# Introduction

Insurance firms employ car insurance claim analysis to determine the worth of a claim. Typically, the analysis includes analysing the claim's specifics, such as the type of damage, the age of the vehicle, and the circumstances surrounding the accident. It may also include an evaluation of any medical bills, missed earnings, or other costs related to the claim (McGuire, Taylor and Miller, 2018). Following the completion of the analysis, the insurance company will assess the level of coverage it can provide and the amount it will pay out to the policyholder. The study can assist insurance companies in determining how much to charge for premiums and in ensuring that they pay out the correct amount on claims.

# Problem Statement

Machine learning can be used to analyse insurance claim data and identify patterns that will assist insurers in better understanding the elements that drive claim prices. This can be accomplished by utilizing supervised and unsupervised learning algorithms to find patterns in consumer behaviour, claims history, and other pertinent factors. Clustering algorithms, for example, can be used to classify consumers into segments based on their claim’s histories, revealing customer segments that are more likely to make costly claims or have a higher risk of fraud. Then, using supervised learning algorithms, insurers may construct models that forecast the likelihood of a claim and its associated cost, allowing them to better assess risk and modify their policies accordingly. Furthermore, machine learning may be used to detect and flag false claims, reducing costs and protecting insurers from fraudulent conduct.

# Aim

The aim is to predict the Insurance claim of different cars with the help of different machine learning algorithms. The algorithms will be implemented using SAS and also different features will be visualized using Tableau to find important patterns related to claim of the insurance.

# Objectives

1. To develop a predictive model that can accurately predict the likelihood of an insurance claim being filed based on customer profile information.
2. To analyse the impact of customer profile features on the likelihood of an insurance claim being filed.
3. To identify the most important features that influence the likelihood of an insurance claim being filed.
4. To identify any potential patterns in the insurance claim data that could be used to improve the accuracy of the predictive model.

# Data Source:

The link of the data is given below

Datasets:

<https://www.kaggle.com/code/setumoraphelakamatlou/insurance-claim-xgbclassifer/data?select=train.csv>

# Dataset Description:

|  |  |
| --- | --- |
| **Column** | **Datatype** |
| policy\_id | int |
| policy\_tenure | float |
| age\_of\_car | float |
| age\_of\_policyholder | float |
| area\_cluster | char |
| population\_density | int |
| make | int |
| segment | char |
| model | char |
| fuel\_type | char |
| max\_torque | char |
| max\_power | char |
| engine\_type | char |
| airbags | int |
| is\_esc | char |
| is\_adjustable\_steering | char |
| is\_tpms | char |
| is\_parking\_sensors | char |
| is\_parking\_camera | char |
| rear\_brakes\_type | char |
| displacement | int |
| cylinder | int |
| transmission\_type | char |
| gear\_box | int |
| steering\_type | char |
| turning\_radius | float |
| length | int |
| width | int |
| height | int |
| gross\_weight | int |
| is\_front\_fog\_lights | char |
| is\_rear\_window\_wiper | char |
| is\_rear\_window\_washer | char |
| is\_rear\_window\_defogger | char |
| is\_brake\_assist | char |
| is\_power\_door\_locks | char |
| is\_central\_locking | char |
| is\_power\_steering | char |
| is\_driver\_seat\_height\_adjustable | char |
| is\_day\_night\_rear\_view\_mirror | char |
| is\_ecw | char |
| is\_speed\_alert | char |
| ncap\_rating | int |
| is\_claim | Boolean |

