Lecture 23 Part 2 - Multi-Layer Perceptron (MLP) & Backpropagation

Multi-Layer Perceptron (MLP)

A **Multi-Layer Perceptron (MLP)** is a class of feed-forward artificial neural network (ANN). An MLP is composed of one **input layer**, one or more layers of TLUs, called **hidden layers**, and one final layer of TLUs called the **output layer**. Except for the input nodes, each node is a neuron that uses an activation function (either linear or non-linear).

- The layers close to the input layer are usually called the lower layers, and the ones close to the outputs are usually called the upper layers.
- Every layer except the output layer includes a bias neuron and is fully connected to the next layer.

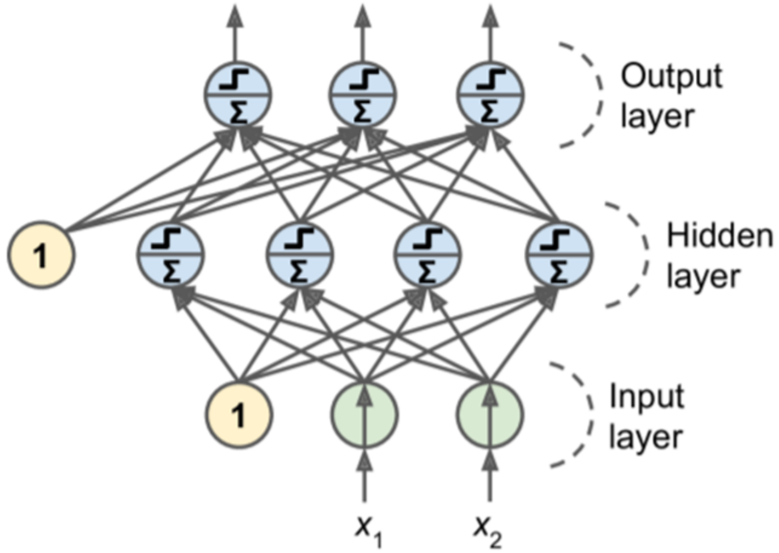
Its multiple layers and non-linear activation distinguish MLP from a linear perceptron. It can distinguish data that is not linearly separable.

When an Artificial Neural Network (ANN) contains a deep stack of hidden layers, it is called a **deep neural network (DNN)**. The field of Deep Learning studies DNNs, and more generally models containing deep stacks of computations.

In [1]:

```
from IPython.display import Image
Image('figures/MLPs.png',width=400)
#Source: Aurélien Géron, "Hands-on machine learning with scikit-learn, keras and tenso
```

Out[1]:



In MLPs, each neuron's output can be subject to different activation functions. The choice of the activation function is in itself a *hyperparameter*.

- Which activation function would you use if your desired labels are $\{1, 2, 3, 4, \ldots, 8, 9\}$?

- *What each of the layers do?*

  - The first hidden layer draws boundaries
  - The second hidden layer combines the boundaries
  - The third and further layers can generate arbitrarily complex shapes

---

Exercise

Suppose you had the following neural network:

<img src="figures/MLP.png", width="300"><!div>

with the activation function:

$$\phi(x) = \begin{cases} 1 & x > 0 \\ -1 & x \leq 0 \end{cases}$$

1. What is the expression of the output value y in terms of the input values?

2. What is the output with the following input values?

   - $[0, 0]$
   - $[-2, -2.5]$
   - $[-5, 5]$
   - $[10, 3]$
3. What does the decision surface of this network look like graphically? Draw it out by hand.
