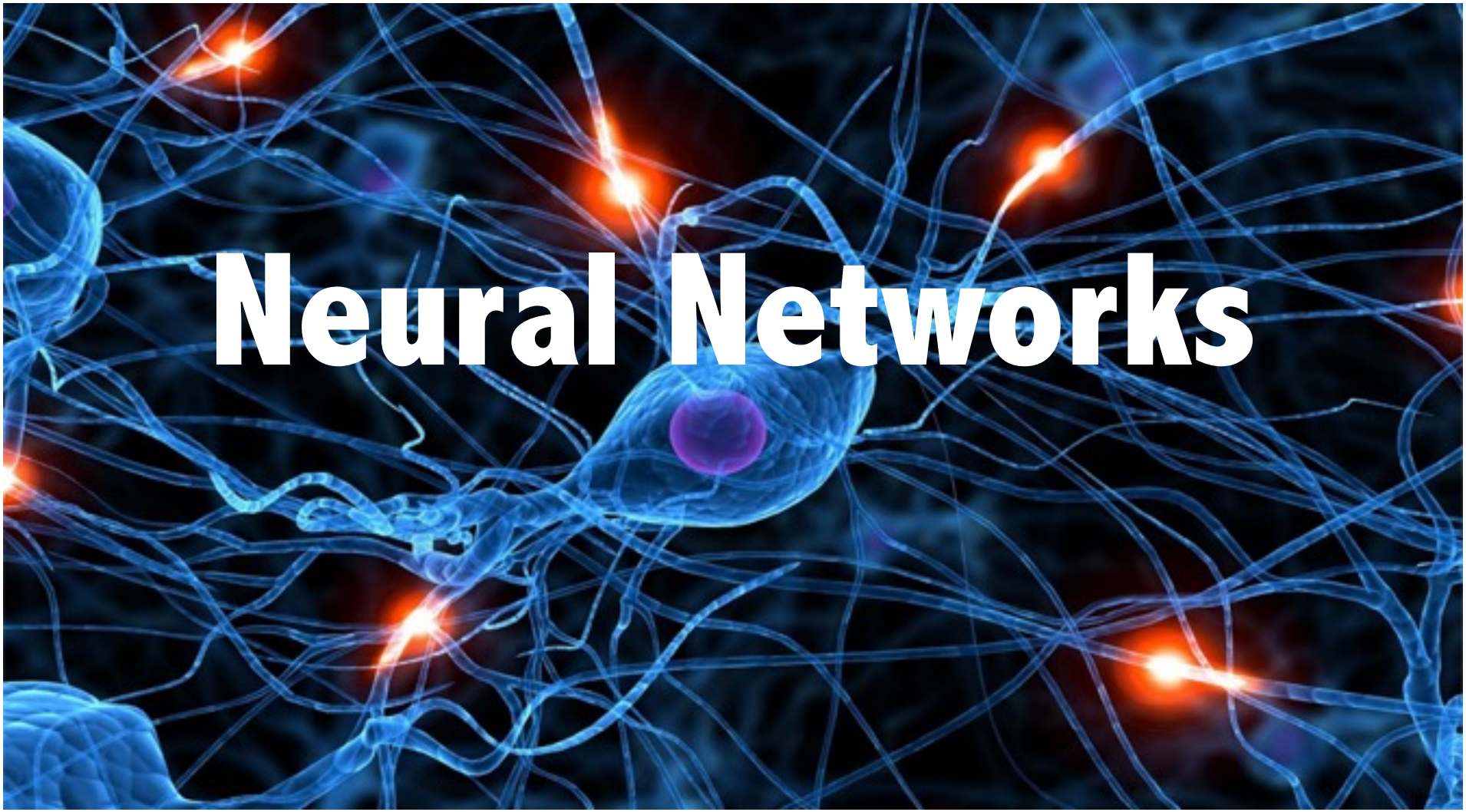

Neural Networks

Review: Machine learning concepts

- Three forms:
- Supervised learning
  - The agent is given some input-output pairs and it learns a function that maps the input to the output.
  - Example: training a naïve Bayes classifier.
- Unsupervised learning
  - The agent learns patterns in the input even though no explicit output or feedback is given.
  - Example: clustering
- Reinforcement learning
  - The agent is given feedback (rewards) during the steps of a task and the agent learns a function from states to predicted rewards.