Total Variable: 44

Character Variable: 27

Integer Variable: 12

Float Variable: 4

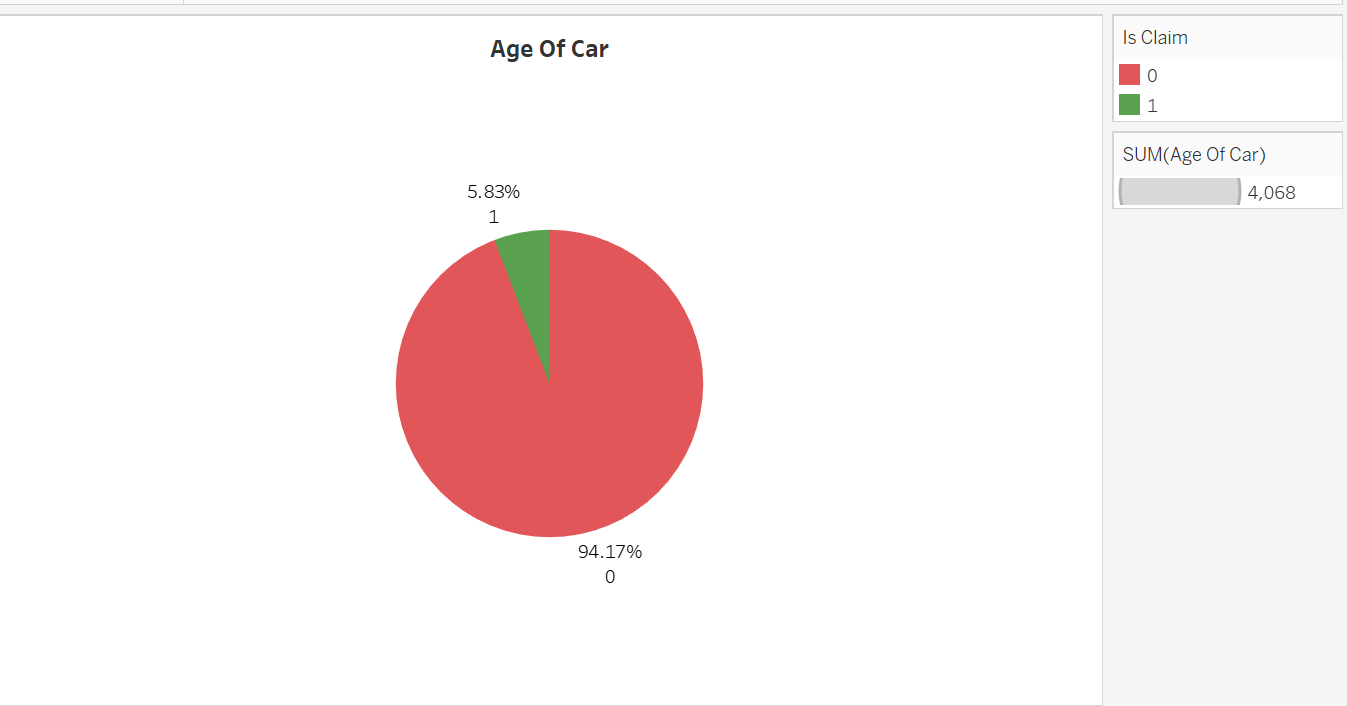
Boolean: 1

Date: 0

DateTime:0

Target Variable: Is\_Claim

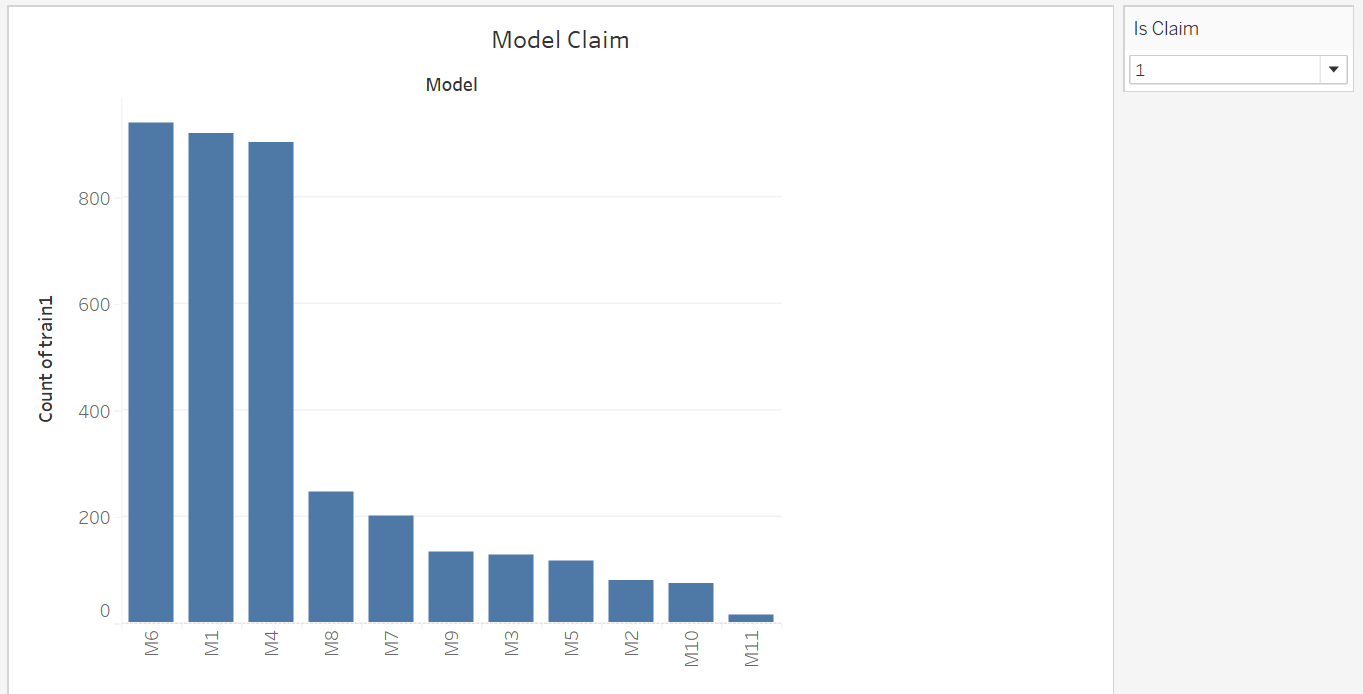
# Data Visualization

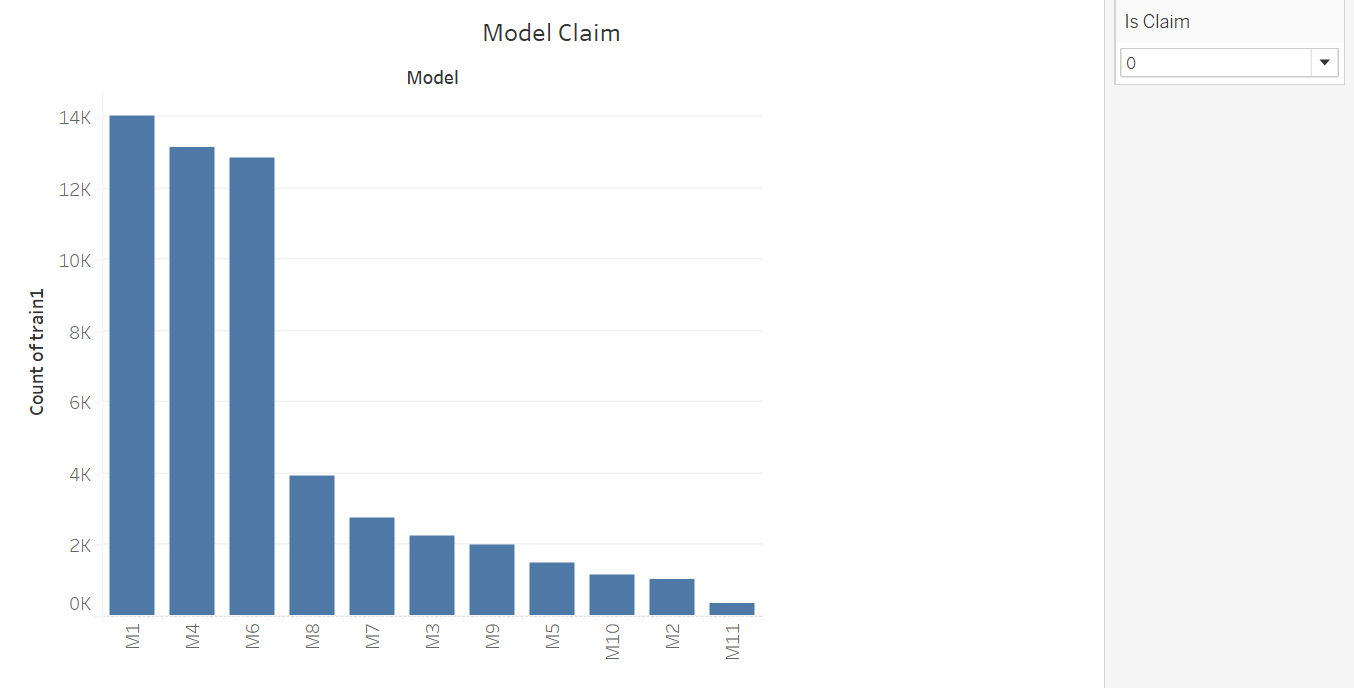
****

The first graph shows age of car and the claim settlement ratio so car with less age has high chances of getting claims.

