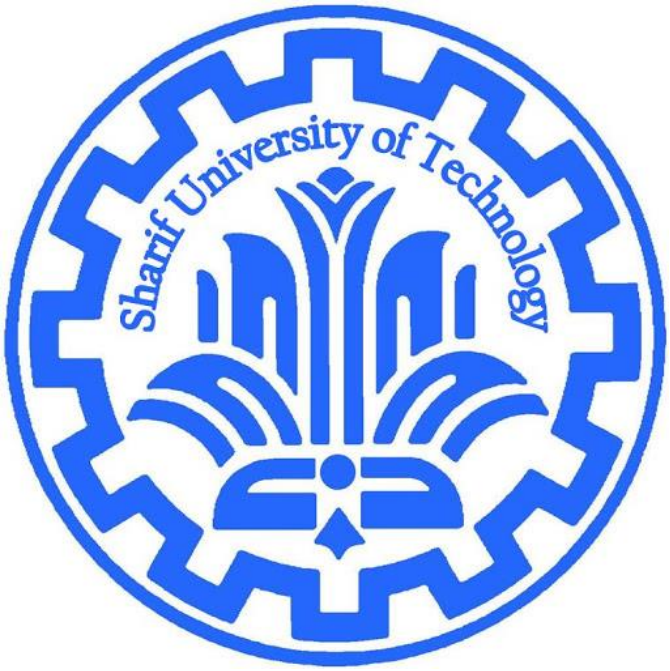
In the name of God



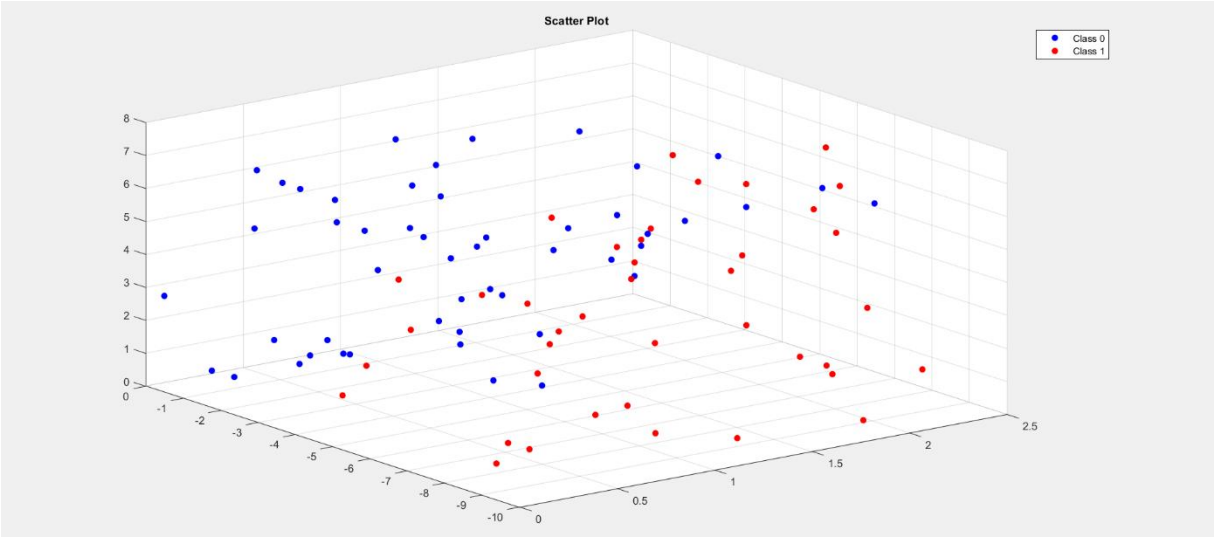
Computational Intelligence Ex2 Report

Ahmadreza Tavana

Student Number: 98104852

Part a)