Supervised learning

- The agent is given some input-output pairs (*labeled* data) and it learns a function that maps the input to the output.
  - The input-output pairs given to the learning algorithm are called the *training set*.
  - The hope is that the function learned will do a good job at mapping previously-unseen inputs (inputs not in the training set) to outputs.
  - Sometimes, in order to evaluate how well a supervised learning algorithm performs, we hold back some of our input-output pairs and have a separate data set called the testing set that we use solely for evaluation, not for training.
- Most common algorithms are categorized as **classification** algorithms (output is categorical) or **regression** algorithms (output is numeric).

Supervised Learning
(Classification Algorithm)

Duck

Duck

Not Duck

Not Duck

Supervised Learning

Predictive Model

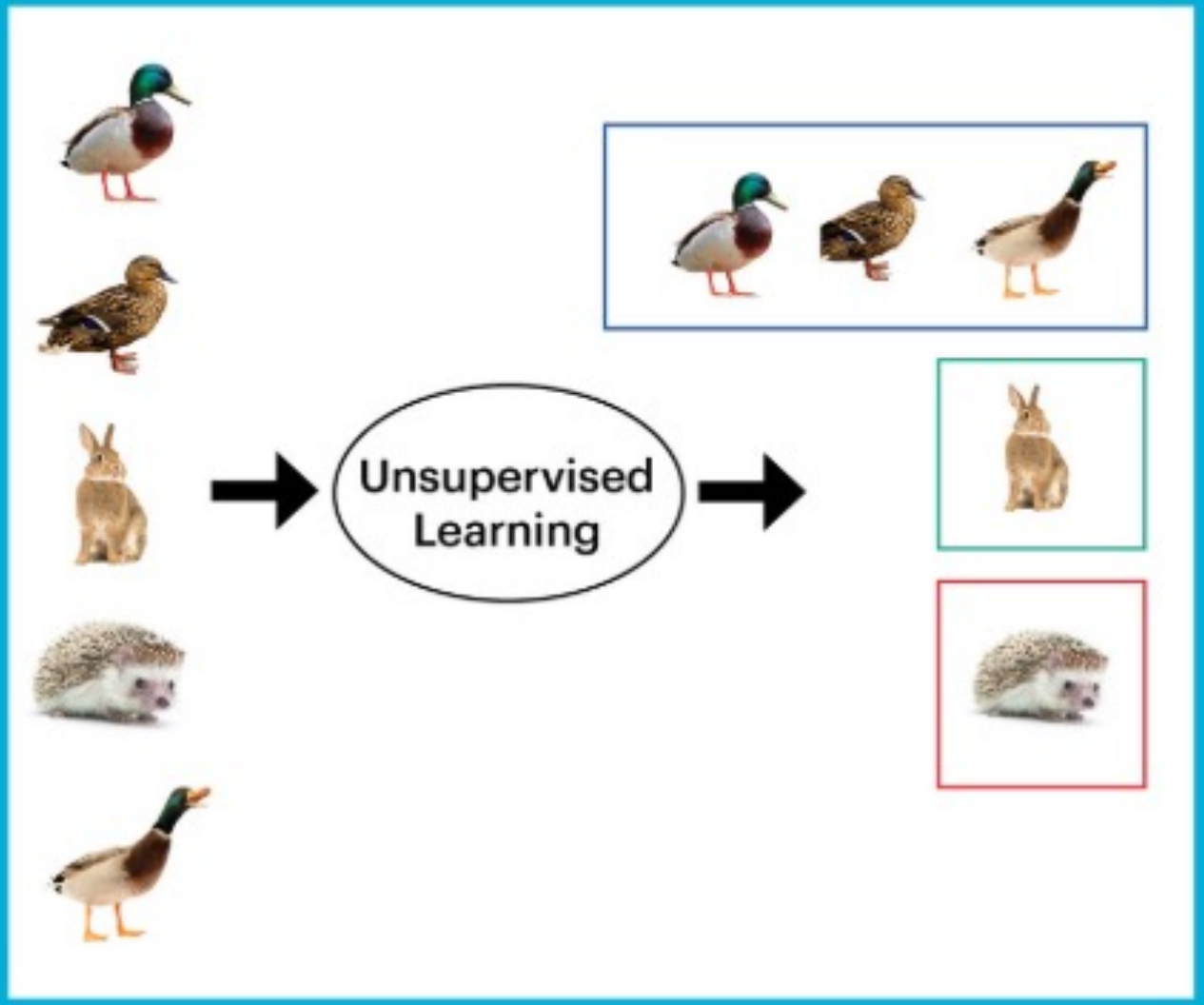
Predictive Model

Duck

Unsupervised learning

- The agent learns patterns in the input even though no explicit output or feedback is given.
- Training data is not labeled, so the goal is not to learn a function, but rather to find commonalities in the training set, and use those commonalities to draw inferences about new data.

