DA 6223 Final Project

For the course project I have used KNIME tool to preprocess, analyze and run logistic regression on Auto data. KNIME, the Konstanz Information Miner, is a free and open-source data analytics, reporting and integration platform. KNIME integrates various components for machine learning and data mining through its modular data pipelining concept.

The primary goal is to predict whether a particular vehicle will give more mileage or not, based on the characteristics of the vehicle.

The snippet of the dataset is as follows:

Row ID	D mpg	I cylinders	D displac...	I horsep...	I weight	D acceler...	I year	I origin	S name
Row0	18	8	307	130	3504	12	70	1	chevrolet chevelle malibu
Row1	15	8	350	165	3693	11.5	70	1	buick skylark 320
Row2	18	8	318	150	3436	11	70	1	plymouth satellite

As seen above, there are various attributes of a vehicle including the number of cylinders, displacement, horsepower, weight, acceleration, year of its make, country code of origin and model name. These will be used as predictor variables to help predict the mileage a vehicle could yield.

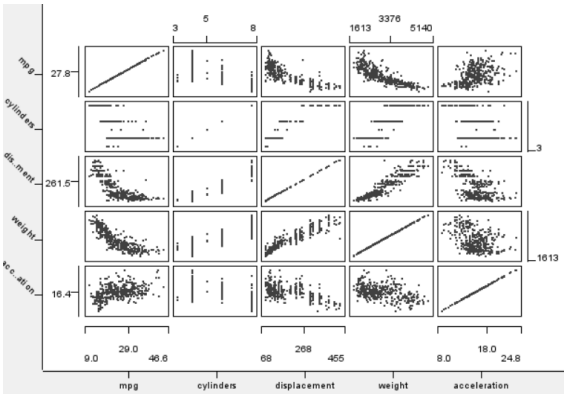
Data preprocessing:

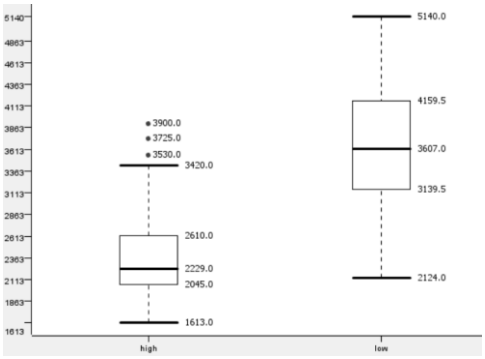
Before analyzing the various characteristics, we modify and manipulate the data to make it suitable for fitting models. The process can be listed out as below:

- Read the data through **CSV Reader node** and remove any missing values with the help of **'Missing Value'** node.
- Change origin variable from integer to categorical and give respective country names to the 3 codes **'Number to string'** and **'Rule Engine'** node. The latter is used to write a small IFELSE code to assign the numbers to country names.
- Since miles per gallon, '**mpg'**, is a quantitative variable and we want to predict it for any vehicle using logistic regression(which requires the dependent variable to be categorical), we change mpg to binary variable: for mpg above its median the value is 'high' otherwise 'low'. We use '**Math Formula'** and '**Rule Engine'** for this purpose.

Data Analysis:

In order to run any model and predict a dependent variable based on one or more independent variables, there needs to be a relationship between the two variables. Therefore, our next task would be to find the association between them using boxplots and scatterplots. This way we can determine which predictor would be more likely to be useful in predicting mpg.





From scatterplot using **'Scatter Matrix'** node, we observe that continuous predictor variables displacement, horsepower, weight and acceleration seem to be associated with mpg.
For boxplots of cylinders, displacement, horsepower and weight we use **'Conditional Box Plot'** node and observe that mpg is high for lower values of the variable and low for higher values. For boxplot of Year vs mpg, as the year increases mpg changes from low to high. For acceleration mpg is low for lower values of acceleration and vice versa enough though by slight variation.
From above observations we can conclude that cylinders, displacement, horsepower, weight, acceleration and year seem to be associated with mpg01.

Logistic regression:

To fit the model, we partition the data into 70% training set and 30% test set using '**Partitioning'** node. We next fit the train data using '**Logistic regression Learner'** node. From the 'Coefficients and statistics' results we find that horsepower and weight are significant and negatively associated with high mpg.
If we look at horsepower coefficient for example, we interpret it as follows: For every one unit increase in horsepower, we expect a 0.061 decrease in the log-odds of mpg.

