BAN 210 – FINAL ASSESSMENT

ANALYSIS ON THE AUTO MPG DATASET

BY-

STUDENT NAME: Rohan Sharma

STUDENT ID: 135812212

INTRODUCTION:

In order to estimate the class of the target variable in the Auto MPG data for the final project of BAN 210, I utilised predictive modelling in the research to forecast the values of Target variable and also used linear regression and neural network models in the evaluation below. I also conducted a study to determine which model will produce the most accurate forecast.

OBJECTIVE OF THE ANALYSIS:

My analysis's purpose is to answer the following questions:

• Predict the value of our target variable.

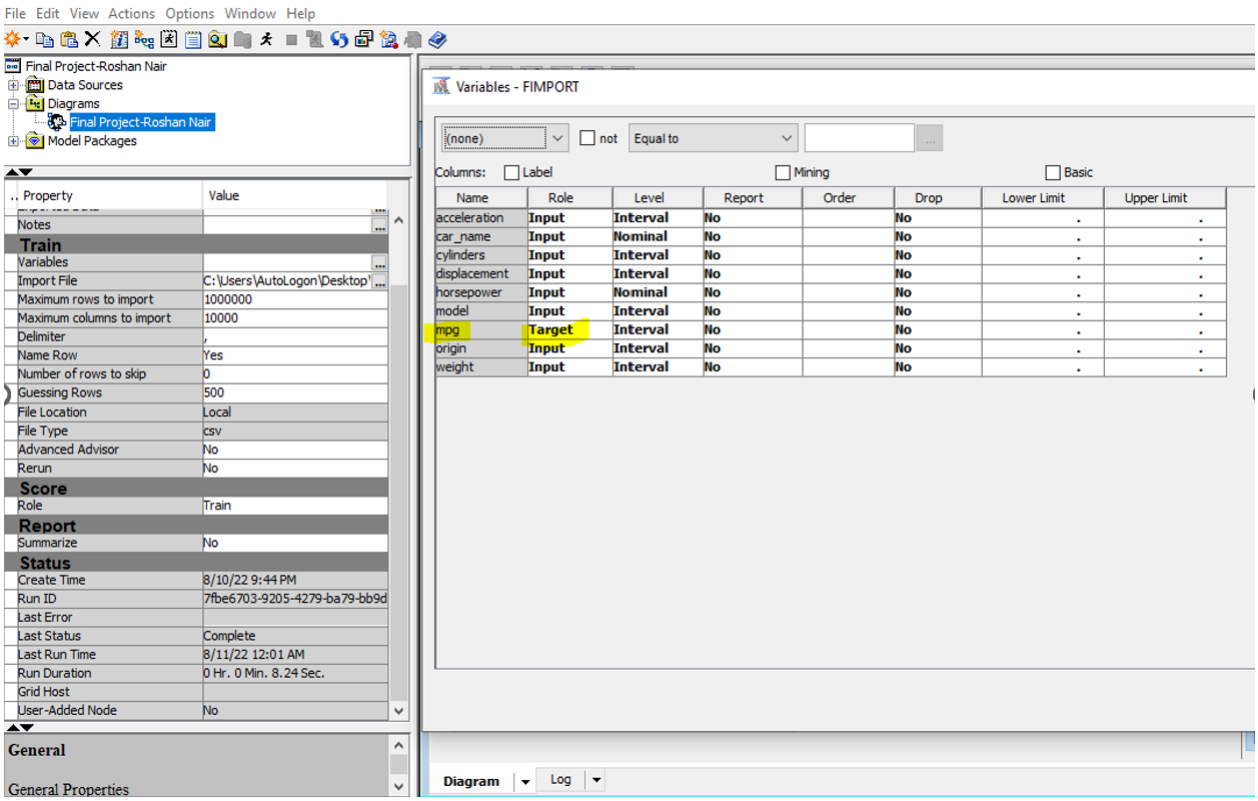
• Determine which model performs better and by how much accuracy.

METHODOLOGY AND INFERENCES:

Below are the steps I followed using SAS Miner to analyze the dataset:

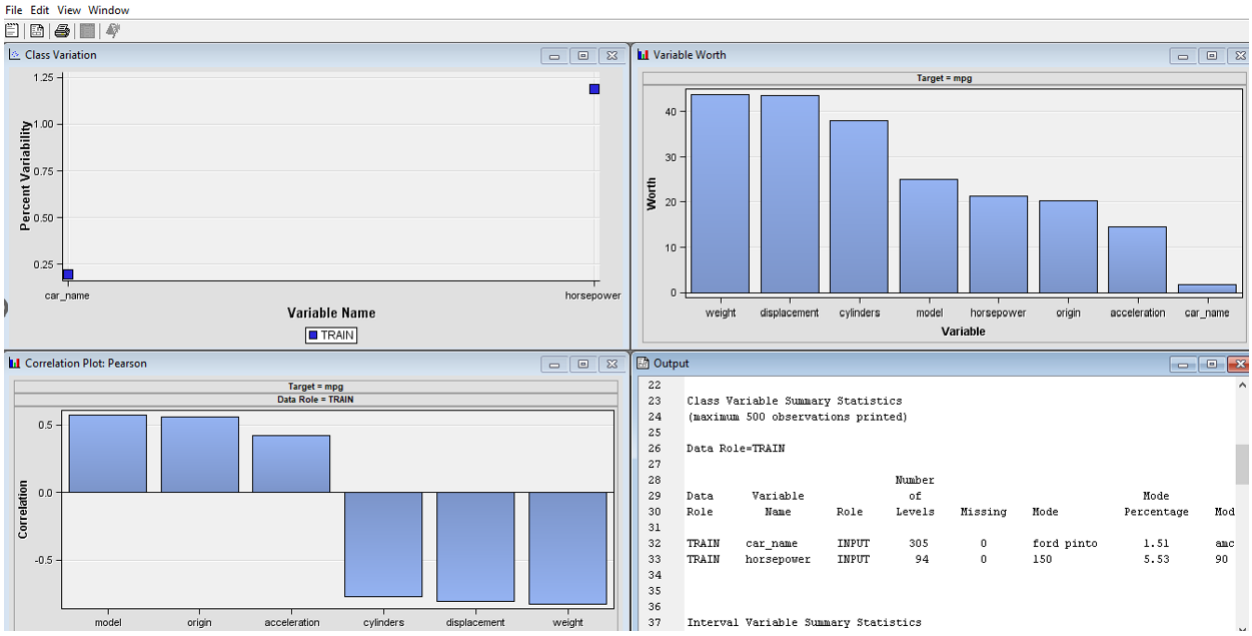
Step 1: File Import

In the first step, the dataset is imported and read on the user system using the "File Import" node. The file may be read by providing the path link on the "Import File" option in the node's Properties. Set Role as the "Target" for the Class variable in the Properties section by clicking the "Variable" button. The other attributes are known as "Input" variables since they are independent variables.



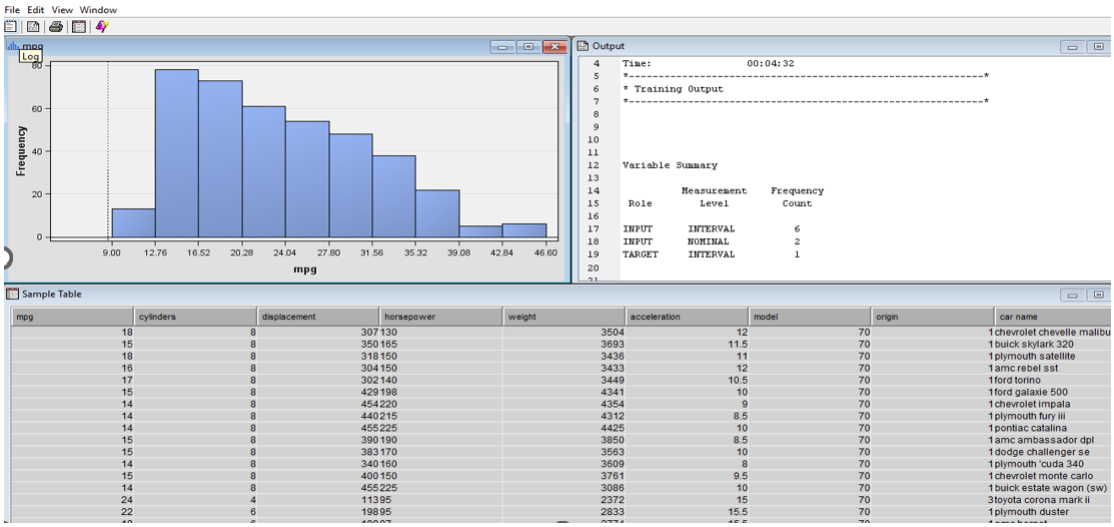
Step 2: Stat Explore

The StatExplore node is a versatile tool for examining variable distributions and statistics in our data sets. The outcome of the StatExplore node is shown in the screenshot below.



Step 3: Graph Explore

The Graph Explore node is a powerful visualization tool that allows us to graphically explore enormous quantities of data to identify patterns and trends and to highlight extreme values in the database, and the output that we obtained is seen below.



Step 4: Data Partition

To avoid overfitting and underfitting, divide the data into 20% Validation and 80% Train datasets.

The population distribution is depicted in the snapshot of the results window below.

Property	Value
General	
Node ID	Part
Imported Data	...
Exported Data	...
Notes	...
Train	
Variables	...
Output Type	Data
Partitioning Method	Default
Random Seed	12345
Data Set Allocations	
Training	80.0
Validation	20.0
Test	0.0
Report	
Interval Targets	Yes
Class Targets	Yes
Status	
Create Time	6/15/22 9:10 PM
Run ID	1b16a686-5b72-4848-b8
Last Error	
Last Status	Complete
Last Run Time	8/11/22 12:04 AM
Run Duration	0 Hr. 0 Min. 5.76 Sec.
Grid Host	
User-Added Node	No

Output

Summary Statistics for Interval Targets

Data=DATA

Variable	Maximum	Mean	Minimum	Number of Observations	Missing	Standard Deviation	Label
mpg	46.6	23.514572864	9	398	0	7.8159843126	