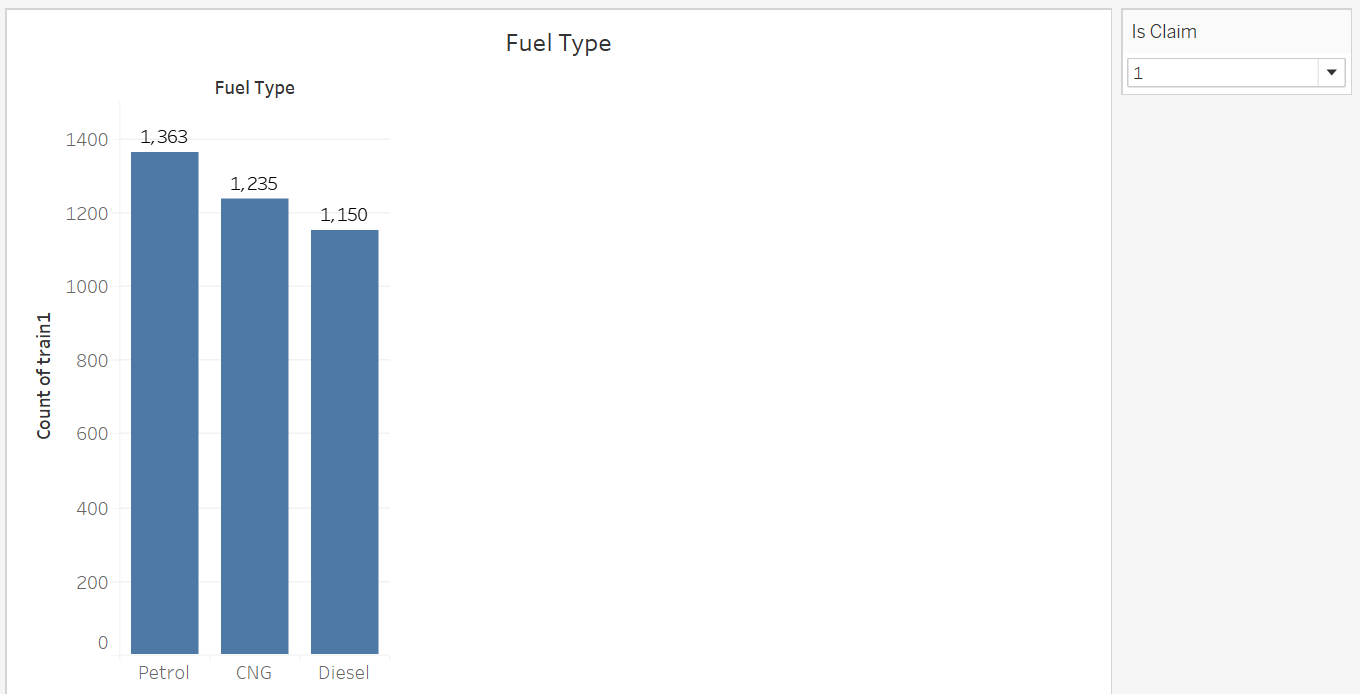


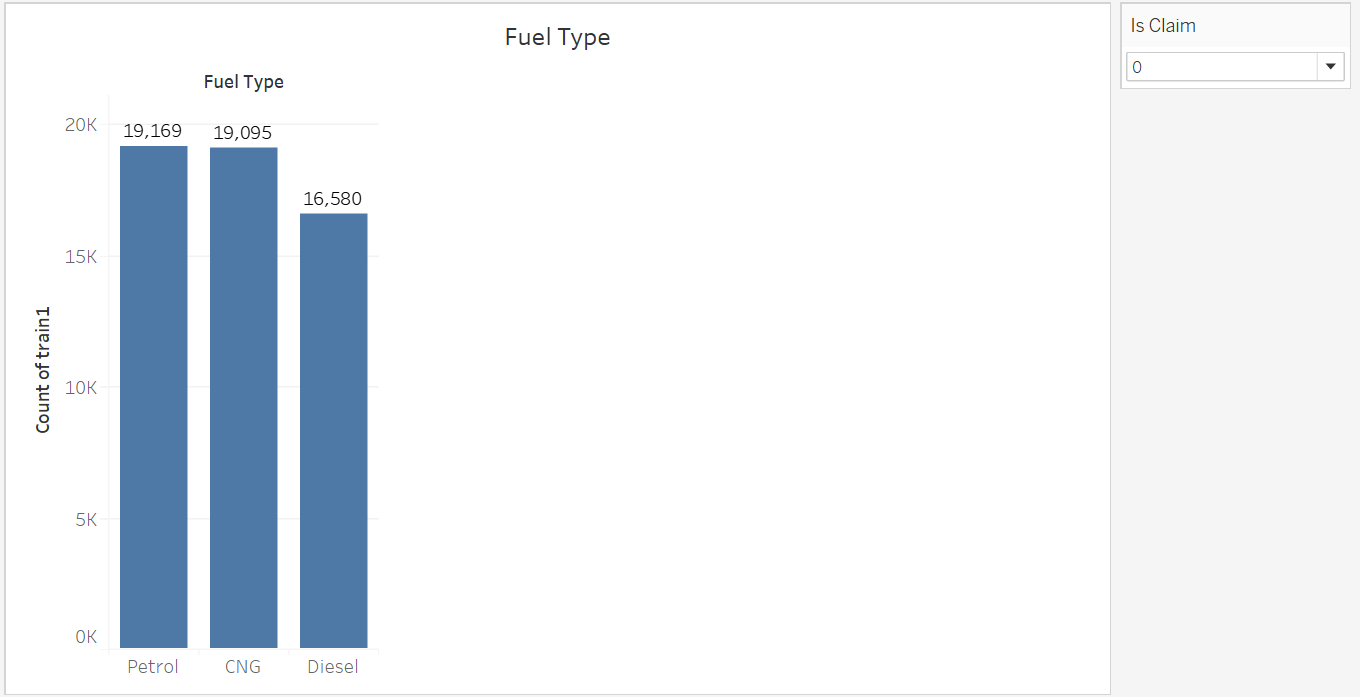
Having day night rear mirror doesn’t have that much effect on getting the claims settled



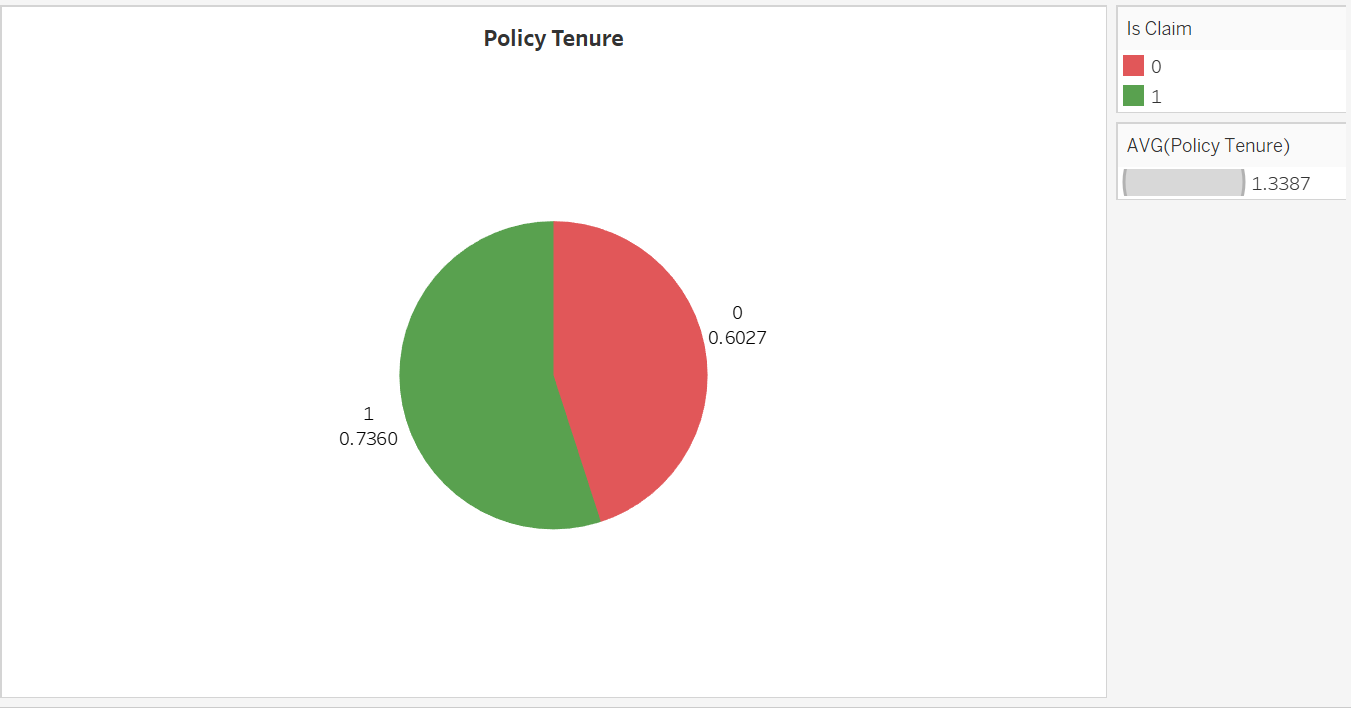


If we see both the data points it’s easily understandable using the graph that claim settlement is good with M6 model were for M1 its sort of both.

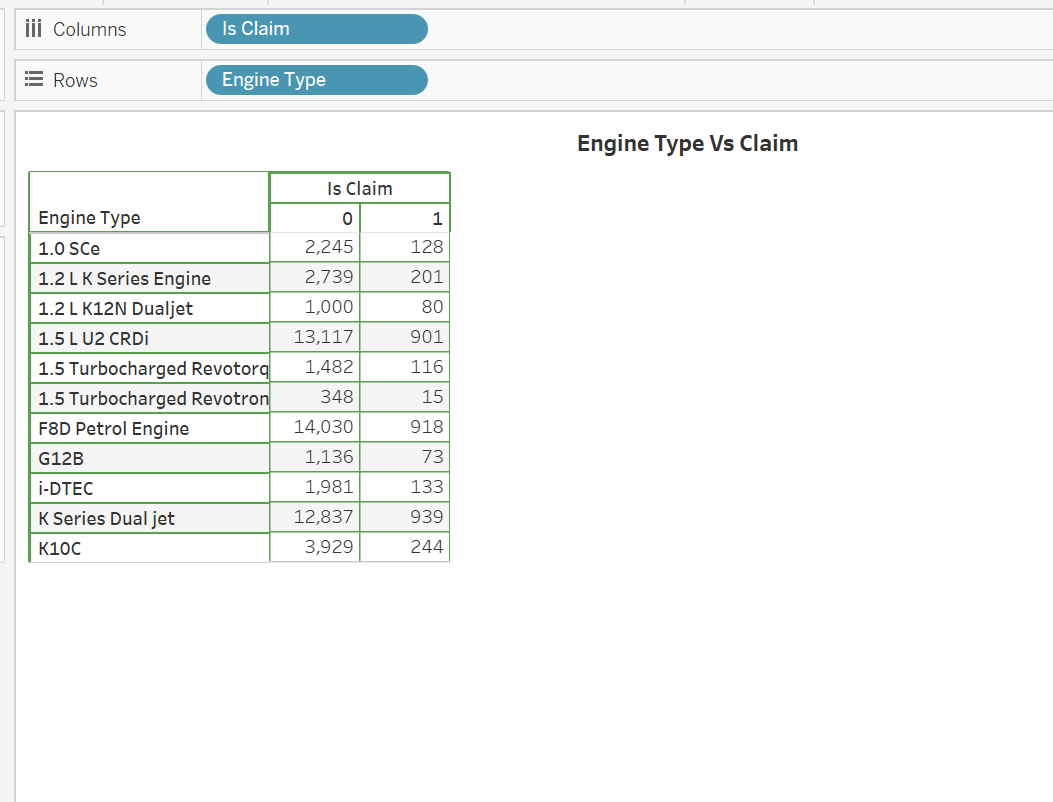




Claim settlement for Fuel type petrol is way better than other two.



In this graph we are show casing the policy tenure. We have taken the average of policy tenure as the policy tenure increase the chances of car claim increases.



The graph shows the Engine type study with respect to claims.