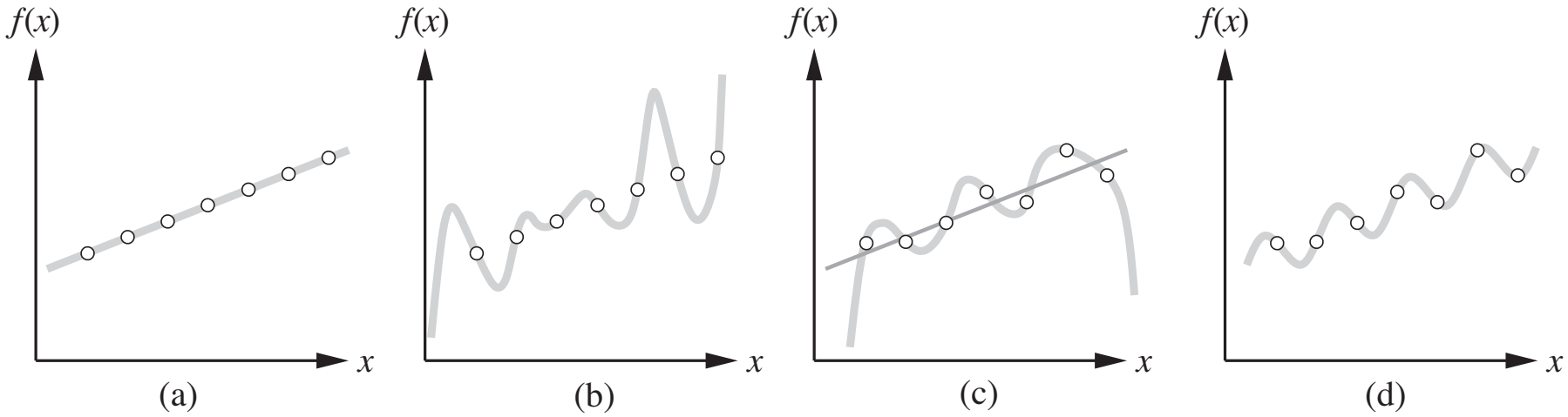
Unsupervised Learning (Clustering Algorithm)



Supervised learning

- Given a ***training set*** of *N* example input-output pairs:
  - $(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)$
- Each y is generated by an unknown function y = f(x).
- Goal: discover a function h that approximates the true function f.
- h is called a ***hypothesis***.
- Machine learning algorithms conduct searches for the "best" h.
- We can measure the accuracy of a hypothesis on a ***test set*** of examples that are distinct from the training set.
- A hypothesis ***generalizes well*** if it correctly predicts examples from the test set (even though it has never seen them before).

Supervised learning

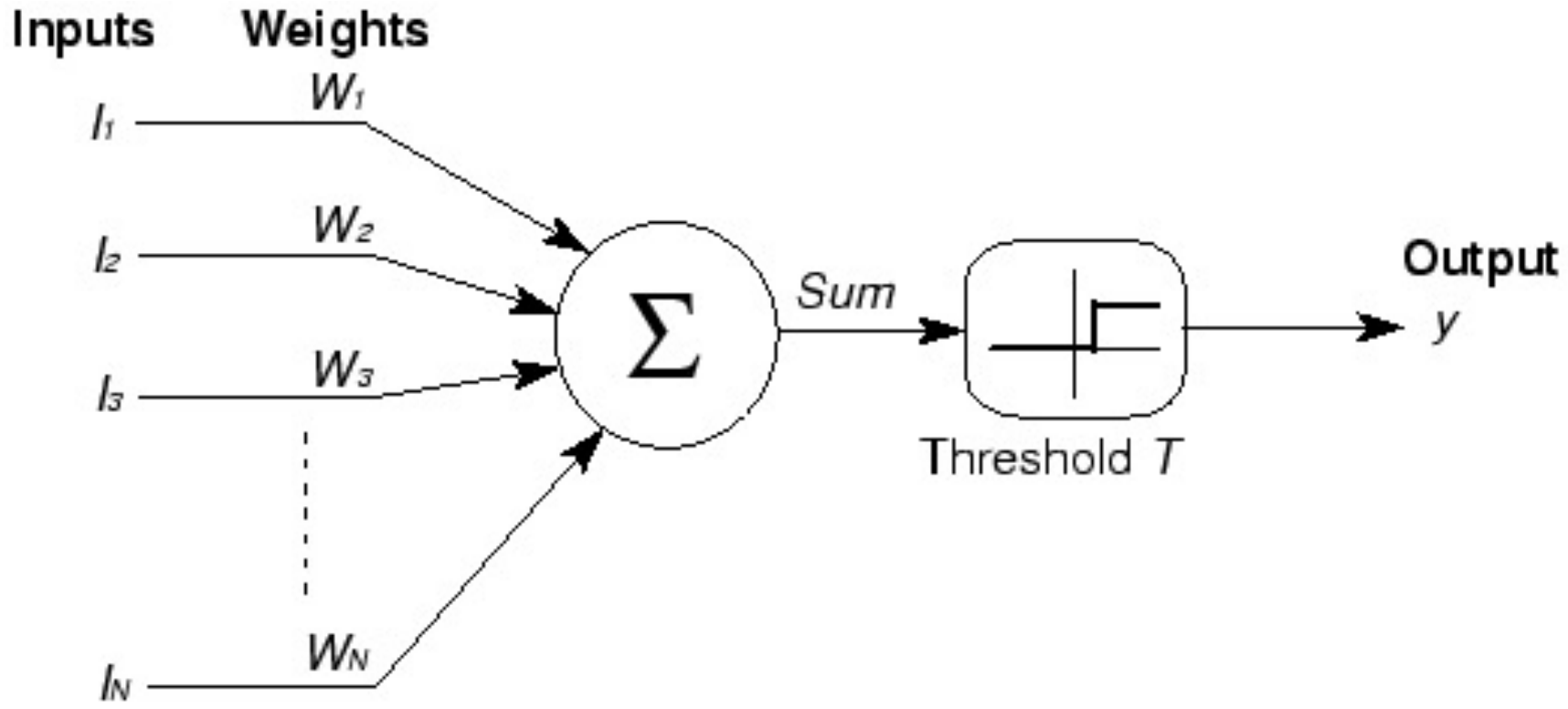


Supervised learning

- Poor generalization is sometimes caused by overfitting: our hypothesis has learned the training set very well, but it has poor accuracy on the test set.
  - Analogous to "memorizing" the training set.
- When the output y is one of a finite set of values (e.g., sunny/cloudy/rainy or true/false), the learning problem is called ***classification***.
- When the output is a number, the problem is called ***regression***.
  - Yes, linear regression is a machine learning algorithm!

