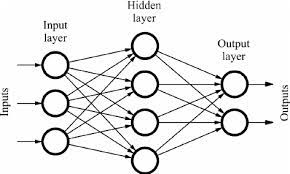
Henry’s Machine Learning Project

# The idea for the project:

The idea for this project is for me to learn more about the maths behind machine learning. I would like to implement my own library for the main functions of a simple feed forward neural network. I want to create efficient code that will allow for me to create a feed forward neural network of variable structure (number of layers and neurons per layer) and with different cost functions and activation functions. The program should allow for backpropagation and stochastic gradient descent to update the weights and biases. It should also allow for prediction.

I will start by examining a simple case where I have a neural network with 1 hidden layer.



(<https://medium.com/random-techpark/simple-feed-forward-neural-network-code-for-digital-handwritten-digit-recognition-a234955103d4>)

Let’s define some general variables to describe the network:

Let a be the number of input neurons, b be the number of neurons in the hidden layer and c be the number of output neurons. Each of these layers of neurons will have given activations. These activations can be represented as real row vectors with dimensions a\*1, b\*1, c\*1. The network could be viewed as having 2 non-linear transformations in it. Both transformations transform the activation vector from a layer the activations of the next layer in the network. These transformations could be denoted as T1 and T2. Each transformation has multiple components including a weights matrix, a bias vector and an activation function to provide nonlinearity. These are denoted by