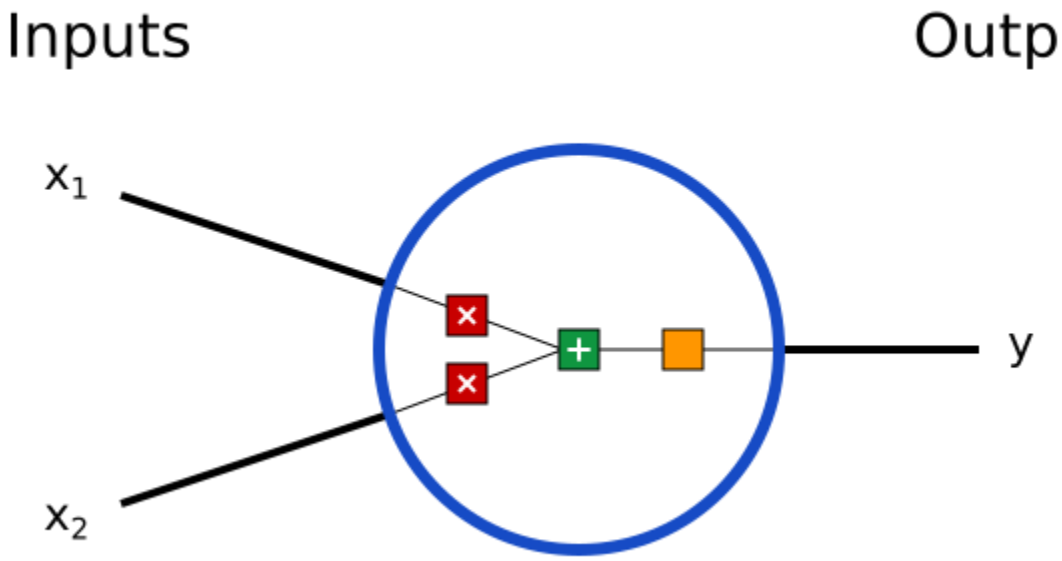
9.01 Neural Networks

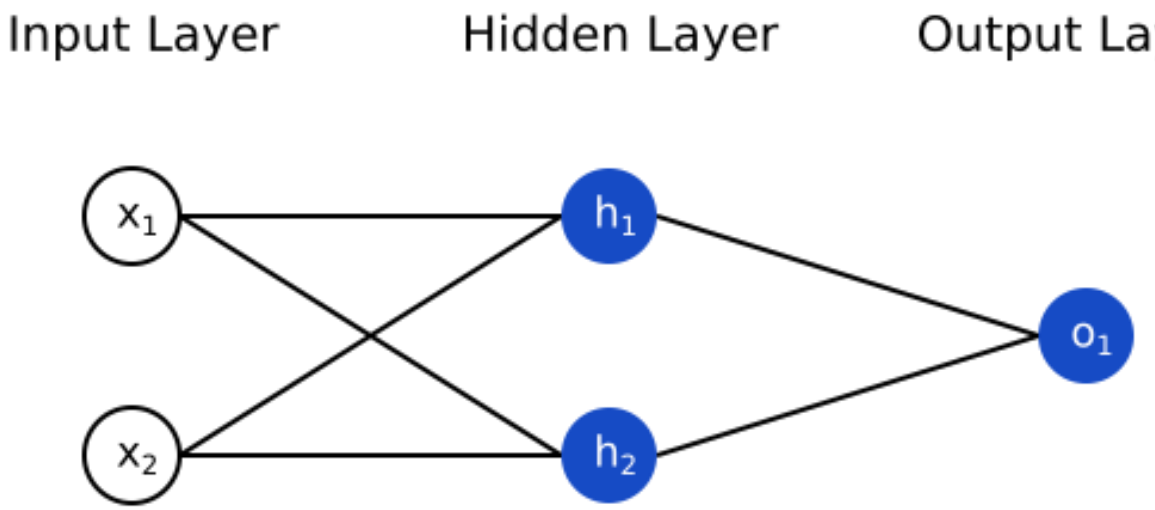
Building Blocks: Neurons

- Basic unit of a neural network
- Takes inputs, does some math with them, and produces one output
- Here's what a 2-input neuron looks like:



Combining Neurons into a Neural Network

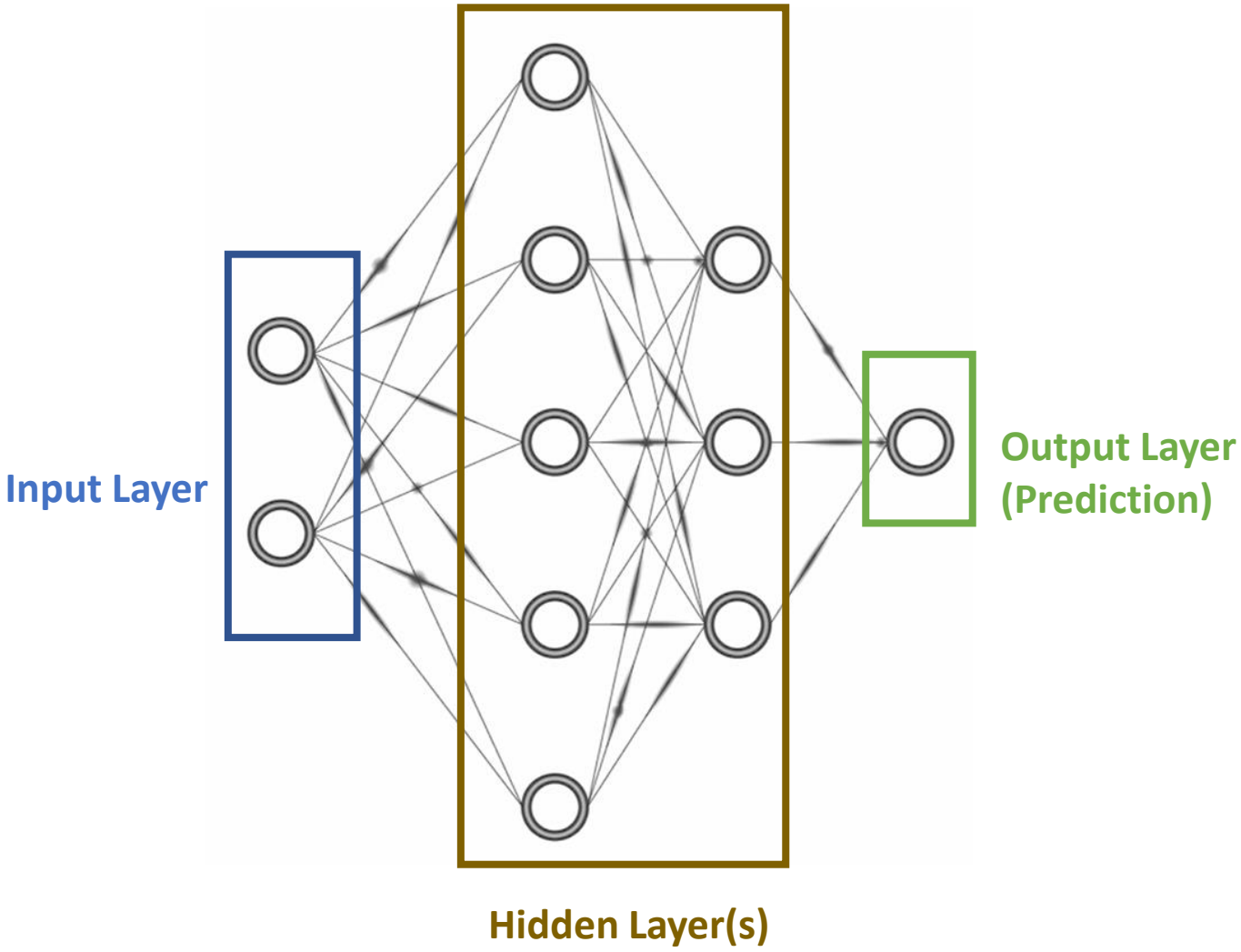
- A neural network is nothing more than a bunch of neurons connected together. Here's what a simple neural network might look like:



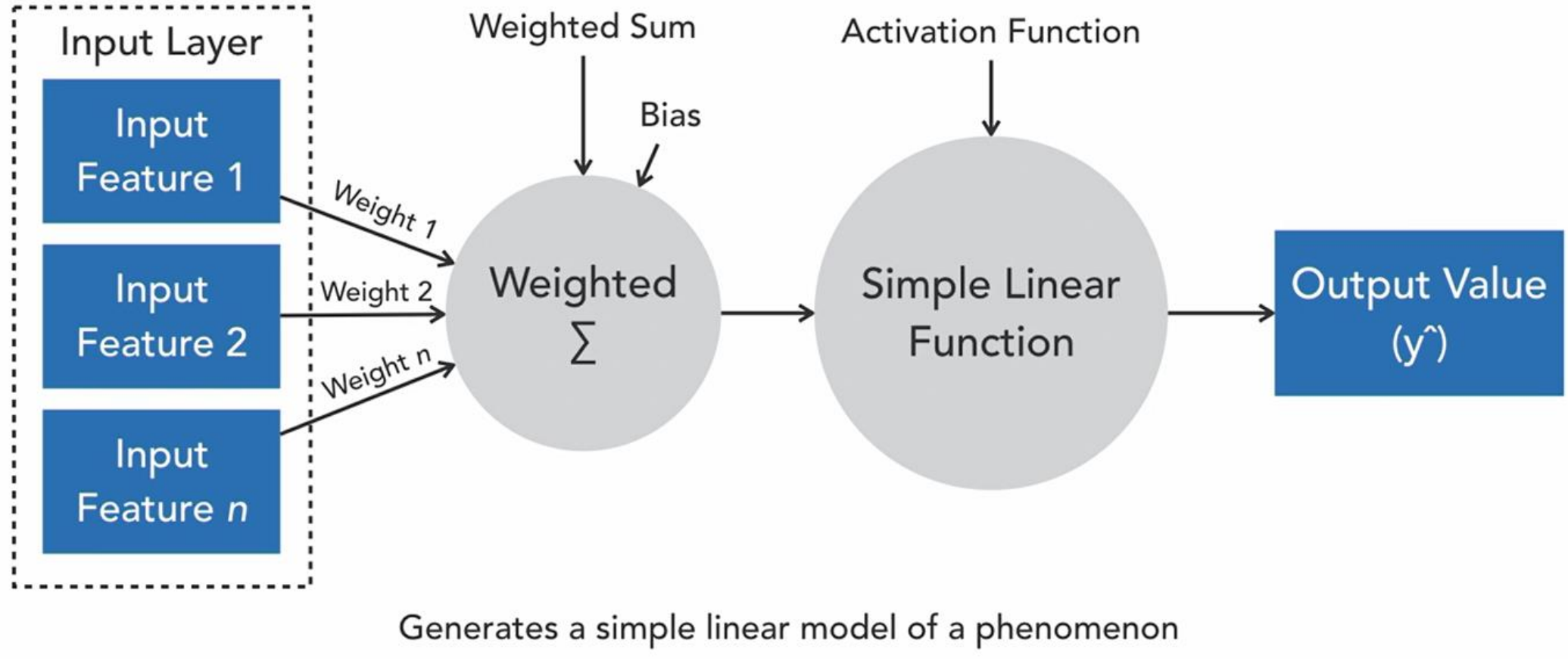
Neural Network

- A neural network can have **any number of layers** with **any number of neurons** in those layers.

- The basic idea stays the same:
- feed the input(s) forward through the neurons in the network to get the output(s) at the end. For simplicity, we'll keep using the network pictured above for the rest of this post.