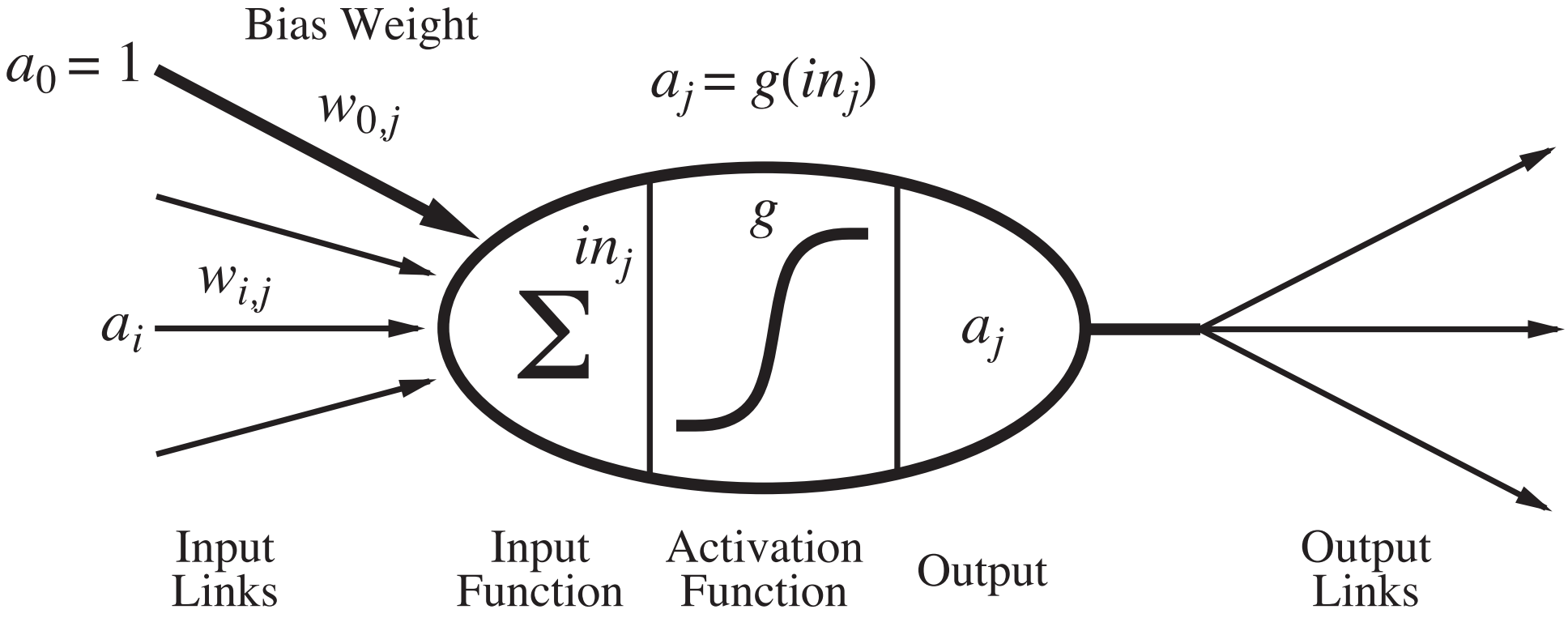
McCullough-Pitts neuron

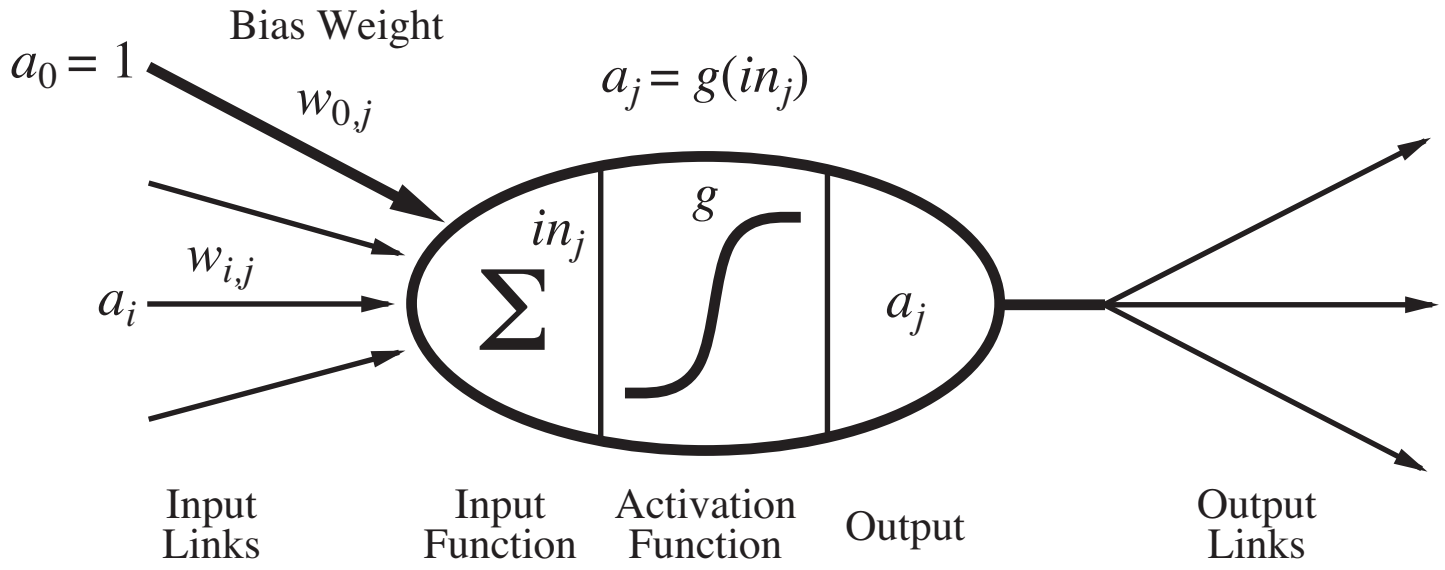
- 1943: Warren McCullough and Walter Pitts, two electrical engineers, develop the first model of an ***artificial neuron***, called threshold logical units.