Data=TRAIN

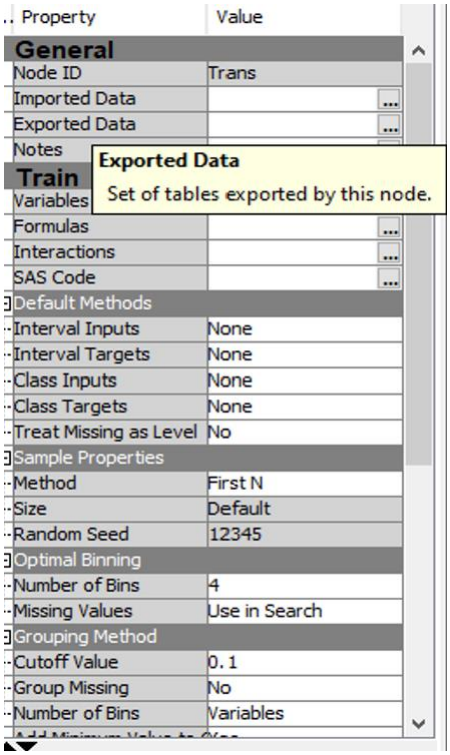
Variable	Maximum	Mean	Minimum	Number of Observations	Missing	Standard Deviation	Label
mpg	46.6	23.523899371	10	318	0	7.7904370653	

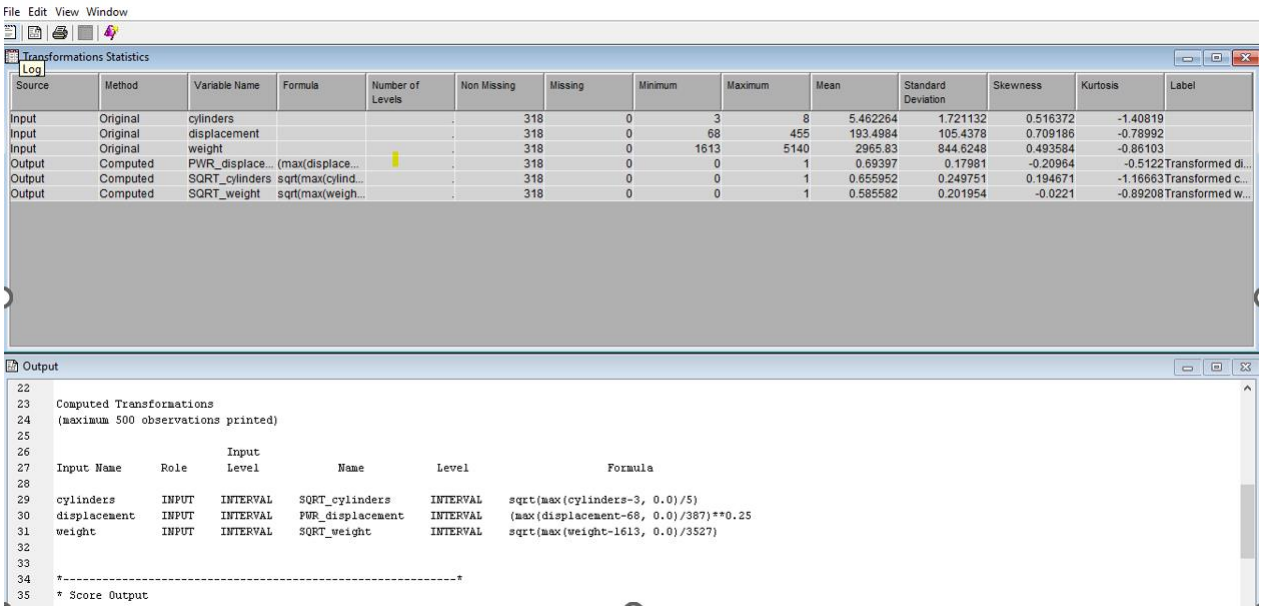
Data=VALIDATE

Variable	Maximum	Mean	Minimum	Number of Observations	Missing	Standard Deviation	Label
mpg	44.6	23.4775	9	80	0	7.966241669	

Step 5: Data Transformation

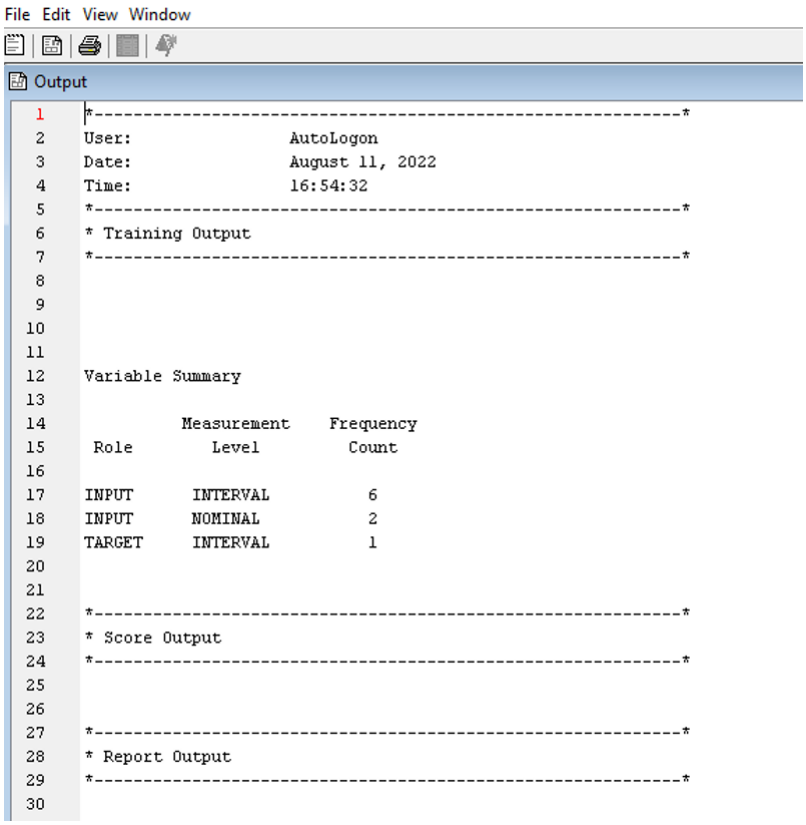
I utilised transformation nodes in this stage since they are important when we want to increase the fit of a model to the data. The output is as follows:





Step 6: Imputation

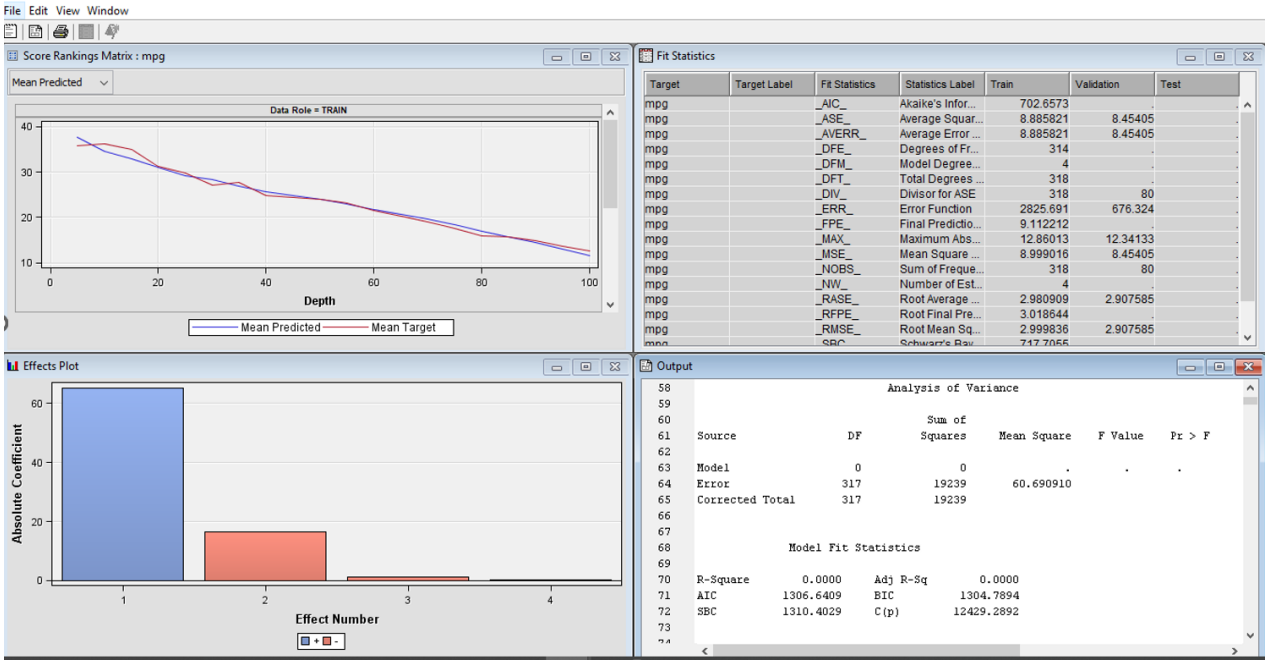
In this phase, I utilised the Impute node, which is used to replace missing values in data sets used for data mining. The output is as follows:

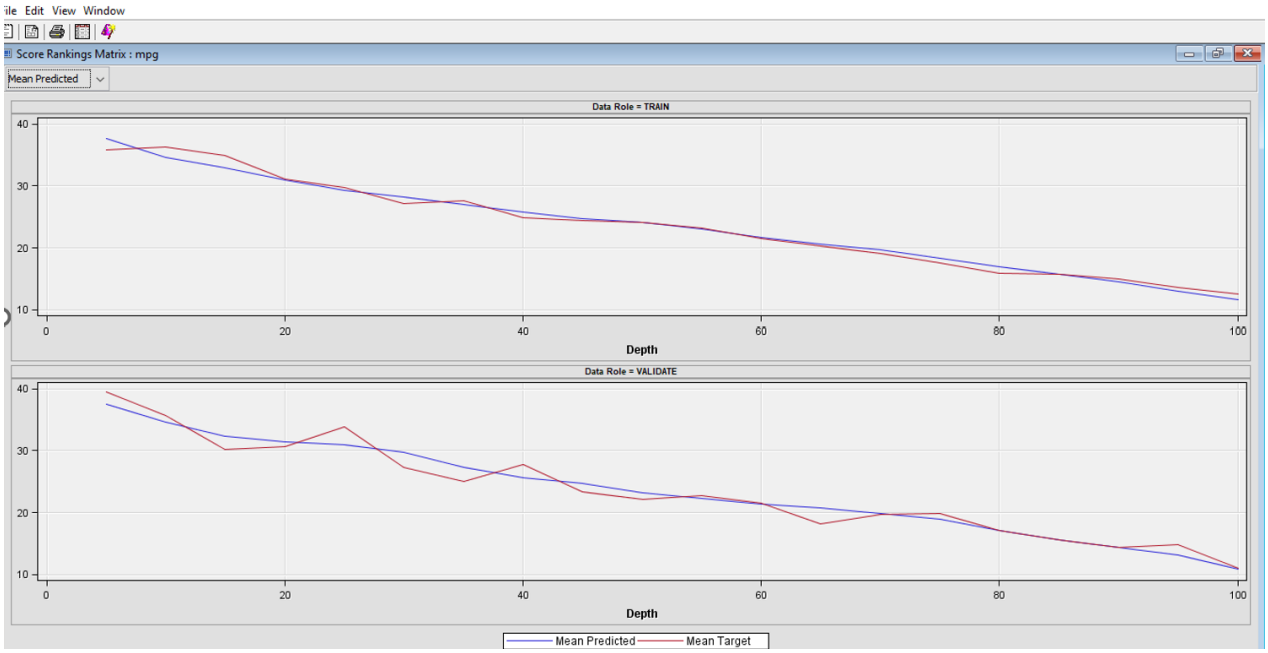


Step 7: Regression Model

Because we are predicting on a categorization variable, we have utilised the Linear Regression model. Logit has been selected under Properties after connecting the Regression node to the data Transformation node.

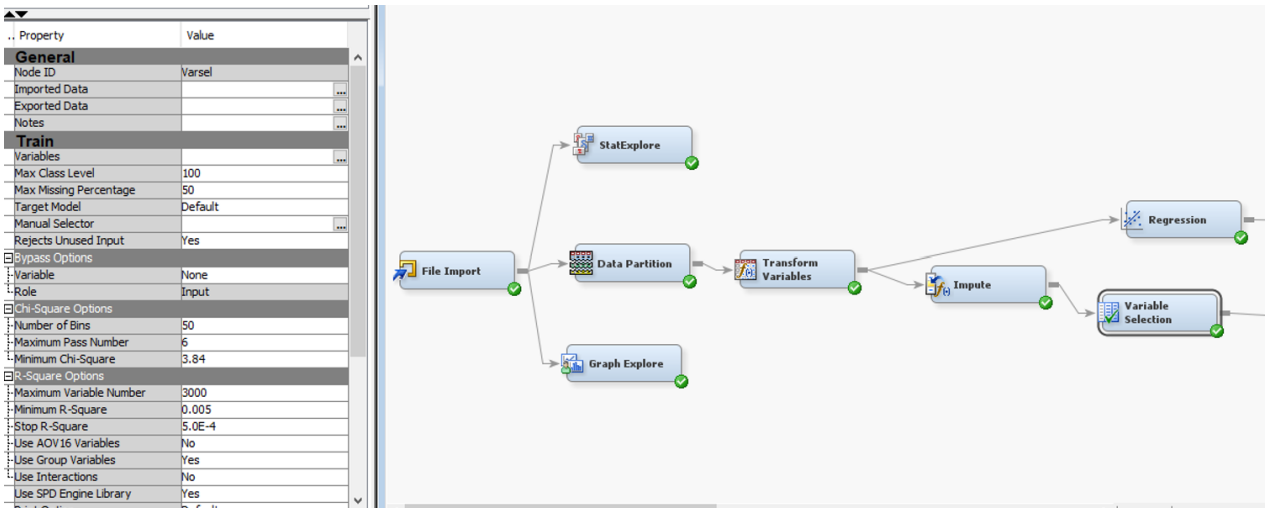
The Regression output yielded the following results:

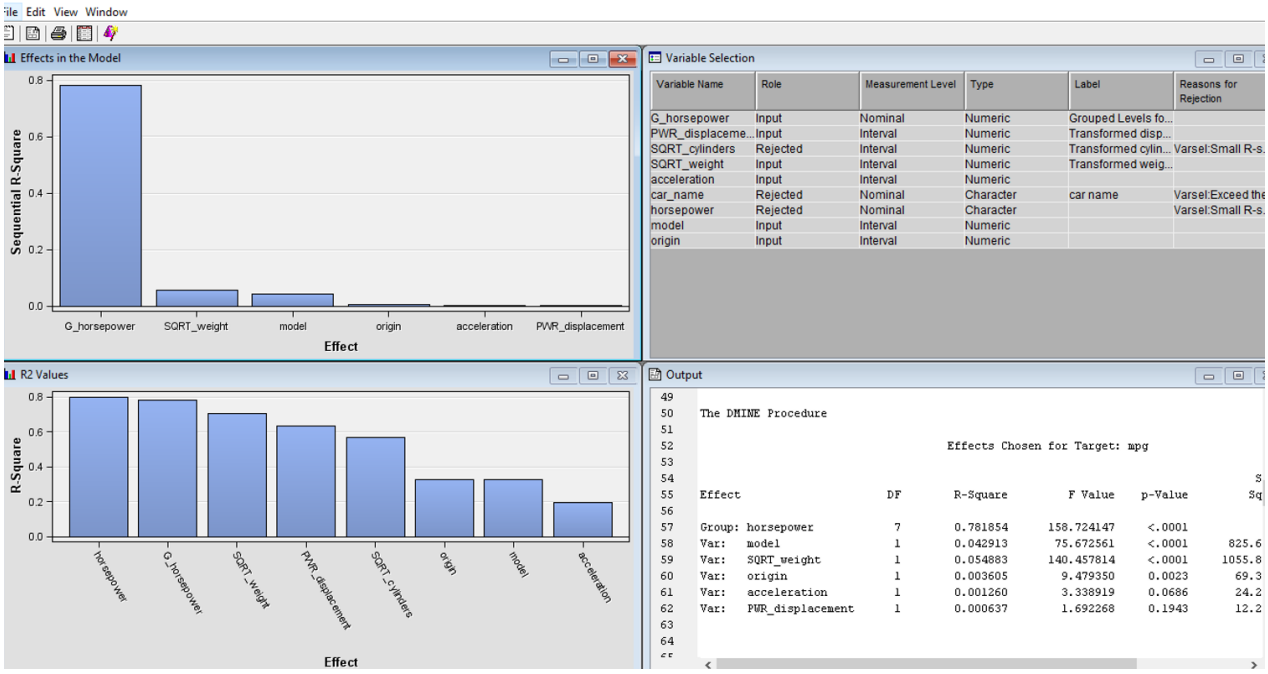




Step 8: Variable Selection

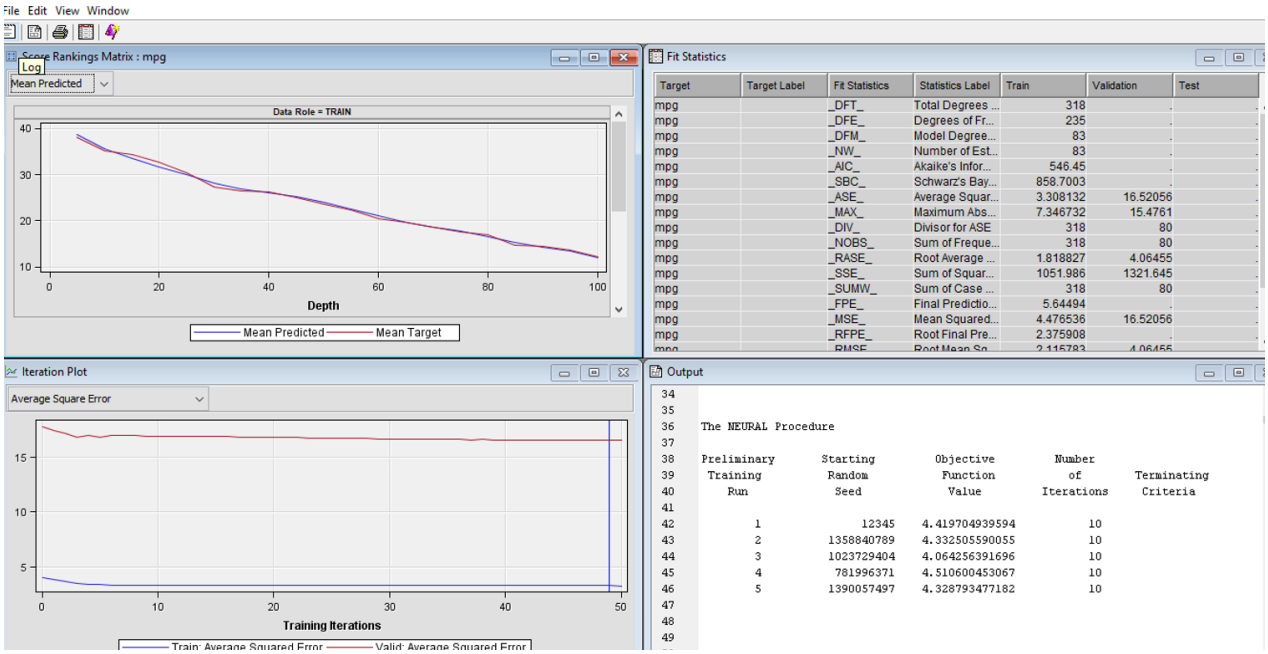
The Variable Selection node helps us reduce the amount of inputs by marking input variables that are unrelated to the objective as rejected. The output is as follows:

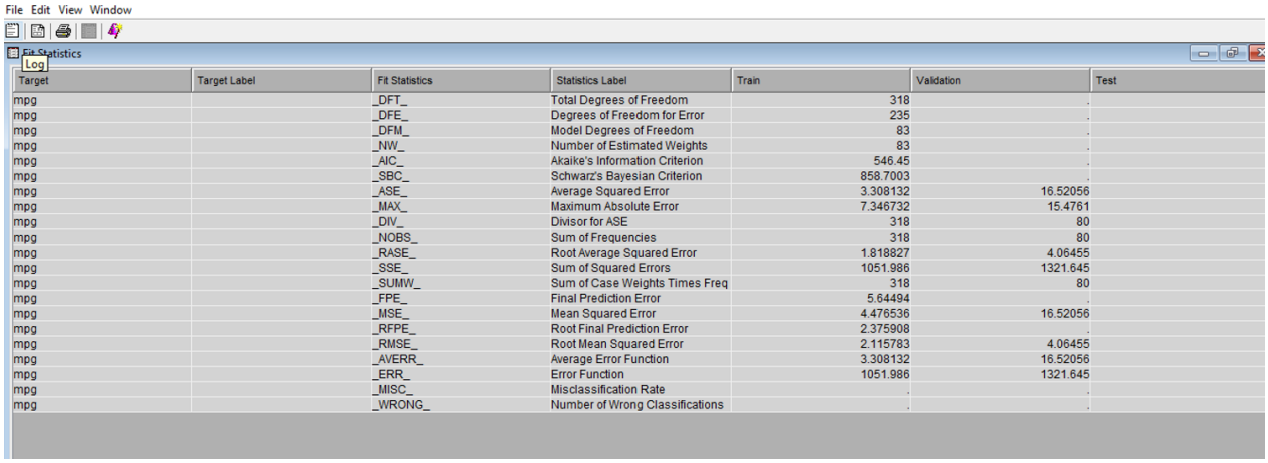




Step 9: Neural Network

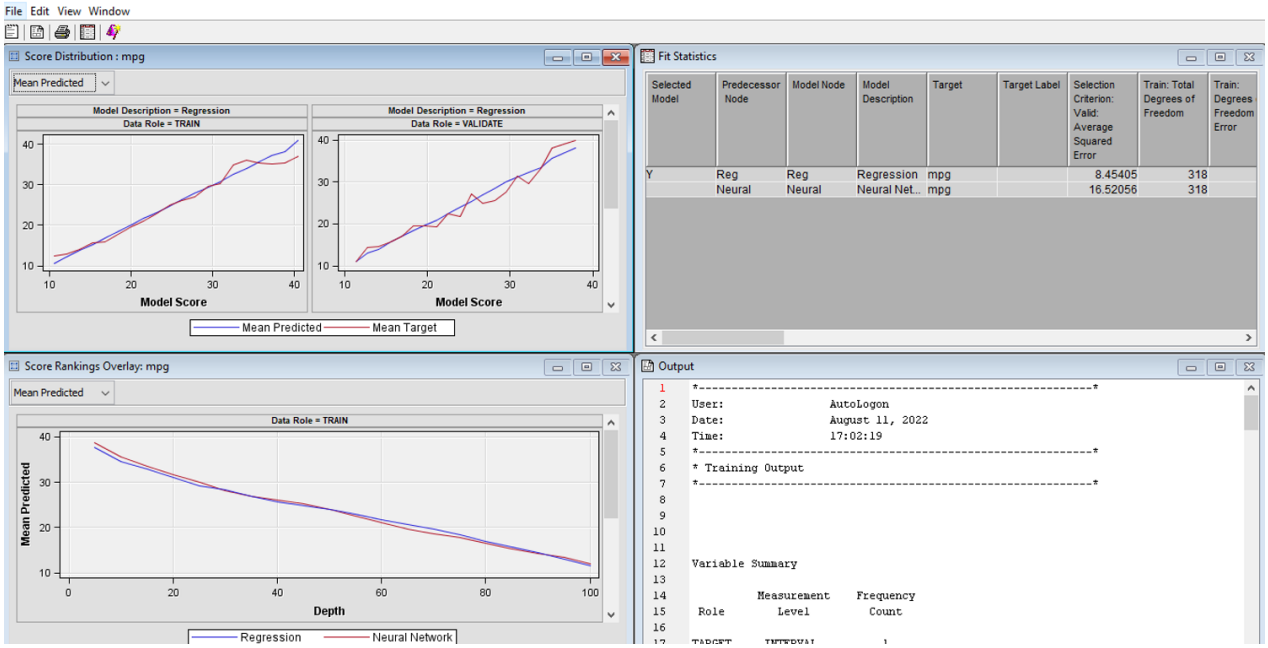
In addition to training, the Neural Network node allows us to score the training, validation, test, and score data sets. The following are the outcomes:

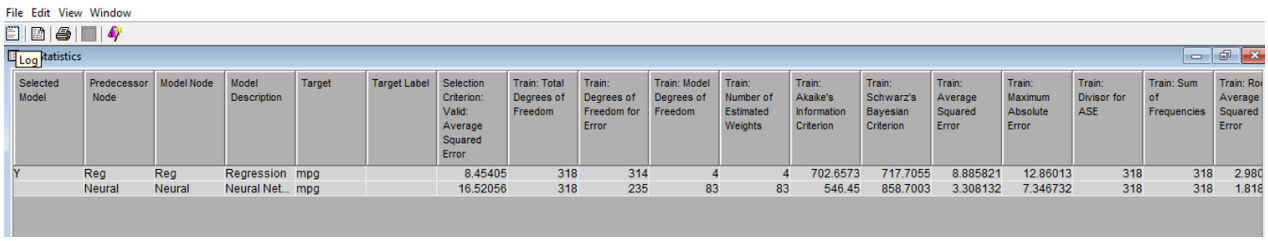




Step 10: Model Comparison

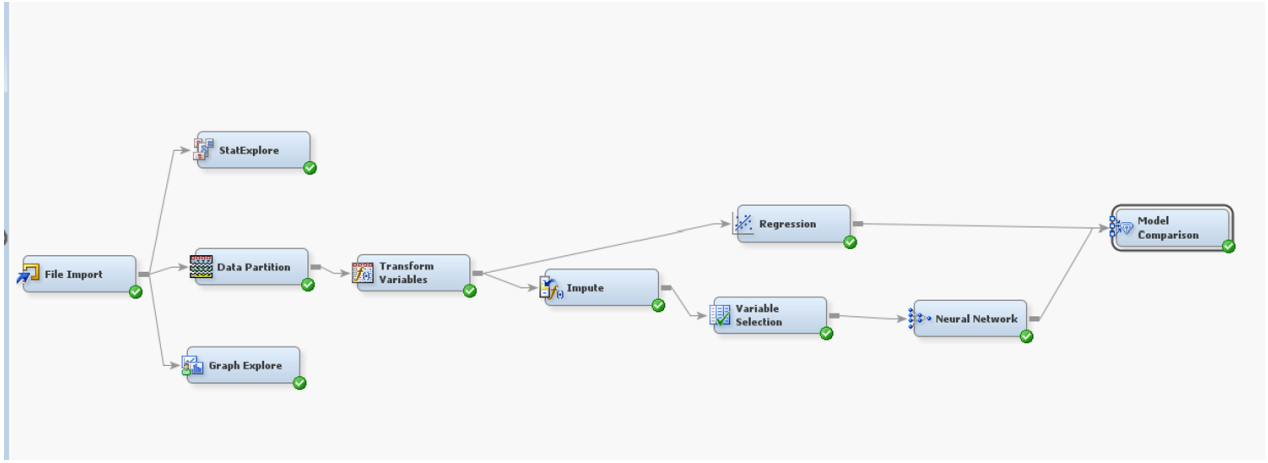
We may compare the performance of competing models using various benchmarking criteria by utilising the Model Comparison node. The following are the outcomes:





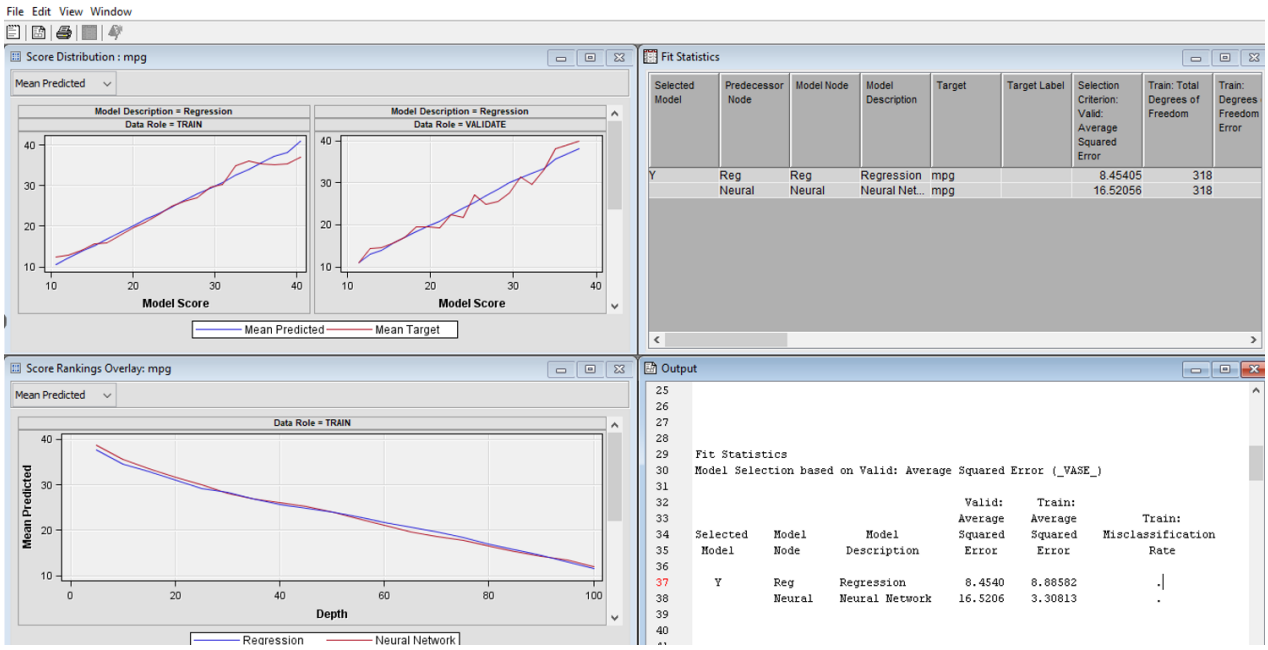
Step 11: Final Diagram

I received the following final diagram:



CONCLUSION:

I compared the two models by calculating the Mean Squared Error (MSE). Because the MSE Score of the Linear Regression model is lower than that of the Neural Network model, it outperforms the Neural Network model.



GITHUB LINK:

Please find below the GitHub:

DECLARATION:

I, **Rohan Sharma**, declare that the attached assignment is my own work in accordance with the Seneca Academic Policy. I have not copied any part of this assignment, manually or electronically, from any other source including web sites, unless specified as references. I have not distributed my work to other students.