# Data Exploration and pre-processing

The dataset provided in the link is related to insurance claims, and it contains information about various attributes of insurance claims, such as the atransmission\_type, gear\_box, steering\_type, turning\_radius, length, width, height, gross\_weight etc

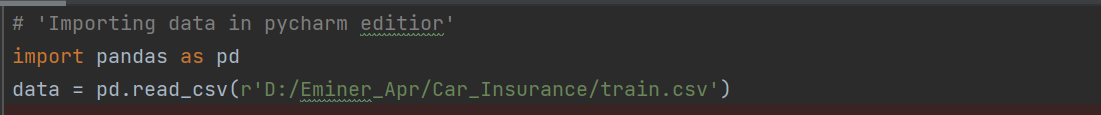
To perform data exploration and pre-processing, one can use various analytic tools such as Python, R, SAS, etc. In this dataset, the SAS can be used for data exploration and pre-processing.

**Missing data handling:**

The dataset contains some missing values that need to be handled. One approach to handle the missing values is to remove the rows that contain missing values. Another approach is to impute the missing values with appropriate methods such as mean, median, or mode imputation, or using more advanced techniques. In this dataset, the missing values can be imputed using the mean or median imputation.

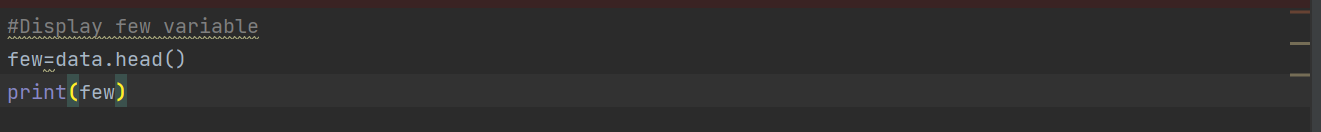
We are using PyCharm for cleansing of data.

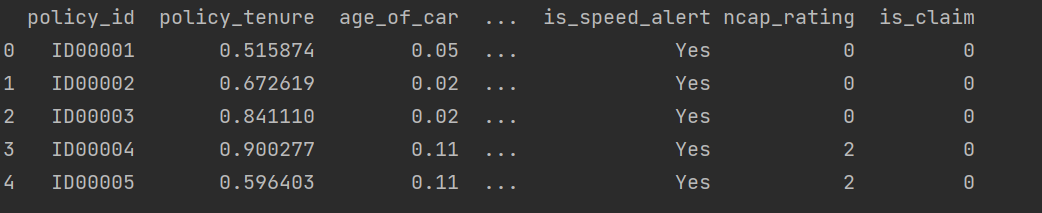
Step1: Getting the data into the editor:



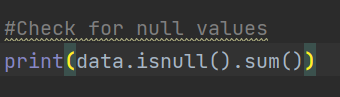
This will bring our data into PyCharm machine.

Step2: Checking for header





Step3: Now let’s check for null value if there is any

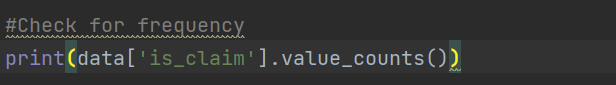


So, there is no null values in data so each and every record can be used in model.

**Inconsistent data handling:**

The dataset may also contain inconsistent data such as misspelled names or invalid entries, which can cause issues during analysis. It is necessary to clean such inconsistent data to avoid problems. In this dataset, we need to check for inconsistent data such as gender information.

Now let’s look into few of categorical variable and their distribution is there some data point which needs to be corrected or changed.





No discrepancy in is\_claim which is our dependent variable



For airbags also data looks good



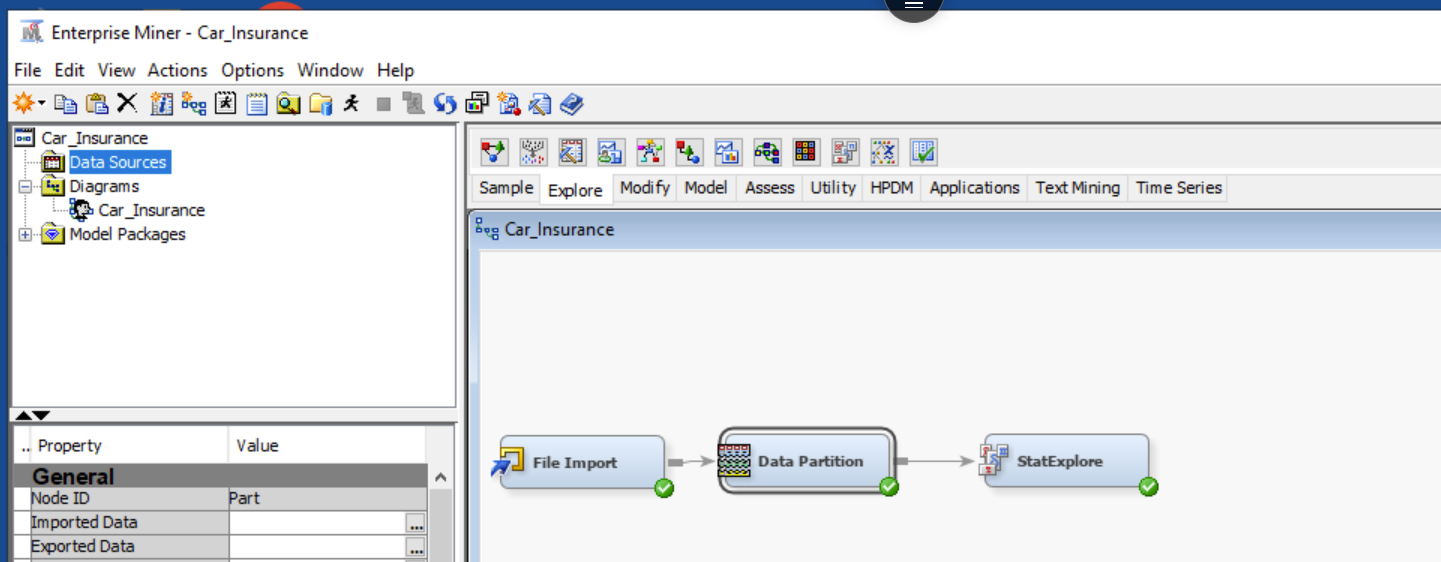
Similarly, we have checked for other categorical variables and everything looks into place.

# Predictive Modelling

Car insurance claims will be predicted using SAS E Miner where we will be using Logistic Regression and Decision Tree.

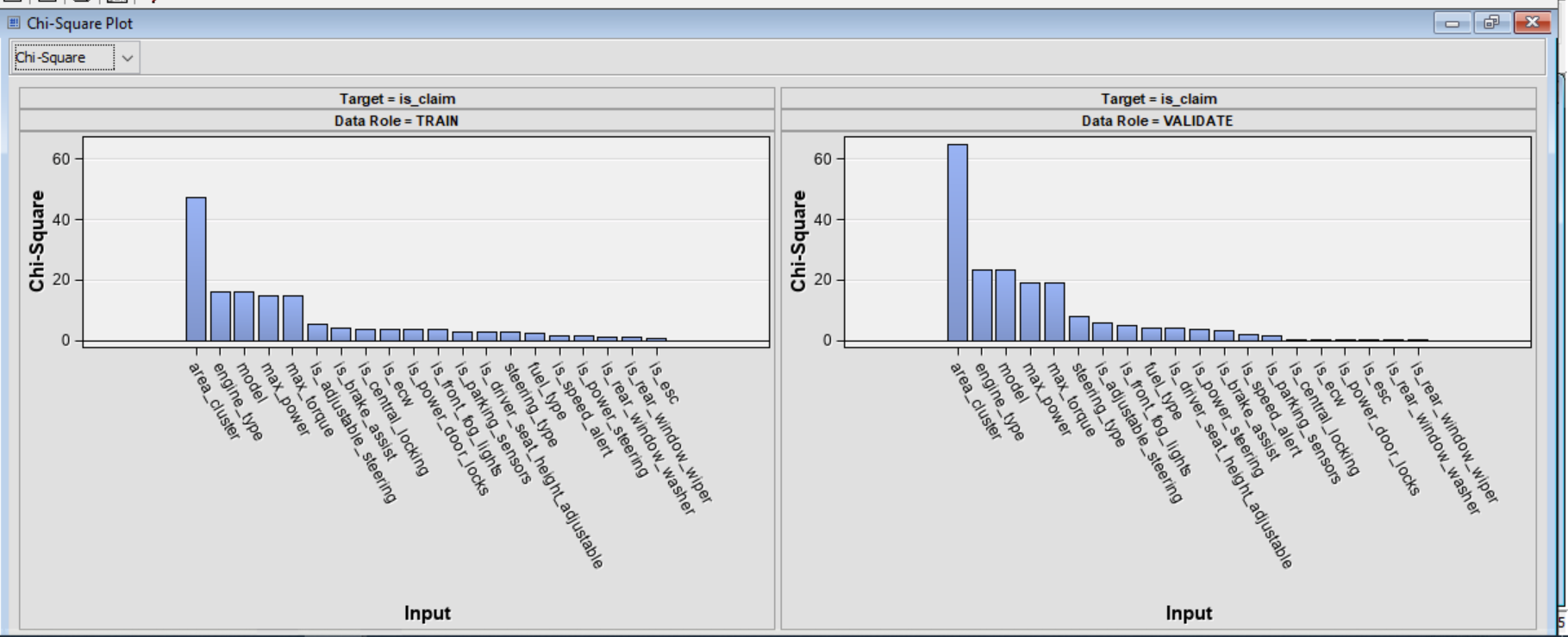
# Overview of models

The different models will be implemented using SAS E Miner. So, let’s get started. The first work is to get data in SAS miner and do starting exploration of data



The data is divided in 50:50 ratio for Training and Validation.

Stat Explorer:



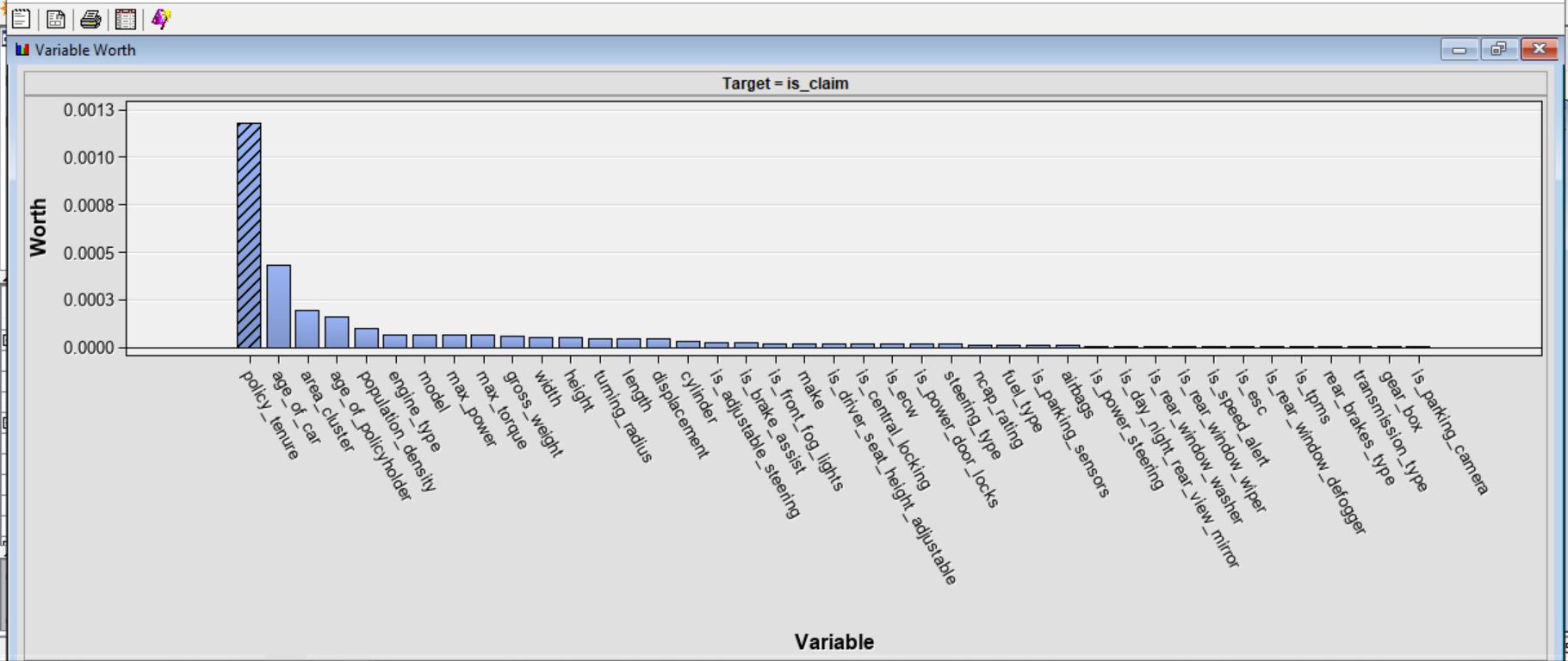
We can see for both training and validation the input variable priority is almost same leaving few of the variables like is\_esc, ls\_adjustable\_steering.

The above test is called Chi-Square test which is used to determine if there is any significant difference between expected and observed frequencies.

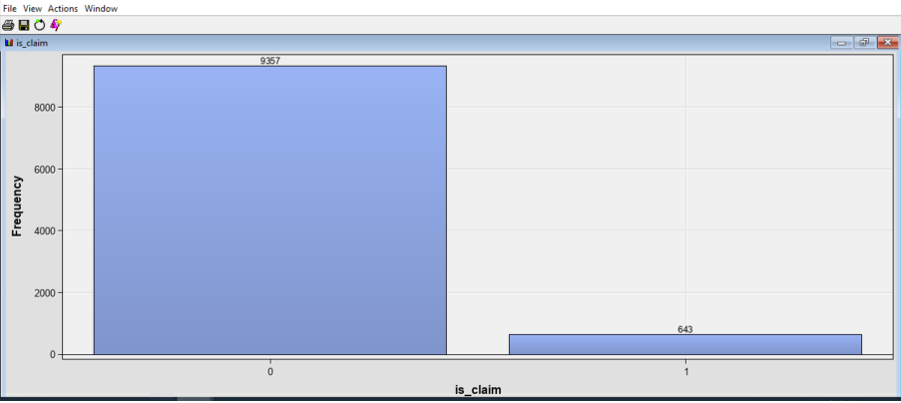
The important variables from the test are expected to be where there is difference:

* Area\_cluster
* Engine\_Type
* Model
* Max\_power
* Max\_Torque

Now the next screen shot show case variable worth in prediction:

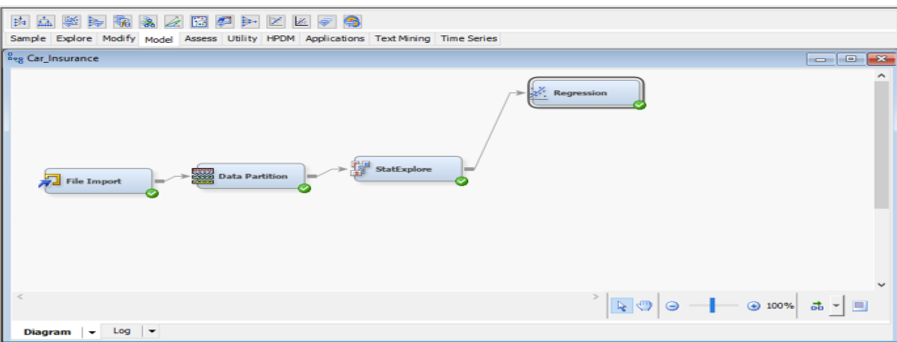


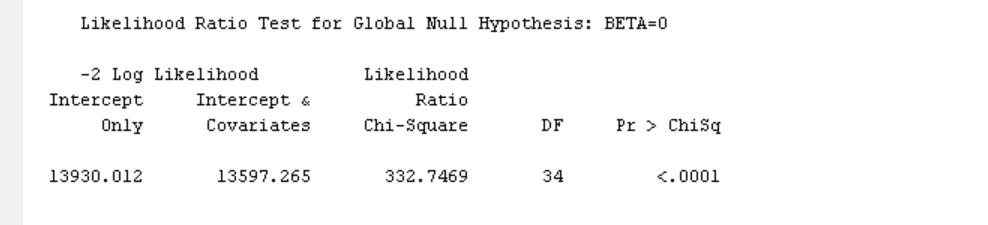
* Policy\_Tenure
* Age\_of\_car
* Area\_cluster
* Age\_of\_policyholder



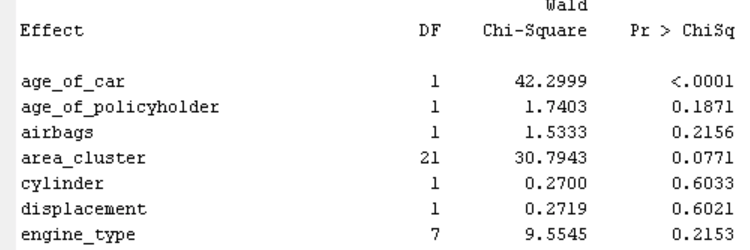
So, if we see the distribution of target variable the distribution is not in good shape. 94% data says zero and only around 6% says 1 which doesn’t look good for model building.

