**Machine Learning by Andrew Ng**

**Week 4 short notes**

Neural Networks model in which we model a Logistic regression problem as a neuron or a network of neurons. Where the dendrites are like the inputs and in case of one neuron system the features X0,X1….,Xn and the output is the hypothesis we intend to obtain to predict accurate results.

X0 is always taken as 1 and is often referred to as the bias unit.

The neural network consist of one input layer, one output layer and multiple or no intermediate layers known as hidden layers.

Here we use the same logistic function as in classification, ***g(θTx)*** , yet we sometimes call it a sigmoid (logistic) **activation** function. In this situation, our " ***θ*** " parameters are sometimes called "weights".

**a(j)i** = "activation" of unit *i* in layer *j*

**Θ(*j*)** = matrix of weights controlling function mapping from layer *j* to layer *j*+1

Note: If network has ***sj*** units in layer ***j*** and ***sj*+1** units in layer ***j*+1**, then **Θ(*j*)** will be of dimension

***sj+1 X ( sj  +1).***

Important Relations:

* **z(j+1)=Θ(j)a(j)**
* **a(j)=g(z(j))**
* **hΘ(x)=a(j+1)=g(z(j+1))**

These neural networks can be used to construct simple logic gates using appropriate parameters.

In case of binary classification the hypothesis gives a single output either a zero or one. When we extend the same to multiclass classification we expect the hypothesis to output a vector. Based on the subsequent activation equations the neural network helps us finalize the apt hypothesis and hence predicting the right category the given data falls.