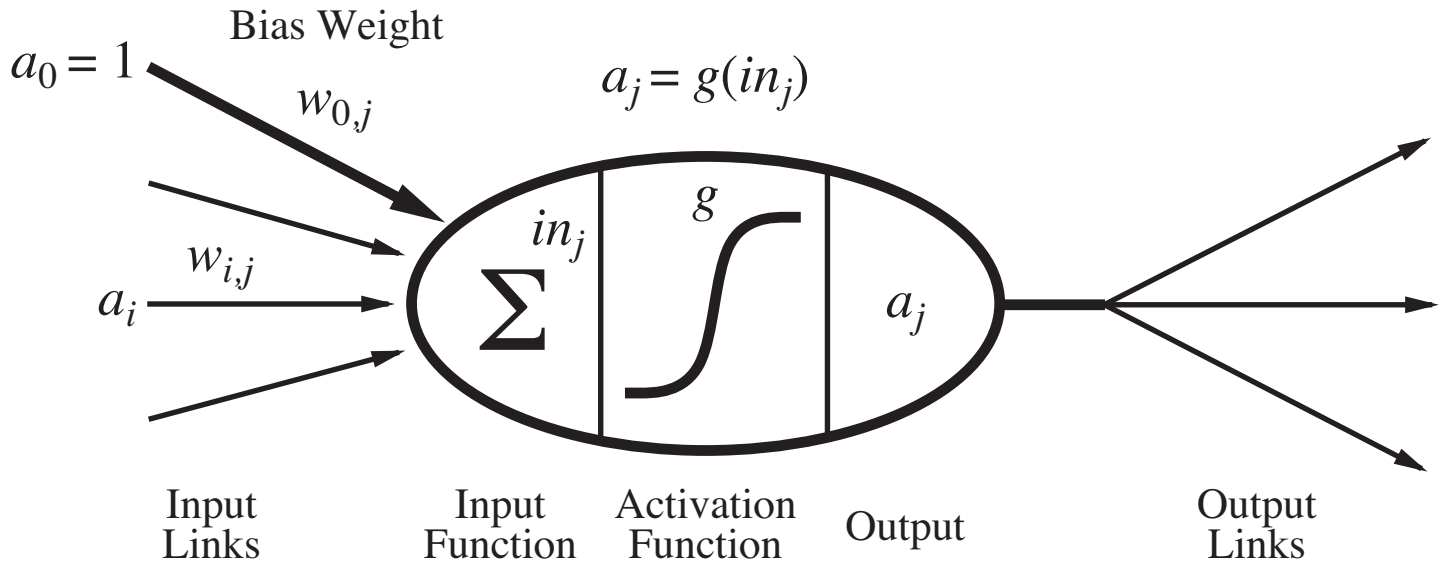
Perceptron

- 1958: Frank Rosenblatt refined the McCullough-Pitts neuron into the ***perceptron***.





- NNs are composed of nodes or units connected by directed links (a graph structure).
- Each unit receives a collection of numeral inputs (a_0, a_1, ...) and produces a numeral output (a_j).
- A link from unit *i* to unit *j* has a weight w_{ij} associated with it.
- Each unit has a dummy input (a_0) that is always set to 1.