Logit	Variable	Coeff.	Std. Err.	z-score	P>|z|
high	cylinders	0.362	0.509	0.711	0.477
high	displacement	-0.011	0.016	-0.719	0.472
high	horsepower	-0.061	0.029	-2.151	0.031
high	weight	-0.004	0.001	-2.529	0.011
high	acceleration	-0.068	0.166	-0.412	0.68
high	year	0.39	0.086	4.511	0
high	origin=Euro...	0.698	0.849	0.821	0.411
high	origin=Japa...	0.014	0.838	0.017	0.987
high	Constant	-12.51	6.335	-1.975	0.048

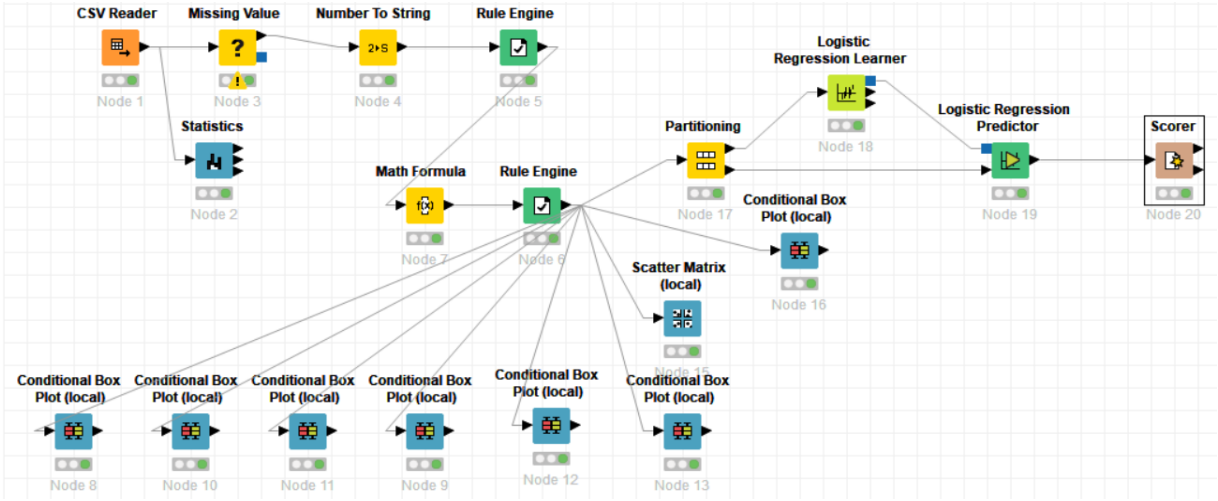
Then we predict the test data using '**Logistic Regression predictor'**. A new column is generated with prediction for test data. Then we use **'Scorer'** node to get the confusion matrix and calculate the accuracy statistics.

Row ID	low	high
low	52	7
high	2	57

Row ID	TruePo...	FalsePo...	TrueNe...	FalseN...	Recall	Precision	Sensitivity	Specifity	F-meas...	Accuracy	Cohen'...
low	52	2	57	7	0.881	0.963	0.881	0.966	0.92	?	?
high	57	7	52	2	0.966	0.891	0.966	0.881	0.927	?	?
Overall	?	?	?	?	?	?	?	?	?	0.924	0.847

The logistic regression model has 92.4% accuracy on test data i.e., is out of 118 data points, 9 have been misclassified and 109 have been classified properly.

Workflow diagram of the entire process



Review of the KNIME Tool:

The tool seems powerful and packed with almost all the necessary tasks like data manipulation, Analytics, DB, Scripting, Reporting, etc. for Data Analysis and modelling. Each of these tasks have different sections and subsections of a variety of functions/modules containing almost anything we can think of. I am amazed by the abundance of functionality and variety of processes that this tool offers. Also, there are many examples to help get an idea of the workflows and most importantly, there is a detailed description for each of the nodes. There are also various forums online, to help us solve any issues we face. Though I struggled to find the appropriate nodes for data cleaning and manipulation, later it felt easy to plot the data or run the models and view various results. Therefore, if intensive coding is required, like for data preprocessing part, the tool may become cumbersome, although it may not seem so once we get a hang of it.