|  |  |
| --- | --- |
| Fundamentals of Neural Networks  **Assignment 1** | Abstract  To build a neural network using MATLAB to estimate the MPG (Miles per Gallon) based on some information about the car.  **Human Computer Interface [ HCI – 594 ]** |

Step 1: *To collect data*

* Take data from database <https://archive.ics.uci.edu/ml/datasets/Auto+MPG>
* INPUTS: Taking 5 inputs
* Car weight
* Engine displacement
* Cylinder count
* Horse power
* Hybrid or gasoline or Origin

OUTPUT : To calculate Miles per gallon [MPG]

STEP 2 : *TO DIVIDE DATA FOR TRAINING AND TESTING THE MODEL*

Overall Data count : 398

Training data = 70%

= ( 398 \* 70 )/100

= 278

Test data = 30%

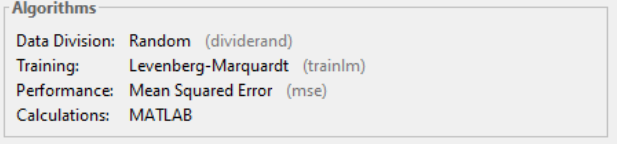
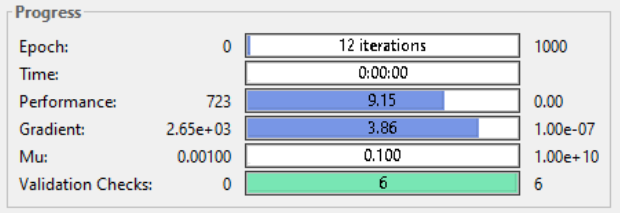
= 398 – 278

= 120

STEP 3: TO CHOOSE ALGORITHM

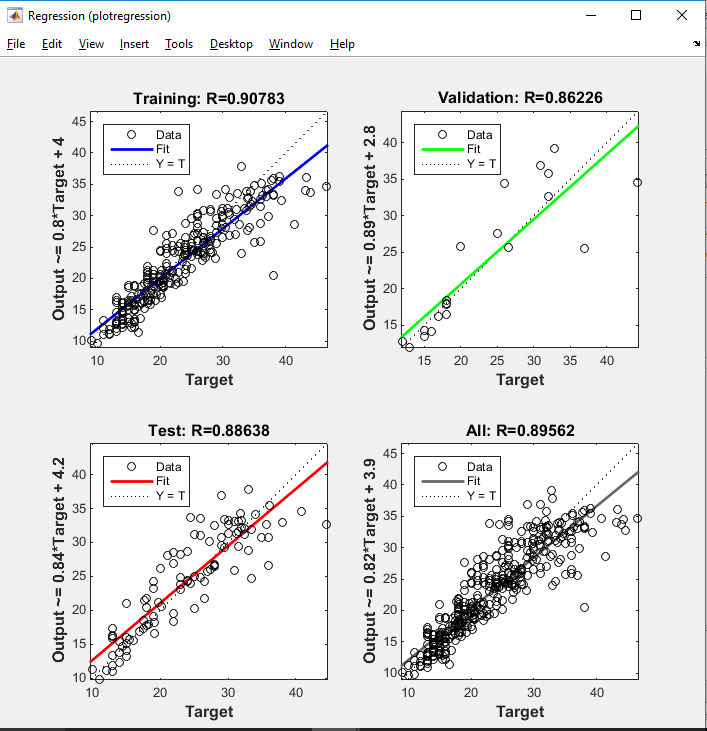
Algorithm used: Levenberg-Marquardt

Hidden Layers: 20



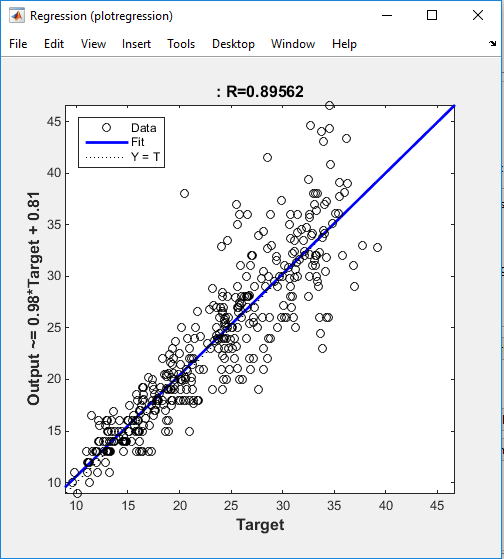
STEP 4: TO TRAIN THE NEURAL NETWORK

* Feed the 70% of data to the network which contains both input and target variable.
* Each input maps to an output
* The neural network [NN] , learns the pattern of mapping from input to the output.
* Check the regression line using regression plot map.
* Regression line is 45 degrees inclined to the X axis and most of the data passes through the regression line.
* When the data is trained with all the above conditions, it produces output with accuracy more than 85%(approximately 90%) which is a very good neural network/model.
* If the NN produces less accurate models, then we can train the network by increasing the number of hidden layers.

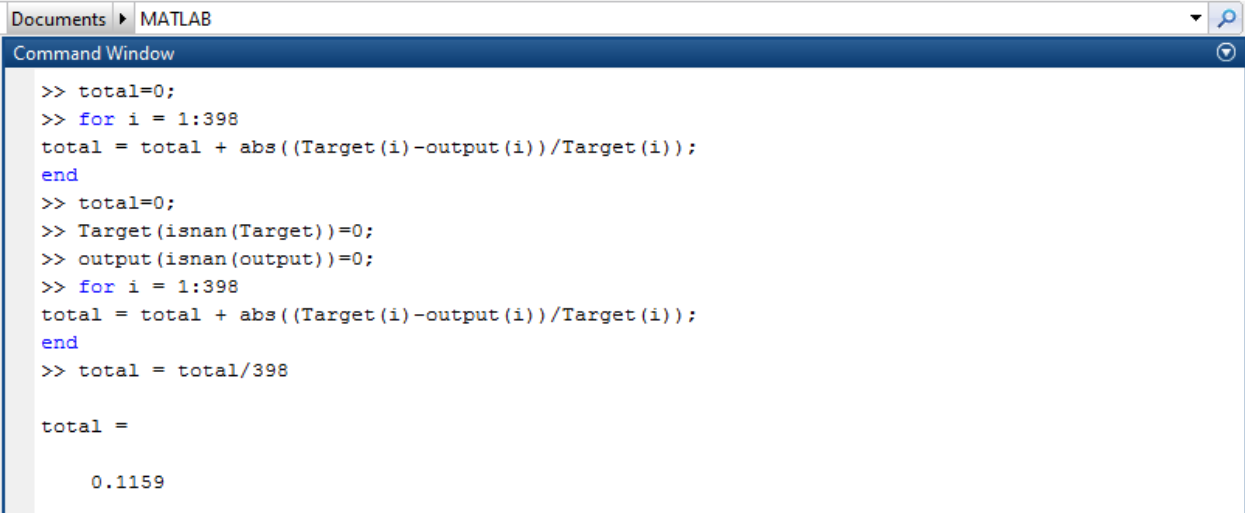


STEP 5: To TEST THE REMAINING 30% OF DATA

* Remaining 30% of the data is used for testing.
* For the above mentioned conditions ( no of hidden layers , no of neurons used) when the model is simulated with the remaining data , it produces a result with 89% accuracy.
* The data fits the model with very less errors.



STEP 6: ERROR CALCULATION



* Error percentage is calculated by using formula.

Total= Total + abs((Target(i) – Output(i))/Target(i))

Total=Total/398

Here ‘i’ is the counter