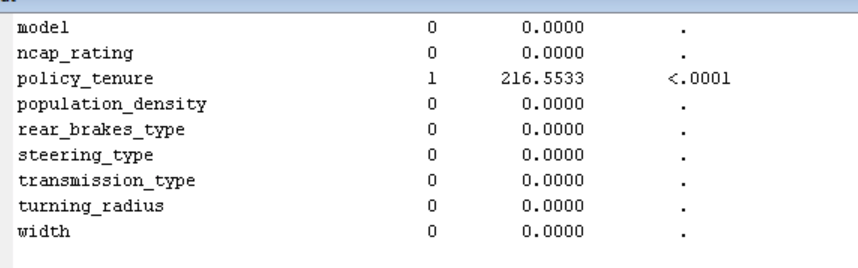
**Logistic Regression**

****

****

So, first thing is to look into the alpha value and we can see that P value is less than Alpha=0.05 that means the model looks good.

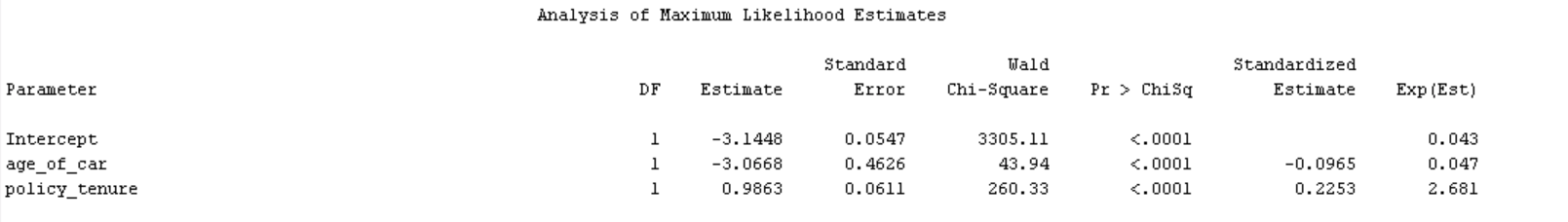




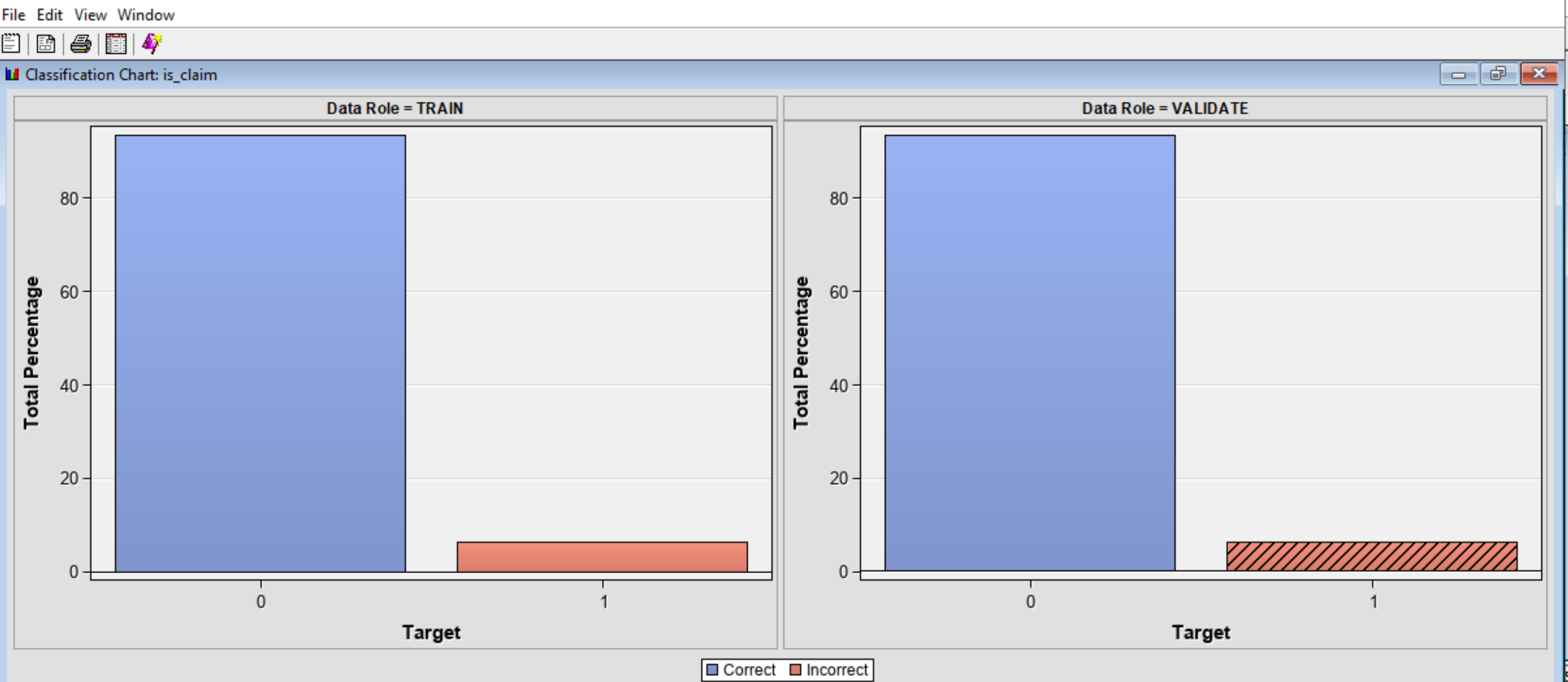
So, looks like in this model only two variables have an impact and other variable doesn’t have that much of impact so now let’s look into other model for regression.

**Forward Logistic Regression:**

So, first thing is what is forward logistic regression method, so model starts with no independent variables and iteratively adds one variable at a time, based on the significance level of each variable, until the model reaches the desired level of accuracy or until all independent variables have been added to the model



The result from the forward model is all most the same as the regression model so we will look into another one which is step wise regression model.