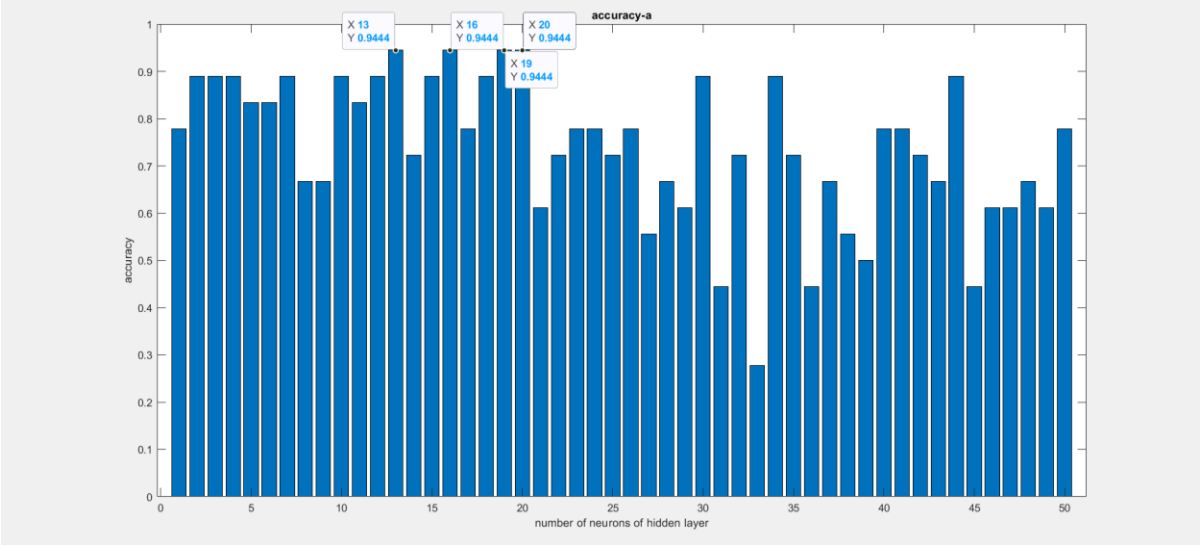
At first I draw the data in 2 different color and plot it with scatterplot3.



For this part I select first 72 data of the hole data set as the train data and the last 18 data of the hole data set as the validation data. Then I train a NN with different number of layers with a hidden and an output layer. In this part I tried many values for number of neurons for the hidden layer from 1 neuron to 50 neurons and calculated the accuracies of our models for each number of neurons in a for loop with 50 times of repeat and then calculated the mean of accuracies and as my calculations I came to realize that a hidden layer with 13,16,19 and 20 neurons has the best accuracy and hidden layer with neurons between 13,16,19 and 20 worked very well and with increase in number of neurons in hidden layer the accuracy decreased because the model had overfit on the data. (but you have to pay attention that number of neurons and accuracies change in each different runs.)

For fitting the model at first, I normalized the train data between -1 to 1 for using **tanh** function as my activation function for neurons in the hidden layer and then with use of **feedforwardnet** and for classification I put the threshold of classifying between to classes equal to 0 so if an output is bigger than 0 its label is equal to one and wise versa and then I calculated the accuracy and save the labels of test data in "**Testlabel_a.mat**".

I put the plot of accuracies for each number of neurons in below for the first model:



Part b)

For fitting the second model with 2 outputs, at first, I do the same thing in data and separate them to train and validation as the first part (actually I used them again) and fit the model and calculated the accuracy of the model for numbers of 1 to 50 and try to find best mean of accuracy in a for loop of 50 repeats and as the first one I came to realize that a hidden layer with just 1,2,7,8,13,16 and18 neurons has the best accuracy and with increase in number of neurons in hidden layer the accuracy decreased because the model had overfit on the data. For example, with layer of 47 neurons we have very low accuracy. (but you have to pay attention that number of neurons and accuracies change in each different runs.)

For fitting the model at first, I normalized the train data between -1 to 1 for using **tanh** function as my activation function for neurons in the hidden layer and then with use of **feedforwardnet** and then calculate the accuracy and save the labels of test data. For putting label in outputs, I compared the values of each 2 outputs of the model and put the label of that one which was bigger than the other one and then classified the test and validation data and then I calculated the accuracy and save the labels of test data in "**Testlabel_b.mat**".

I put the plot of accuracies for each number of neurons in below for the second model: