By John 'Jack' Collins

A beginner's guide to neural networks for social scientists

20/10/2023

Overview

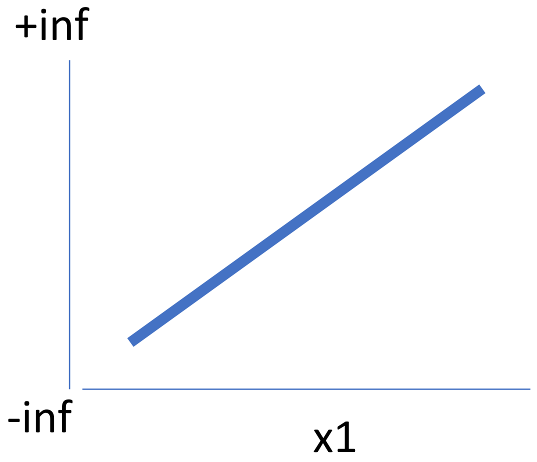
- Neural Networks are a very powerful type of ML mode.
- But their power is that they can be used for very complex problems.
- For simpler problems, simpler models are usually better.
- The simple neural networks we cover here are stepping stones to more complex types of NNs, such as:
  - Convolutional Neural Networks which are suited for processing images.
  - Recurrent Neural Network: Suited for handling time sequences of any length
  - Transformers: which are the core of GPT (ie: ChatGPT) and BERT (ie: Google Translate)

Revision: What is ML

- Data about cases go in
- Predictions about those cases come out
- Regression = The prediction is a continuous number (ie: annual income)
- Classification = The prediction is a category, or the probability that the given case belongs to certain categories (ie: sub-species of Iris flower).

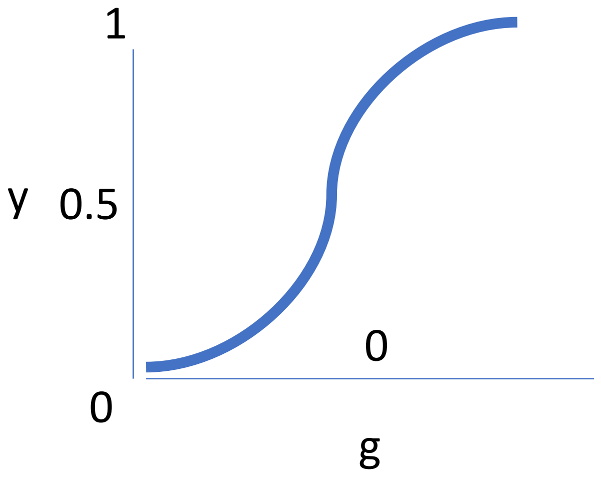
$$\hat{y} = f(X)$$

Revision: Logistic Regression

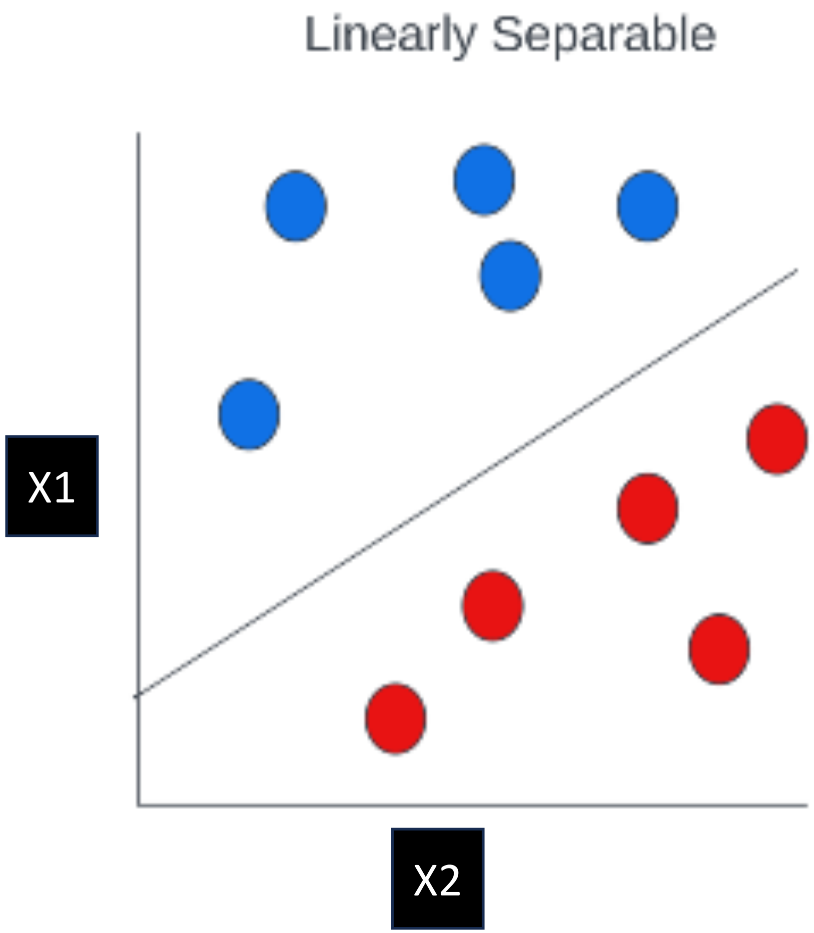