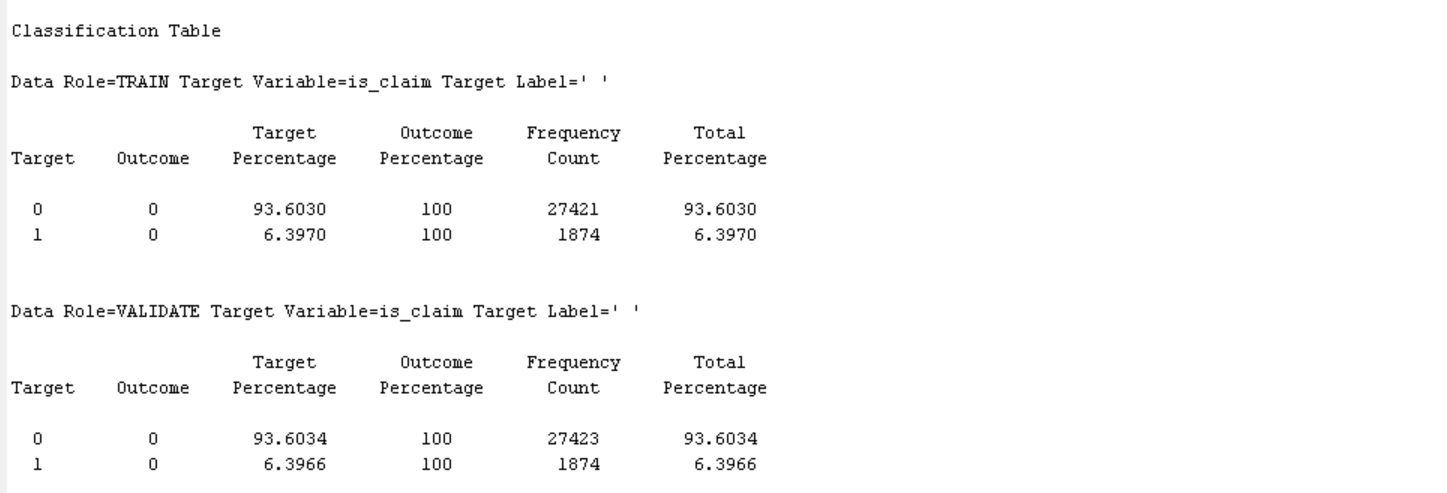


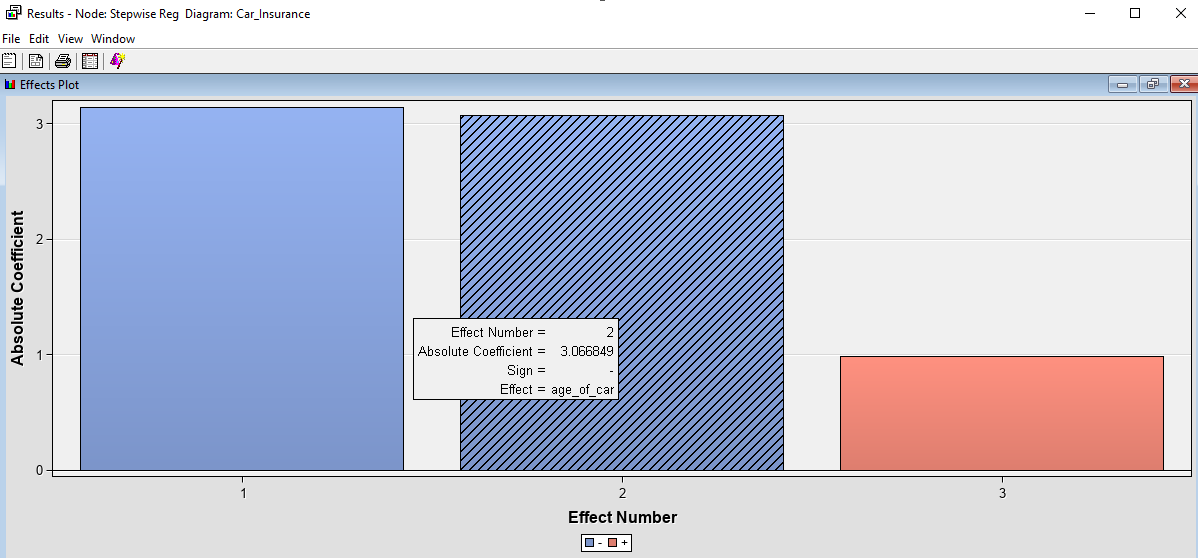
The graph highlights correct and incorrect prediction.

**Stepwise Logistic Regression:**

It’s combination of forward and backward approach.

For this model also the important variables remain the same so let’s look into train and validate data result how much of true prediction was done.





The stepwise shows the intercept for different variable here only two one is age\_of\_car and Policy\_tenure

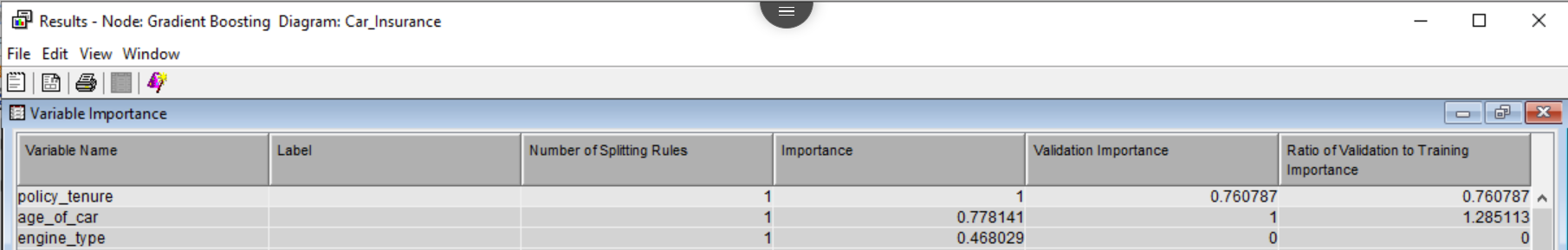
The one in blue is negative coefficient and one in red is positive coefficient.

**Gradient Boosting:**

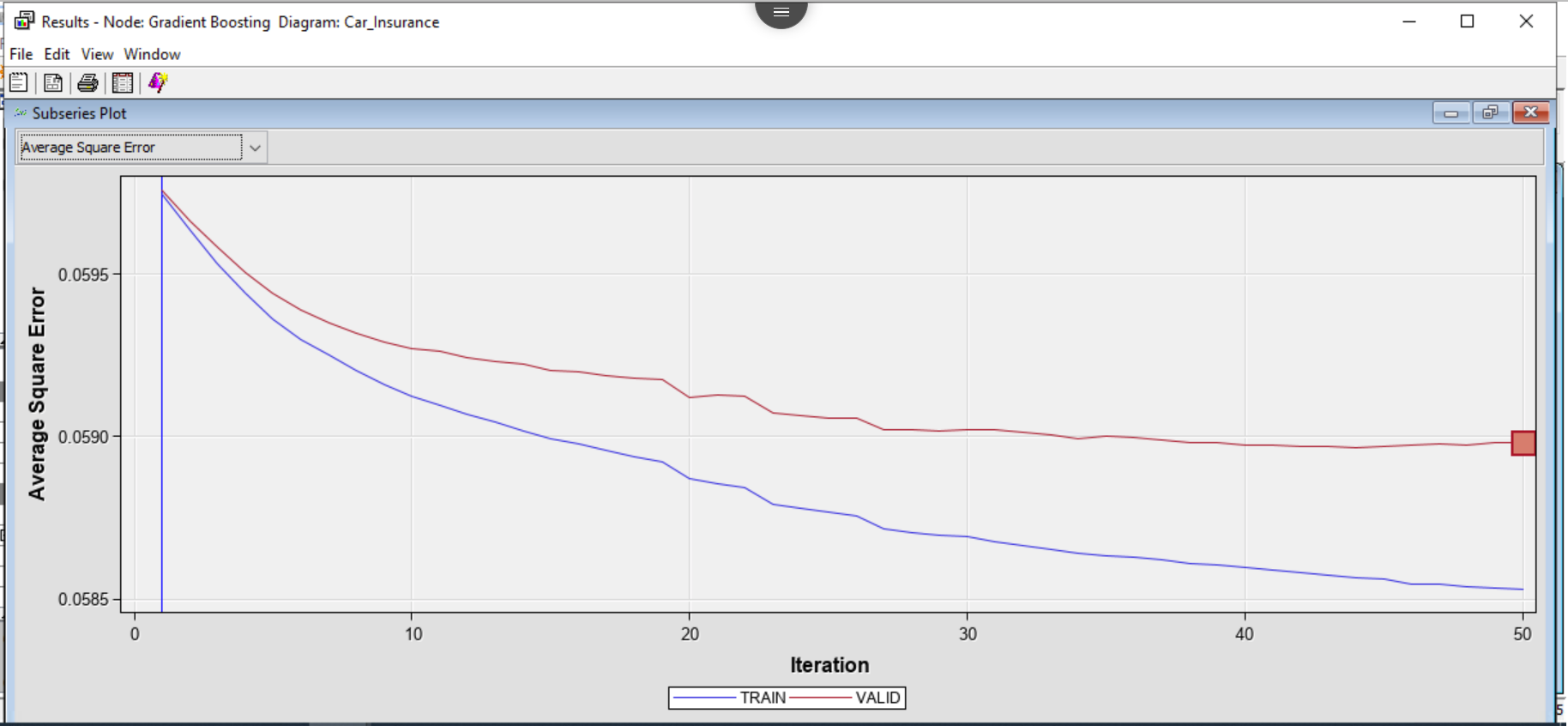
Gradient boosting is an algorithm used for both classification and regression problems. It works by combining multiple weak models into a strong model to make more accurate predictions.

In gradient boosting, model first starts with a single decision tree, which may not perform very well. The algorithm then evaluates the errors made by the decision tree and assigns weights to each instance based on how much error it contributed. A new decision tree is then trained to predict the errors made by the first tree, and this process continues iteratively until the final model is created.

So why we used Gradient Boosting as we see that it starts from single decision tree and by fixing the error it moves to multiple. So, while I was using decision tree it was not moving from the first node so because of that I had to use Gradient boosting.

