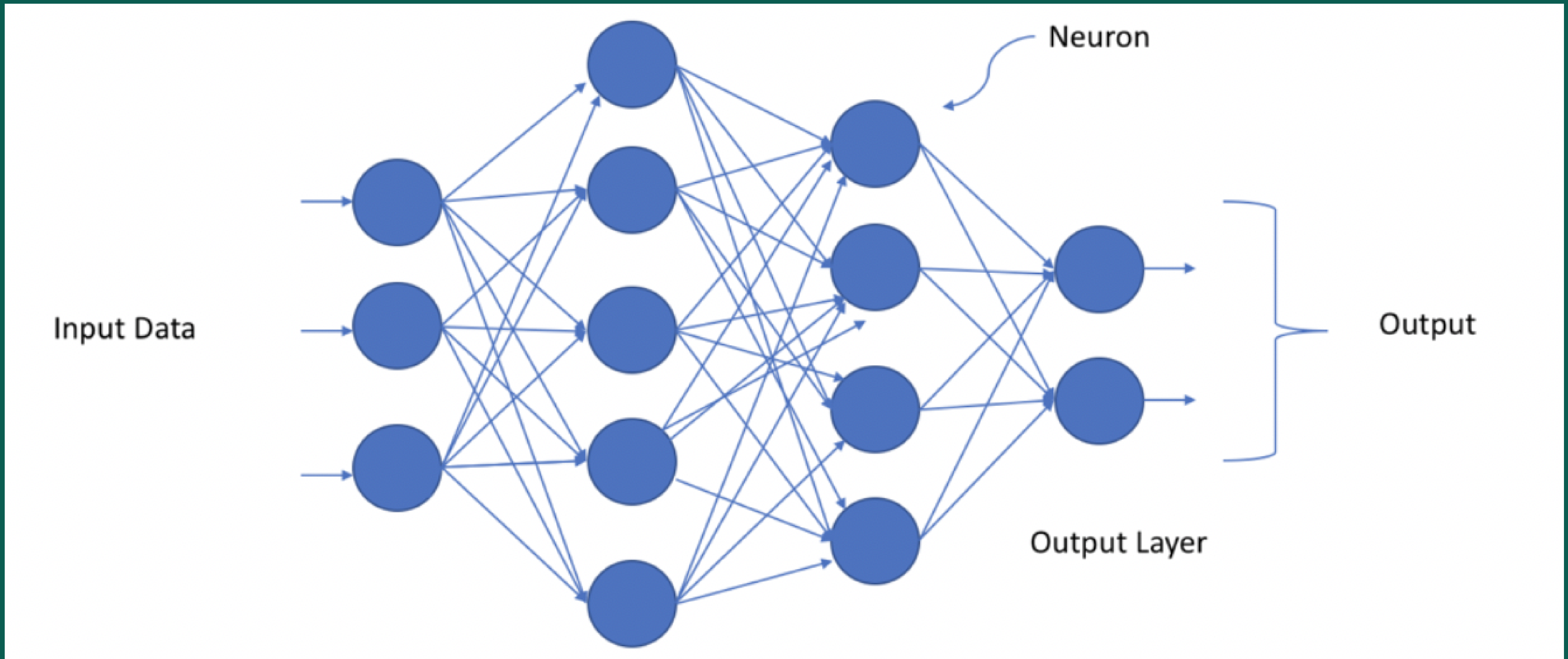
- Input nodes take in data, then output values to the next nodes, which in turn pass values to nodes until the output node is reached (returning the result).



Activation Functions

- Neurons apply (usually non-linear) activations functions to the inputs they receive, transforming them into outputs.

- The inputs to individual neurons come from either raw data, or the output of the previous neurons (weighted and modified by a propagation function).