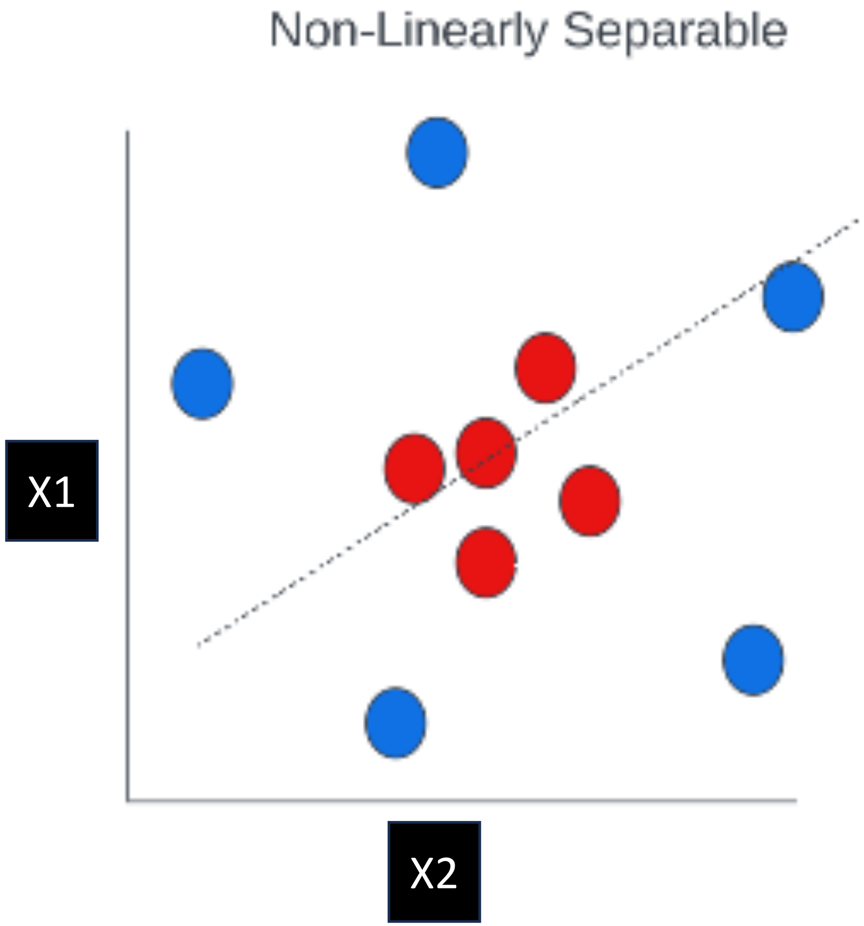
$$g = x_1 \cdot w_1 + x_2 \cdot w_2 + \cdots x_n \cdot w_n + \beta$$

