**MACHINE LEARNING FOR DATA SCIENCE**

**ASSIGNMENT No: 11**

**INSTRUCTIONS:**

Use neural network and see if you predict the type of faults in steel plates from numeric attributes only. [Note: To save time and energy use the hidden layer numbers, and number of nodes in hidden layers that your computer can handle].

**SOLUTION:**

The Steel Plates Faults Data Set used in the study comes from the UCI Machine Learning Repository. Steel Plates Faults Data Set is one of the datasets in the Repository, which classifies steel plates’ faults into 7 different types: Pastry, Z\_Scratch, K\_Scatch, Stains, Dirtiness, Bumps and Other\_Faults. The goal is to train machine learning algorithm to predict the types of faults in the steel plates.

ANNs are analytic techniques modelled after the processes of learning in the cognitive system and the neurological functions of the brain and capable of predicting new patterns (on specific attributes) from other patterns (on the same or other attributes) after executing a process of so-called learning from existing data. Multilayer Perceptron Neural Network (MLPNN) with back-propagation is the most popular ANN architecture. MLPNN is known to be a powerful function approximator for prediction and classification problems. There is at least one hidden layer, where the actual computations of the network are processed. Each neuron in the hidden layer sums its input attributes xi after multiplying them by the strengths of the respective connection weights wij and computes its output yj using Activation Function (AF) of this sum, in our case, a linear function.

**Steps to be followed:**

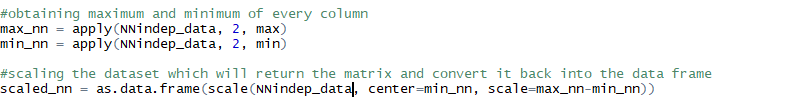
1. Start the process by loading the dataset in to the R studio.

2. Rename the headers of the columns to their corresponding meaningful names from the UCI repository.



3. As mentioned in the instruction, keeping only the numeric attributes for the purpose of our fault diagnostics.

4. Scaling the dataset to make sure that all the values of the dataset lie between 0 and 1.



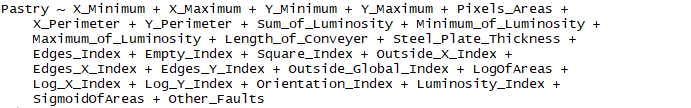
5. Split the records into training and testing

6. Splitting the variables in such a way that we keep only one dependent variable for case 1, the target variable is “Pastry”, the target variable

7. Collect the column names of the variables in the dataset and store it in a variable.

8. Now construct a formula in such a way that we can remove the target variable from the dataset and construct in a way that it can perform the neural network function on it.

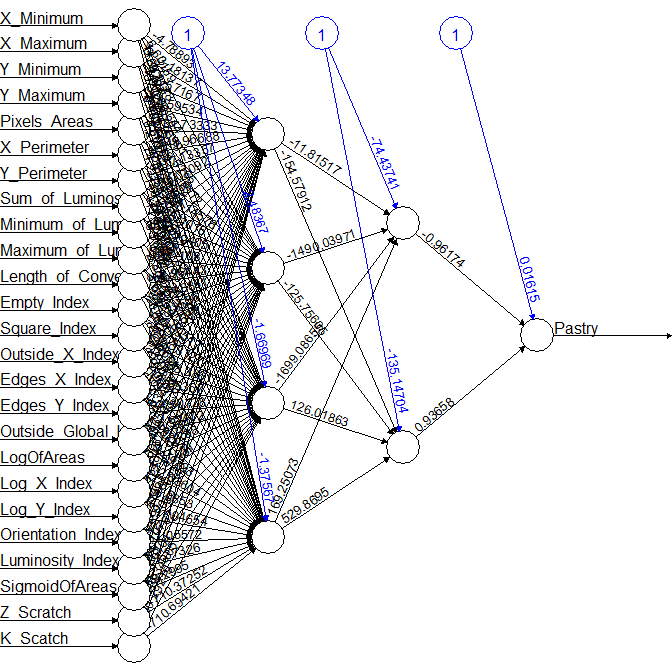




9. Unlike the linear regression, Neural network does not support the above input for its predictive analysis.

10. Perform the neural network on the training dataset for the target variable Pastry with two hidden layers and 4 nodes in the 2nd layer. and plot it to get the visualization





11. The above plot gives us the visualization of the neural network that we have created for predicting the Pastry, one of the faults type of steel plates.

12. The blue line indicates the bias line and the black line indicates the weight given.

13. This model, by default works with the back-propagation algorithm which implies

Pastry = (0.01615) + (-0.96174) \*(-744.3741) + 0.93658(-135.14704) +…..

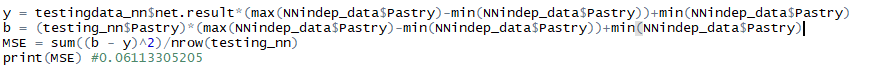
In the generalized form: y = w0 + w1\*x1+ w2\*x2+……..wn\*xn ======🡺 the linear function output for every layer.

14. The flows goes in the backward direction; every layer gets its input from its previous layer.

15. Thus the neural network will help to optimize and predict the target value as close as the actual values for every row by fitting the weights layer by layer.

16. Let us test the model on the testing dataset and check the Mean Square Error for its performance accuracy measure.



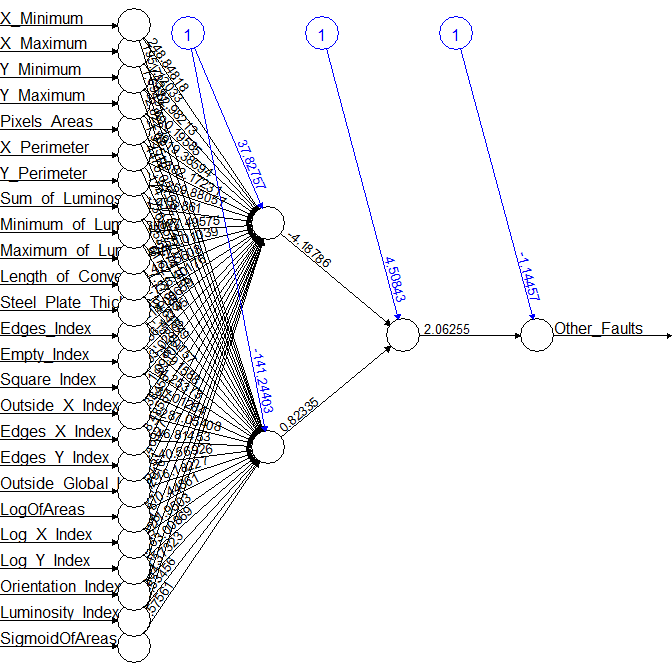


17. The Mean Square Error is calculated to be 0.06113305205 which is a very decent error size and we can say that our model is a good model.

18. The same procedure has been tried with the target variable as “Other\_Faults” and the hidden layer as 1 and two nodes and calculated the MSE with the data prediction on testing dataset to be 0.0498675 which also makes a good model.



19. Below is the visualization for the Neural network plot of the model with the target variables as “Other\_Faults”.



20. Artificial Neural Networks are designed to give us the prediction as close to the observed values, so it can be used as the most efficient machine learning technique in terms of accuracy.

**Conclusion:**

The Better model relies on so many factors like Data Preprocessing, Normalization, Cross Validation testing samples, and the number of hidden layers that we are setting in the neural network function. Therefore, we can conclude that the diagnosis accuracy may be limited by the hidden layer numbers and correlated training parameters of neural networks.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_