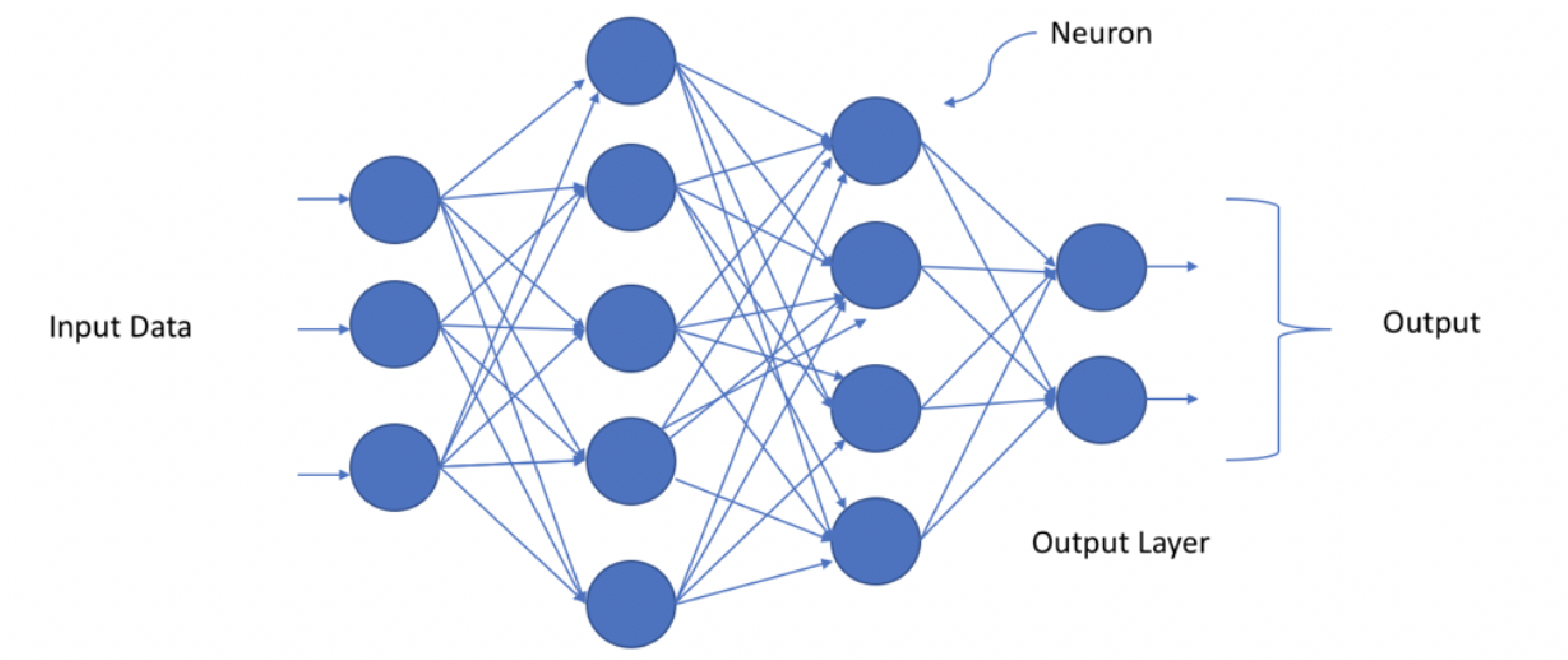


Connections

- Connections bridge the gaps between neurons. They take the output values from previous nodes, passing them as inputs into the next nodes (or output).

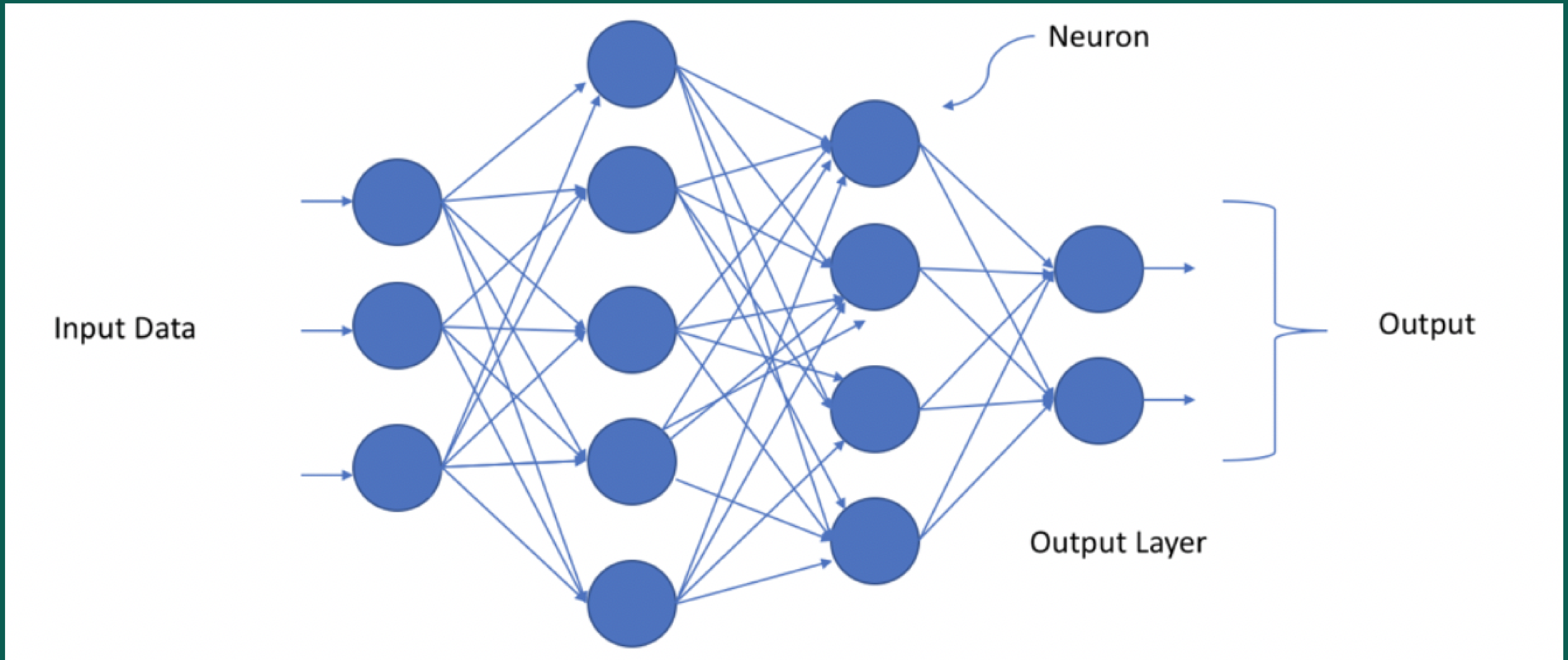
- They carry with them individual weights that represent their importance. Weights are usually assigned randomly, and alter the values that pass into neurons. These weights are usually the only thing that is changed during the learning process.

- Each neuron can have multiple input and output connections, but each neuron only produces one value.



Propagation Functions

- Propagation functions compute the final input of each neuron by taking the combined weights and values of each connection as a sum (sometimes a bias term is added to this value)

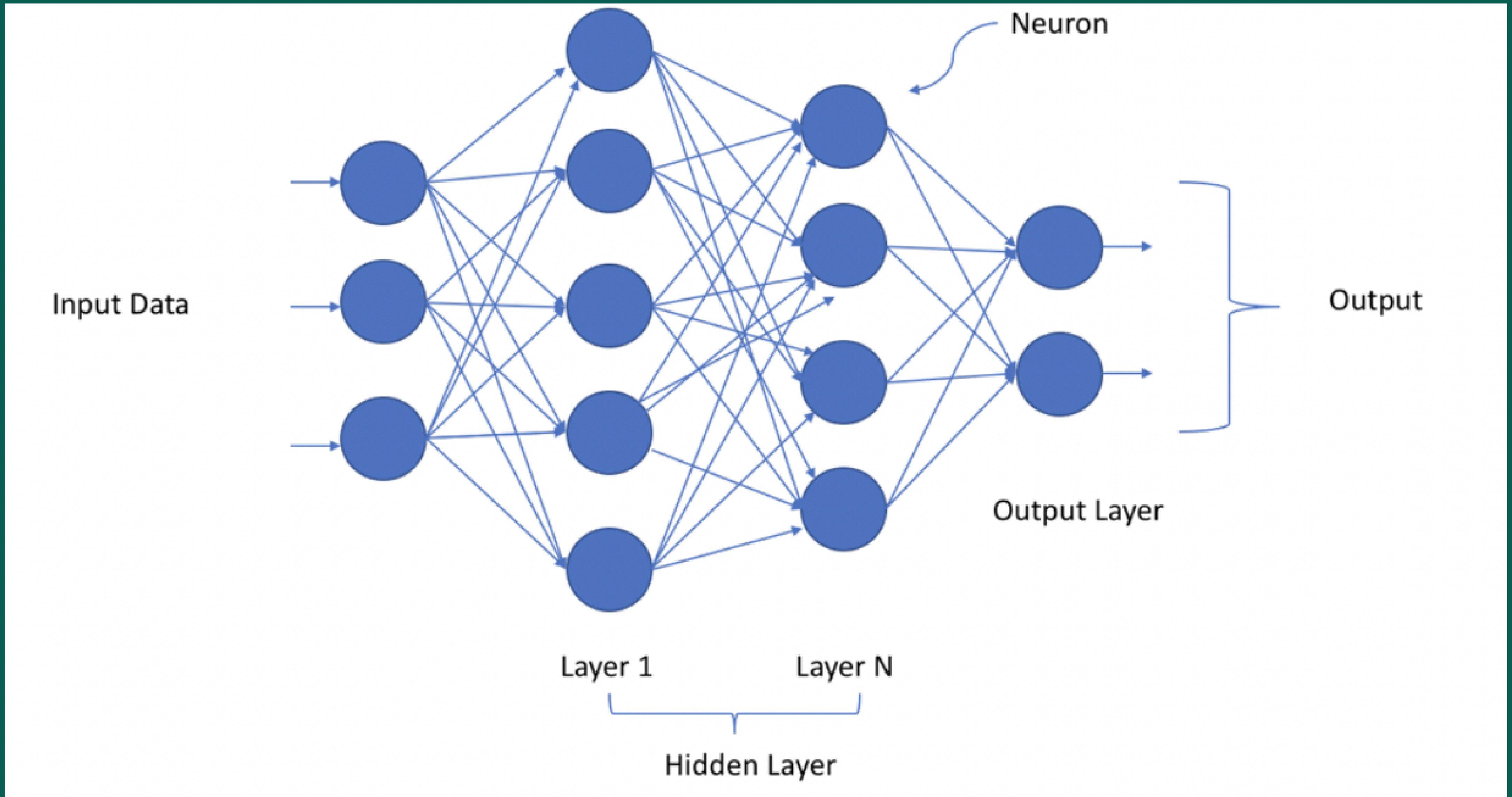


Layers

- Layers represent a group of neurons separated by connections.

-Layers that do not directly receive input or produce output are considered "hidden" layers (as seen below).

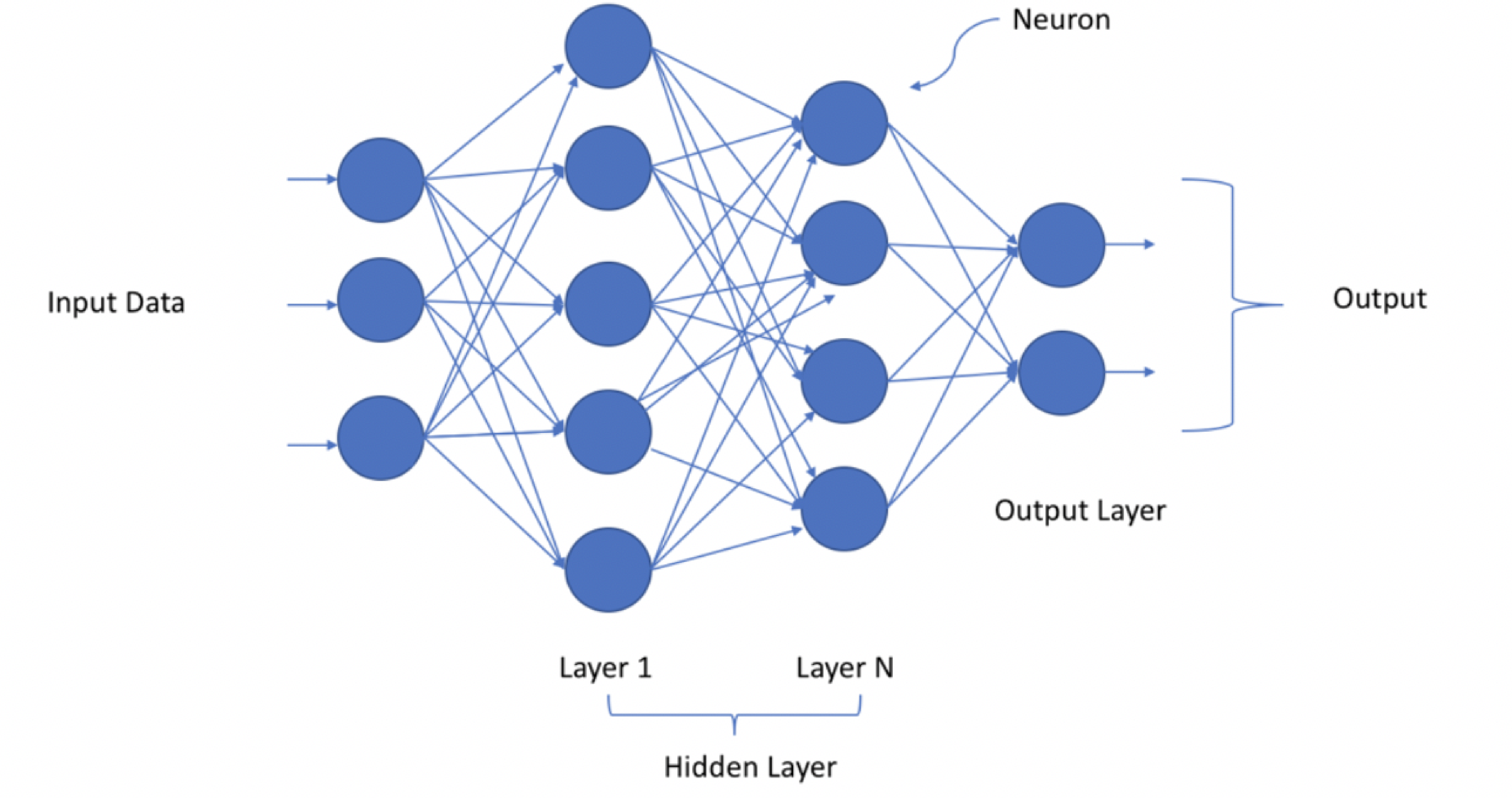
- In addition to the number of neurons and connections, the number of layers greatly contributes to the complexity of the network.



Layers (2)

