**Plot generated after neural network run-**

As seen below this is a 4layer neural network with 2 hidden layers 1 input and 1 output layer each.   
The input layer consist of 4 inputs of data sets- x1,x2,x3,x4  
And these are processed at 4 input nodes where their computation is done based on computation formula. At next layer that is hidden layer consists of 2 nodes – A & B which receive input from the output of the previous layer to process the data. On these nodes there is an external input called weights added for the additional computation and activation to process accurately the results derived from the network.   
The activation function used here is sigmoid function that computes the hidden layer node value at Node A and node B in layer 1 and for node C & node D for layer 2. The given Weights are entered to the R code in a matrix which has to be processed with each of the 4 input node entry. The final node Y which receives output from hidden layer 2 output processed with a weight input and computed with sigma function to reduce to the prediction result as required.   
On completing 3788 steps the result generated an error of – 0.347508, this is further used to reduce the error in backpropagation task where the learning rate and weights are tweaked as per the changes made in the back propagation error correction.

Chart, diagram, radar chart

Description automatically generated

**Project report**

Team 1

Members –

**Challenges faced –**

1. Planning to choose which coding language to use – R or Python
2. Defining a the dataset input and restructuring according to the model format.

**Skills Developed during course that helped to complete project-**

1. Understanding the problem statement
2. Application of specific model to the problem statement
3. Solving the problem with raw dataset in an excel sheet in a stepwise manner
4. Coding to implement the model in real-time

**Coding language or the technology (e.g., Excel, Python, C++, ...) you used to fully train the network**

**Steps -**

1. Taking the input data
2. Making a prediction
3. Comparing the prediction to the desired output
4. Adjusting its internal state to predict correctly the next time

**Result Summary-**

**Final weights**

**Fully trained network**

**Training time**

**Other factors-**

summarizes your findings including the final weights in the fully trained network, training time, and other interesting factors.