$$\hat{y} = \frac{1}{1 + e^{-(g-1)}}$$



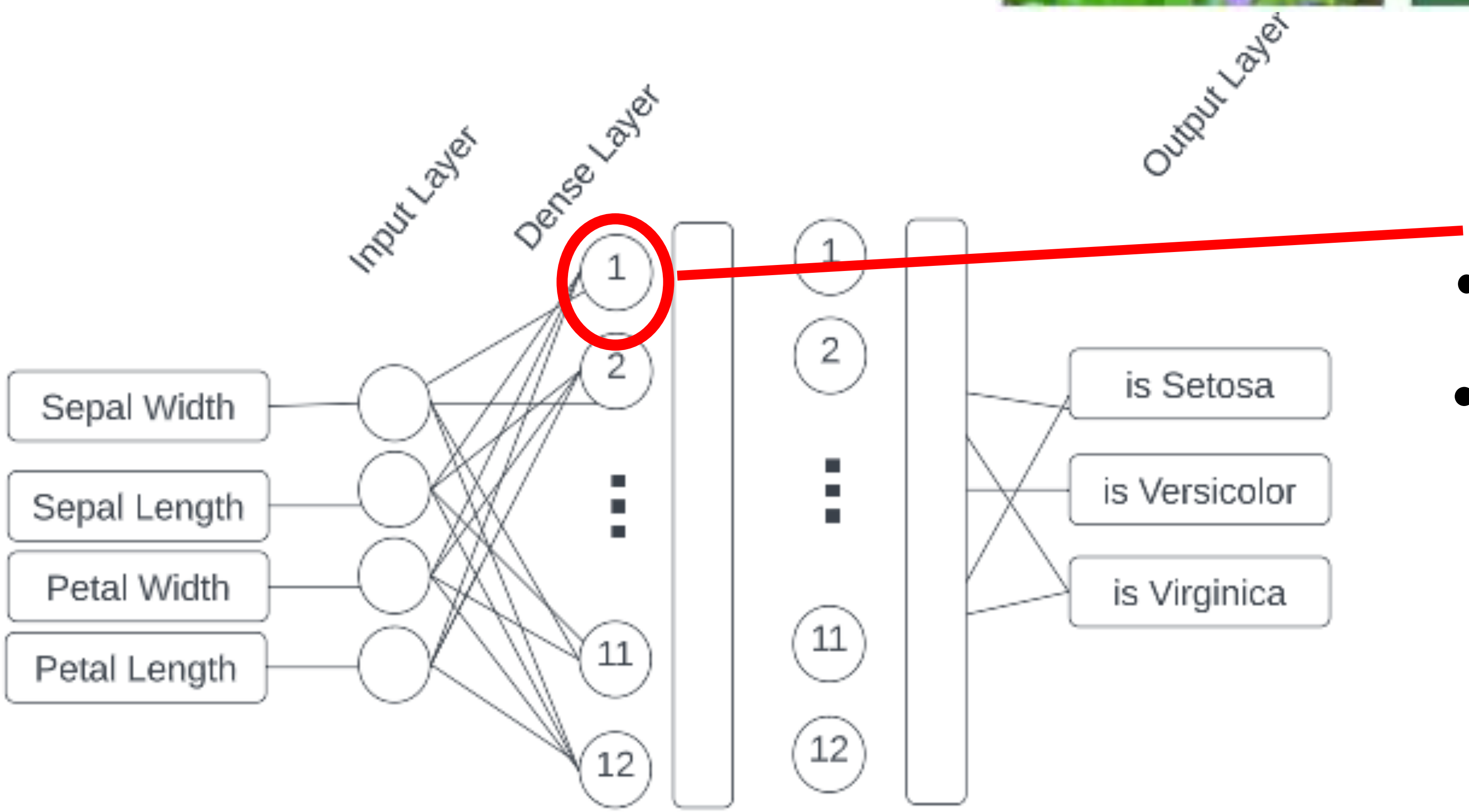
Linearly Separable Problems





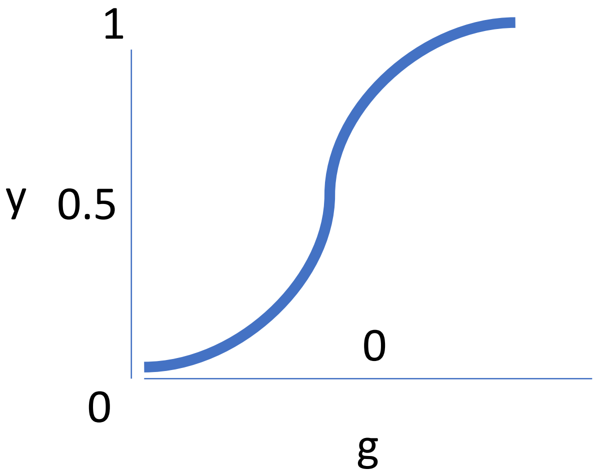
Neural Networks solve non-linear problems!

Classifying Iris Flowers



Activation Functions

- g= $x_1 \cdot w_1 + x_2 \cdot w_2 + \cdots x_n \cdot w_n + \beta$
- $y = \dfrac{1}{1 + e^{-(g-1)}}$

