Neural network endeavors to recognize relationship between a set of data through a process. It designs such a way that it can mimic human brain. In other word it has neuron that replicates human thinking process.

**Overview**

Neural network is generally multilayer to classify series of data. It generally has 3 layers as followed :

1. Input Layer
2. Hidden Layer
3. Out Layer

Every layer is connected to other layer.

To understand this let’s start with single neuron:

Prediction

-===

Weight (B1)

x

X is our input layer and B1 is slop estimator of logistic regression . In hidden layer we sigmoid function for activation. Let assume B0 is our bias. So Our predicted probability will be

Predicted probability=sigmoid (B1\*x+B0)

For multilayer:

W1,1

W2,2

W1,2

W2,1

In2

Output

In1

Here w1,1 means weight that associates with input1 and neuron1 .

We have in the above picture 2 inputs and 2 neurons . Output of the hidden layer will be calculated as :

Z1=W1,1\*In1+W2,1\*In2+bias\_n1

Neuron 1 activation =sigmoid (Z1)

So we can summarize this as

[W1,1 W1,2 \* [x1 + [b1 =[z1

W2,1 W2,2] x2] b2] z2]

In our project we are assigned to train backgammon game .

So we can the divide our project in to 2 parts :

1. Data generation
2. Train neural network with generated data.

Data generation

To train model first we need to identify features on which model will train upon . We have identified 31 features which are essential to train our model. They are given as followed

1. Number of checkers on each room of the board(as board has 24 room we counted 24 features and indicate as R2-R25 in CSV )
2. Number on rolling dices(2 features for each dices indicate as R30-31 )
3. Number of checkers on jail(2 player so 2 features indicate as R26-27 in CSV)
4. Number of checkers on home(2 player so 2 features indicate as R28-29 in CSv)
5. R1 is nothing but starting index.

Here we train our model observing each move made by each player after rolling dices. We store all data in csv file (comma separated file) . Here positive number indicates first player and negative number indicates second player . For example

If column 1 indicated as R1 has value 2 which means in room1 player 1 has 2 checkers and if it has value -3 means in room1 player2 has 3 checkers. Player 2 indicates computer it self . So whenever player 2 has all of its 15 checker on R29 which means computer/player2 wins and is classified as 1 otherwise if pc loses classified as 0 .

That is how we generate data .

Training part:

After generating data we have designed our neural network as followed:

1.Input Layer (neuron number 31 as we have 31 features)

2.Hidden Layer: We have 5 hidden layers

3.Outpur Layer: We one put layer as we have binary result i.e 0 or 1.

Function used : We use sigmoid fuction at output layer for classification .