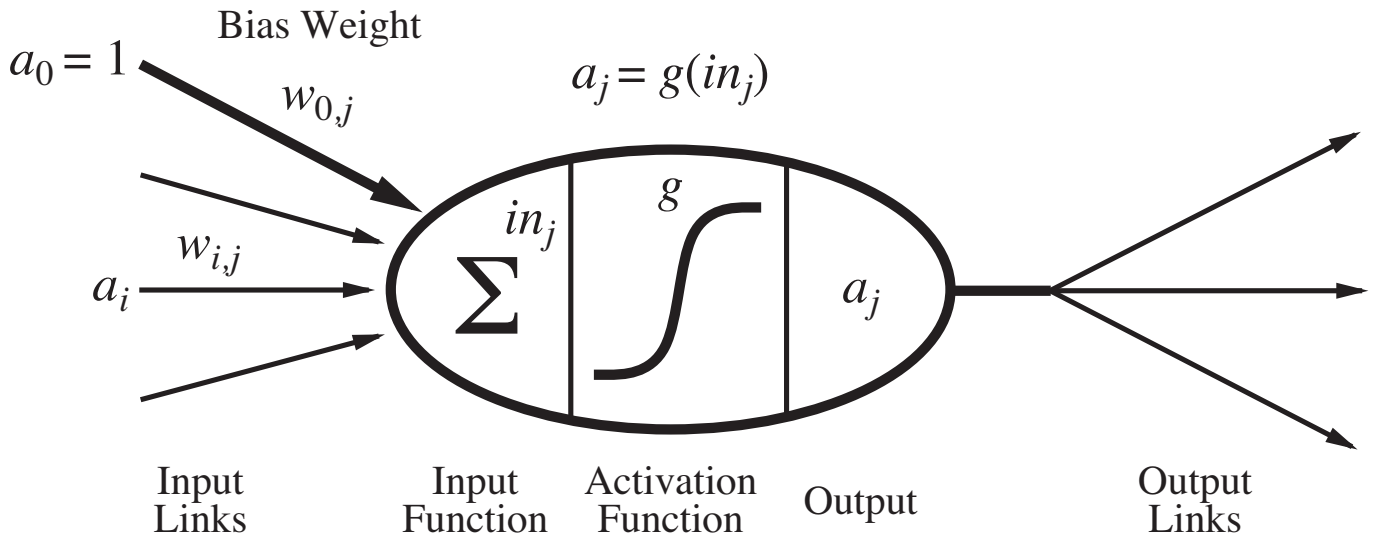


- Each unit j first computes a weighted sum of its inputs:

$$in_j = \sum_{i=0}^{n} w_{i,j} \cdot a_j$$

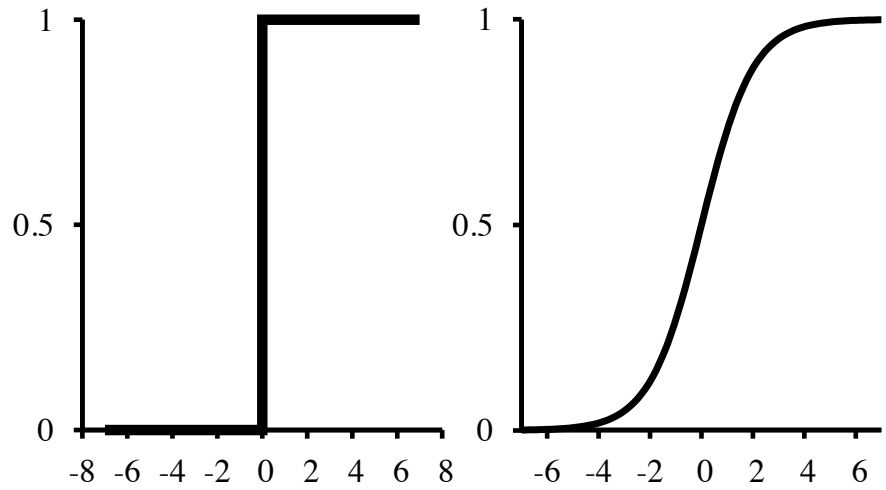
- Then it applies an activation function g to this sum to produce the output:

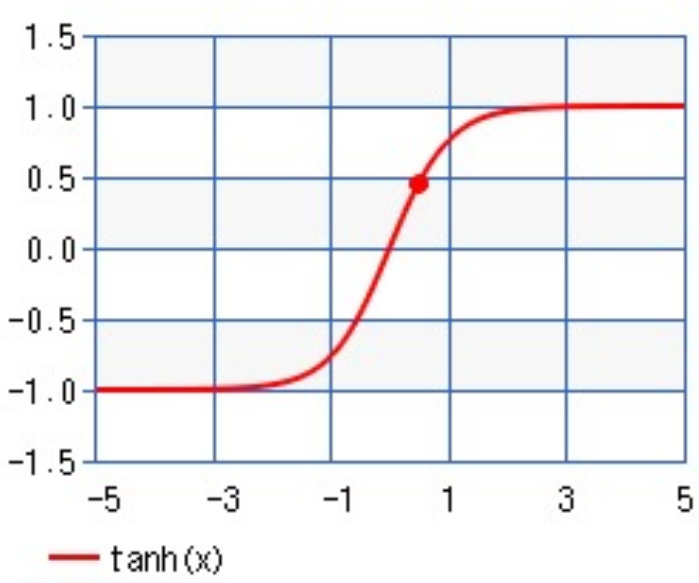
$$a_j = g(in_j)$$

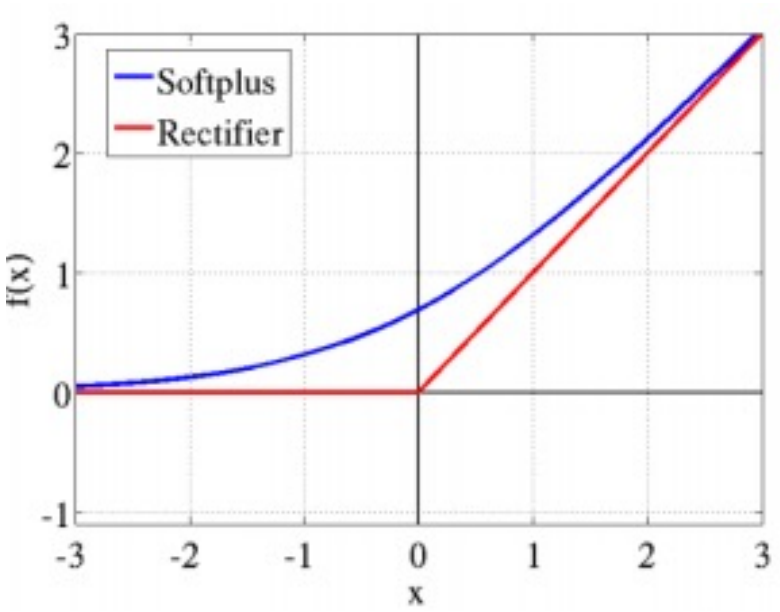


- A few different activation functions are common:
- threshold
- logistic or sigmoid
- ReLU (rectified linear unit)
- softplus
- tanh (hyperbolic tangent)

- A few different activation functions are common:
- threshold
- logistic or sigmoid = 1/(1+e^x)
- ReLU (rectified linear unit) = max(0, x)
- softplus = log(1 + e^x)
- tanh (hyperbolic tangent)





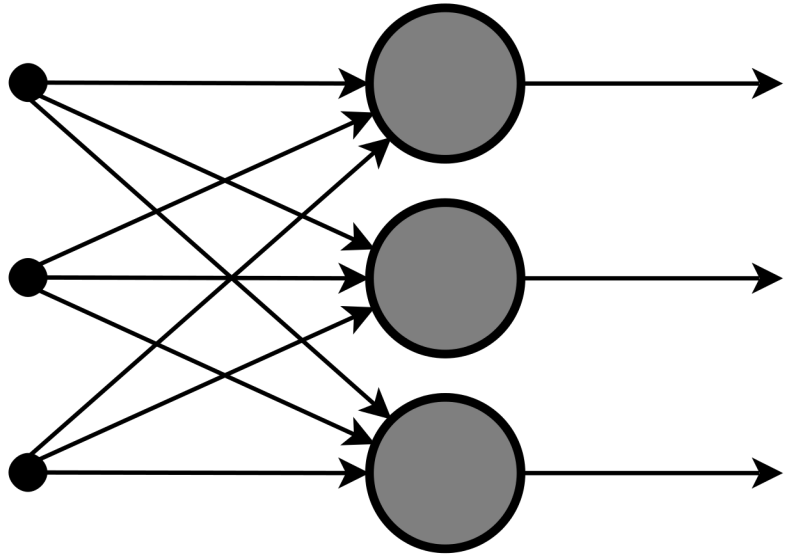


Neural networks

- Two basic types of networks.
  - Feed-forward: Links are only in one direction (DAG).
  - Recurrent: Allows outputs to feed back into inputs.
    - System may reach a steady state or may exhibit oscillations or chaotic behavior.
- Feed-forward networks are usually arranged in layers, where each layer only receives input from the previous layer.
  - Single layer – all inputs connected directly to outputs
  - Multi-layer - one or more ***hidden layers*** of units in between input and output.

Single layer feed forward networks

- One input layer (which is just the raw inputs).
- One output layer (of perceptron units).
- Example.



Single layer feed forward networks

- One input layer (which is just the raw inputs).
- One output layer (of perceptron units).
- Let's design a network to add two bits together.
- Needs two inputs (x_1, x_2), and two outputs (y_3, y_4).