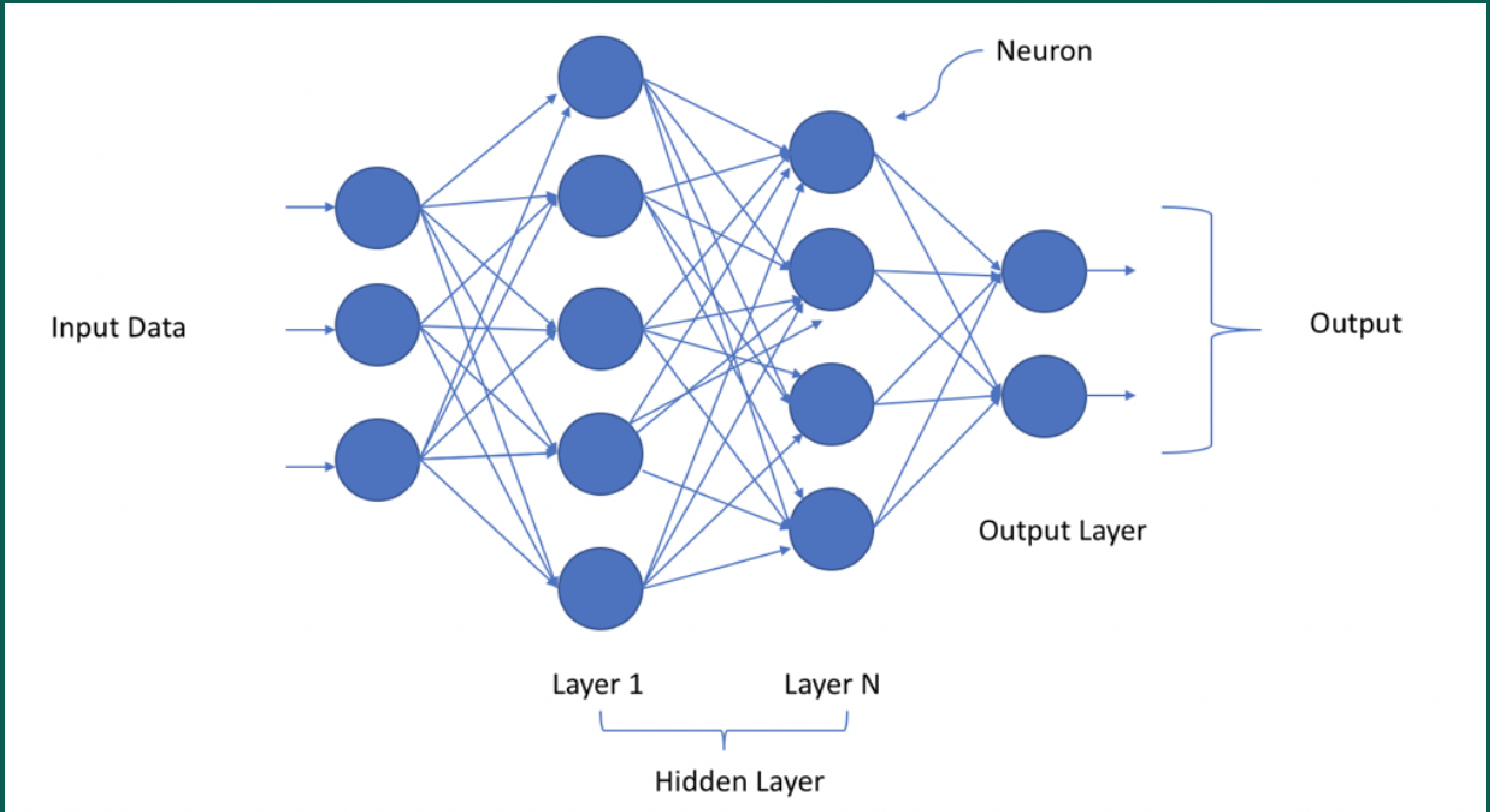
- Increasing the number of layers in a network greatly increases the computation time and abstraction of the model .

- "Deep Learning" neural networks are charactarized by models with many layers - (an increase in layers can be considered the model becoming "deeper".



Overview

- Input -> Input neurons (activation function) -> connections (with weights and propagation functions) -> layer 1 neurons (activation function) -> (repeat for n layers) -> output



Extras

- Disclaimer : This is not a comprehensive explanation of neural networks, simply an introduction.

-Neural networks can get extremely complicated, and include things such as :

Back-propagation (to adjust weights of connections)
Cost functions (to increase accuracy based on output accuracy)
Learning rates (to define how quickly models adjust to inaccuracy)
Biases (to help with the randomization and adjustment of weights)
Sigmoid Functions (to change continuous results to binary or vice versa)

Python Example

References

https://news.mit.edu/2017/explained-neural-networks-deep-learning-0414

https://towardsdatascience.com/first-neural-network-for-beginners-explained-with-code-4cfd37e06eaf

https://en.wikipedia.org/wiki/Artificial_neural_network