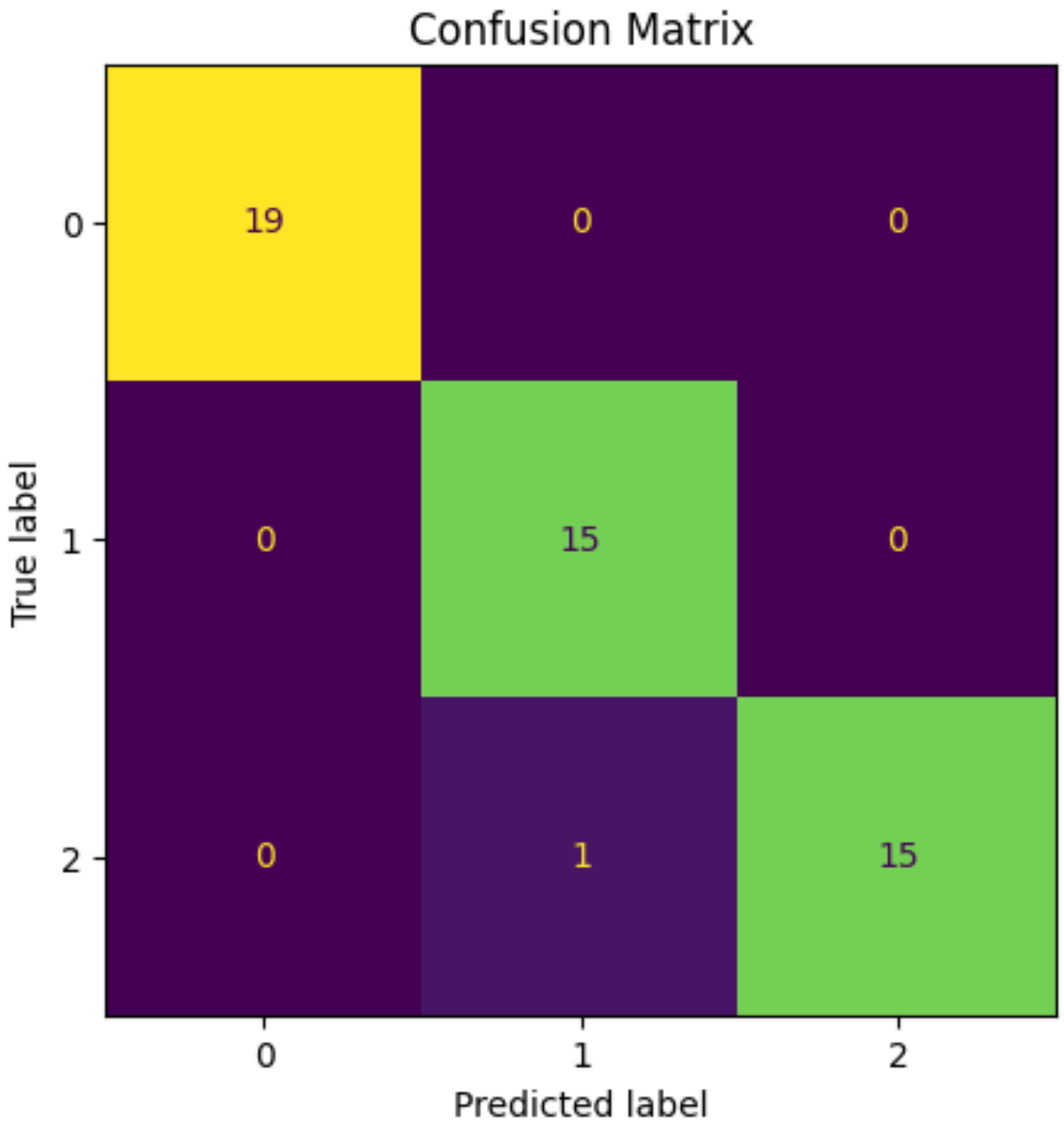


Results

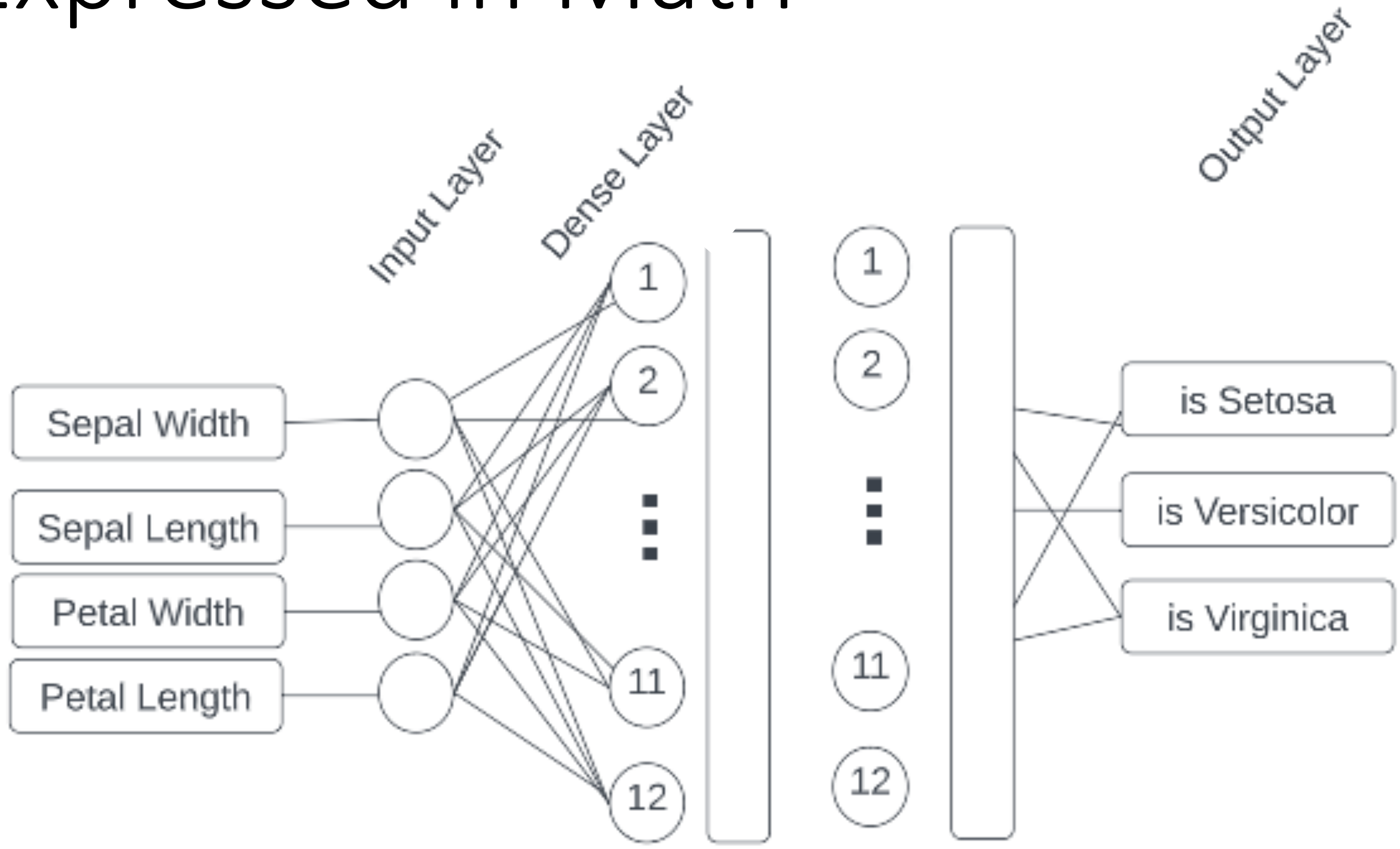
Setosa

Versicolor

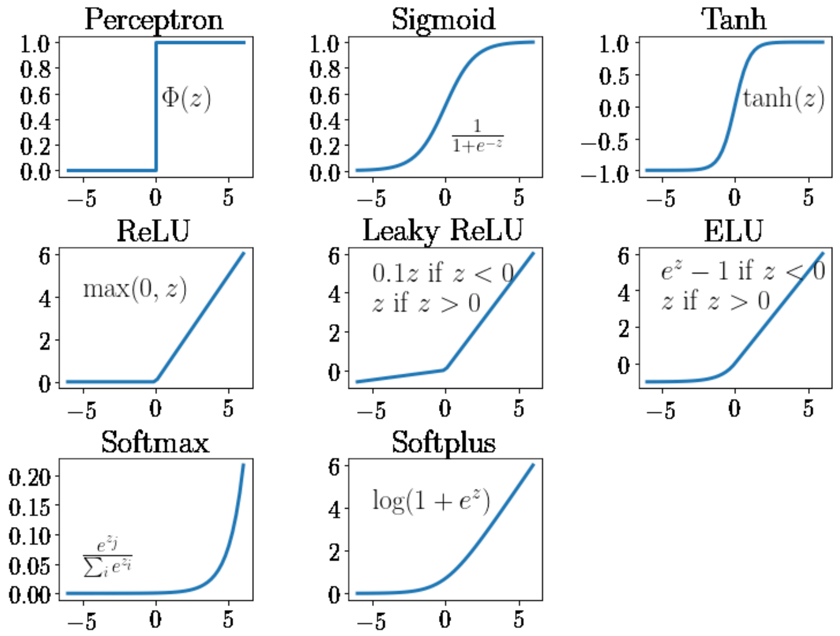
Virginicia



Expressed in Math



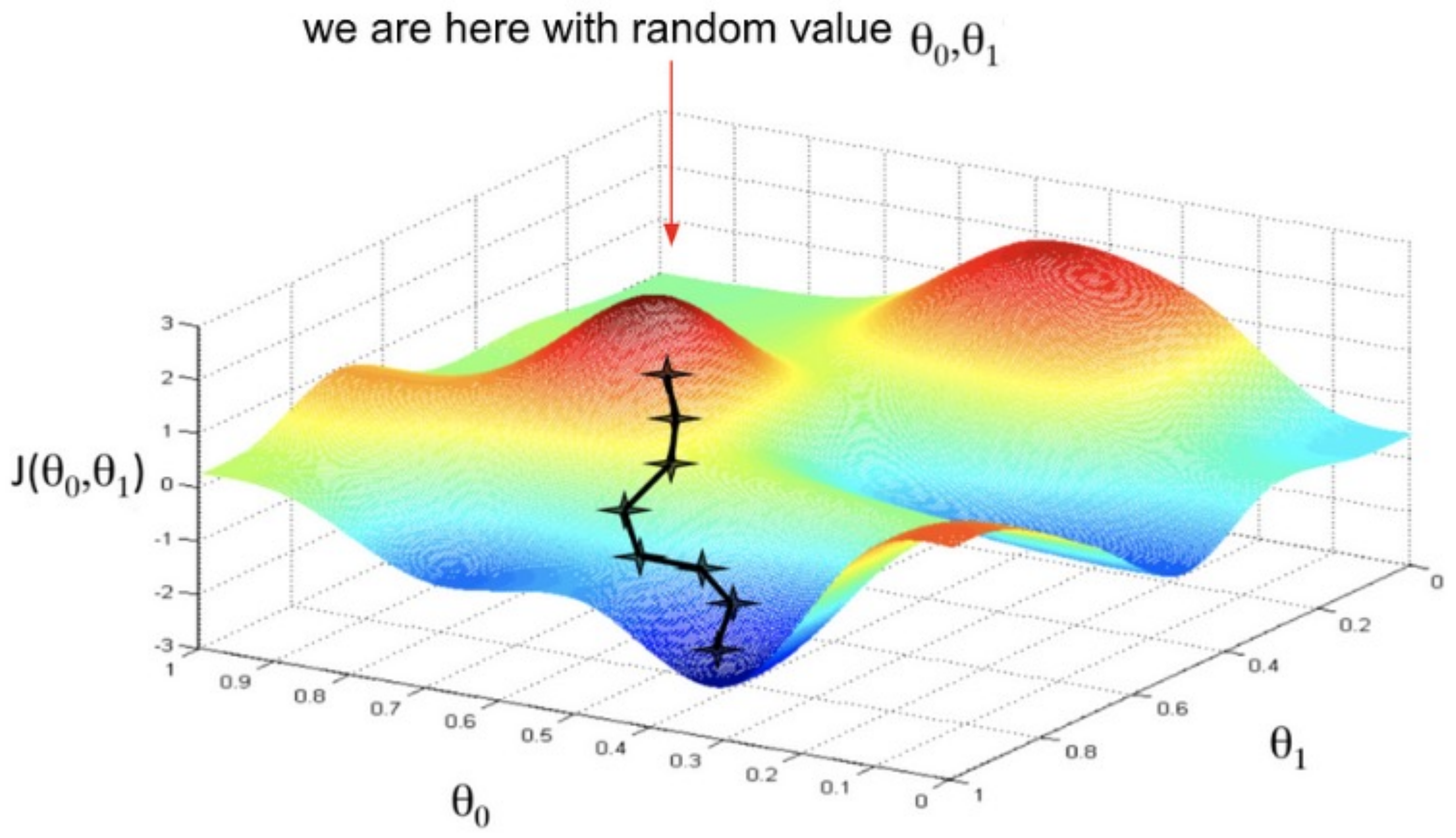
Activation Functions



$$\hat{y} = \alpha(W_L \cdot X_{L-1} + \beta_L)$$

How do we train neural networks?

- Trial and Error (almost) = Gradient Descent



Neural Networks solve non-linear problems!

How can I test ANNs myself?

What have we learned?

- What types of problems neural networks are best at: complex problems, with many feature interactions
- The mathematics of Feed Forward Neural Networks
- A step towards understanding more complex neural networks