Single-Layer Neural Network



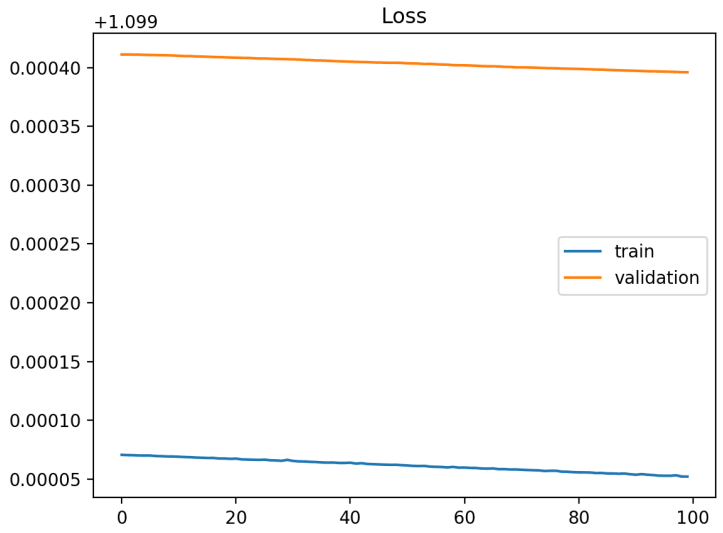
max_iter Hyperparameter

- The maximum number of passes over the training data (aka epochs)
- Defines the number of times that learning algorithm will work through entire training dataset
- The number of epochs is traditionally large (100, 1000, etc.) allowing the learning algorithm to run until the error from the model has been sufficiently minimized.

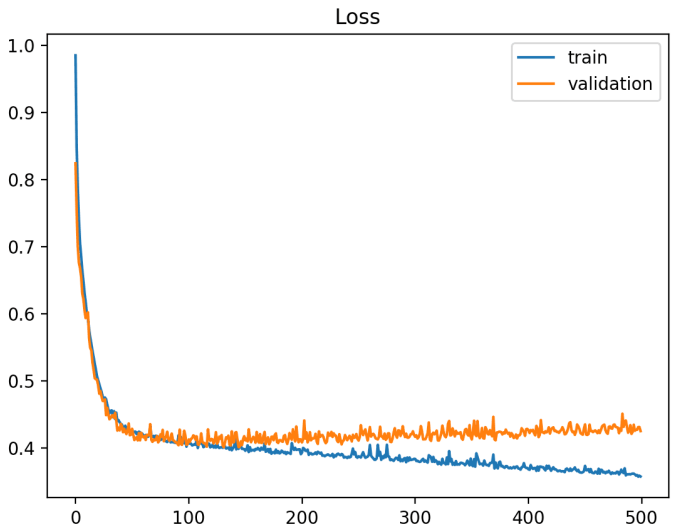
max_iter Hyperparameter

- It is common to create line plots that show epochs along the x-axis as time and the error or skill of the model on the y-axis.
  - These plots are sometimes called learning curves.
  - These plots can help to diagnose whether the model has **over learned, under learned, or is suitably fit** to the training dataset.

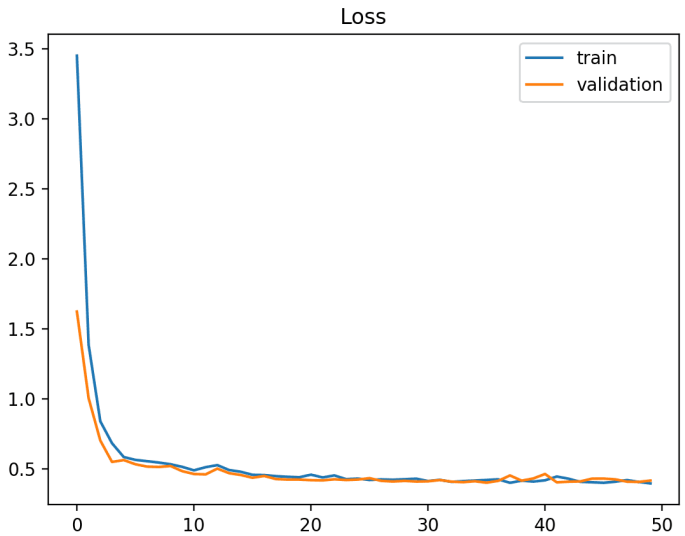
Learning Curves of Training Losses



Learning Curves Showing An **Underfit Model** That Doesn't Have Sufficient Capacity For Complexity Of Data Set



Learning Curves Showing an **Overfit Model** that has learned training data too well, including noise or random fluctuations



Learning Curves Showing a **Good Fit** where loss decreases to a point of stability and has a small gap with the training loss.

Multi-layer Perceptron (MLP)

- A common form of Neural Network applicable for time series data, which can model complex relationships over time
- A great way to find meaning in your temporally structured data that is always moving in one direction in the future
- Ability to distinguish non-linearly separable data