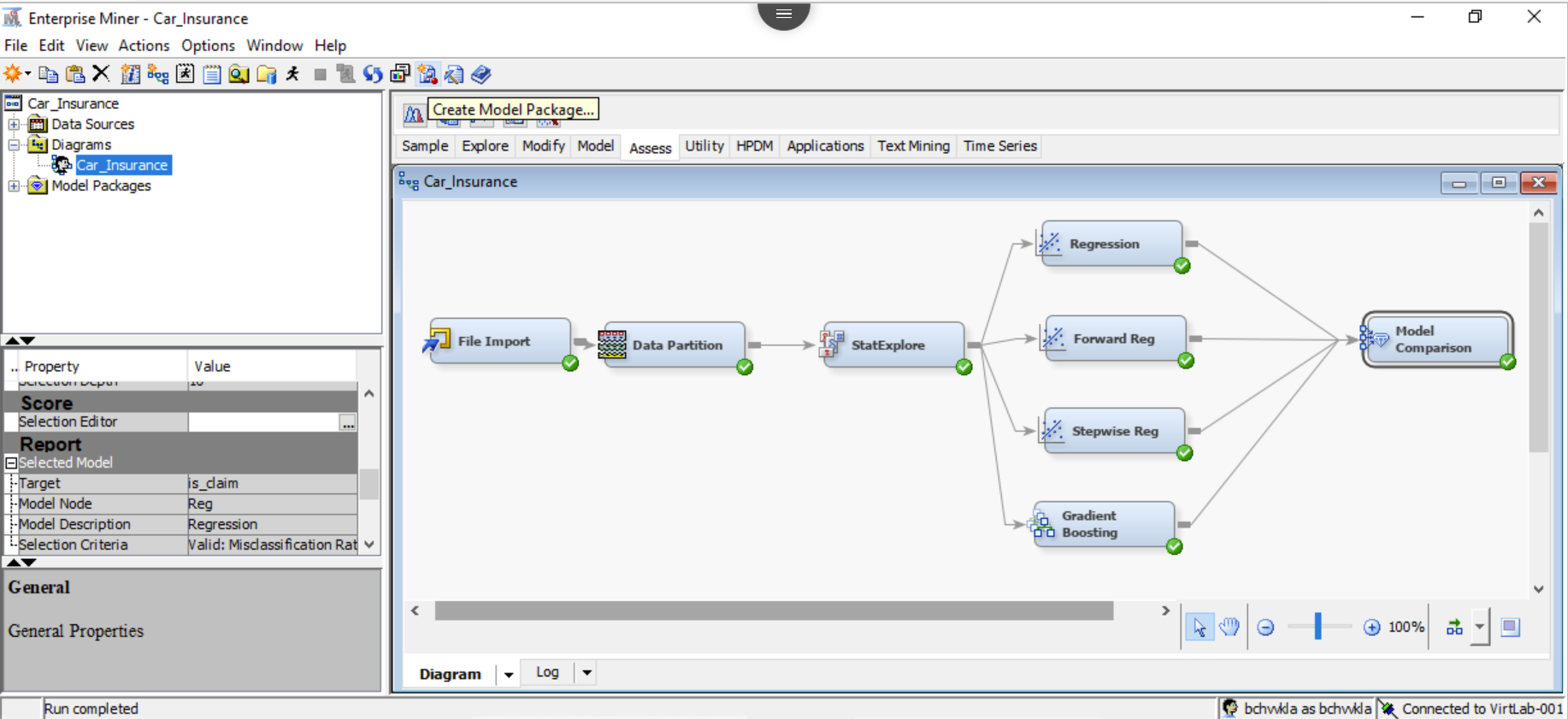


So, when we used Gradient boosting the number of important variables in model increased and it’s also using engine\_type.



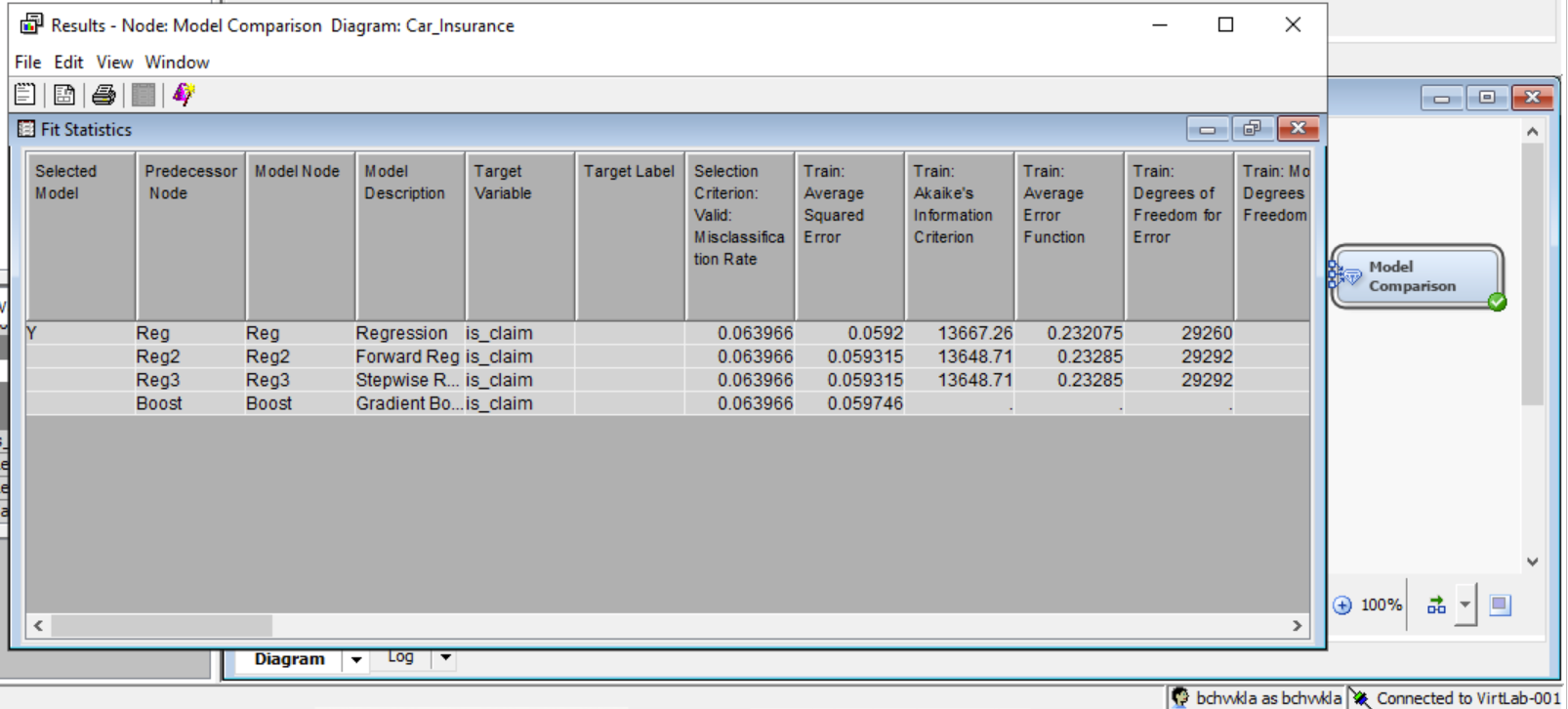
The average sq error is less for train data than validation and total number of iterations it has done is 50

So, the final model looks like something this:



As we mentioned earlier only that there is sync missing for the target variable as the data is not that accurate it will be really tough to build the model and that’s what we have seen the decision tree model didn’t give us any sort of result.

Let’s dive into Model comparisons:



To determine which model is best we look into misclassification rate and avg square error and we can see that for all our model it is pretty close.

But between all the four model Logistic regression is better and can be used for prediction purpose.

# 

# Interpretation of the findings and recommendations

### **Interpretation of the findings**

* While data exploration the important variables were: Policy\_Tenure, Age\_of\_car, Area\_cluster, Age\_of\_policyholder
* In model building for Regression model important variables were Age\_of\_car and Policy\_tenure
* While in case of gradient boosting there was an addition engine\_type
* As the target variable distribution was not up to mark most of model were not able to fit data
* Decision tree failed so we included Gradient boosting which improved the model
* A high misclassification rate is found due to imbalance data

### **Recommendations**

* The data needs to be corrected to get better result model wise
* Other than regression model we can use boosting technique this might increase the performance of model

# Summary and lesson Learned

This project aimed to develop a predictive model to accurately predict the likelihood of an insurance claim being filed based on customer profile information. Data analysis was conducted to identify the most important features that influence the likelihood of an insurance claim being filed. The predictive model was then developed using machine learning algorithms and evaluated for accuracy.

**Lesson Learned:**

This project highlighted the importance of understanding the data and the features that influence the outcome of a predictive model. Also, analysis on the tableau helped to learn the concept of data visualization that helps in the further processes. Furthermore, it demonstrated the need to evaluate different machine learning algorithms and techniques to develop the most accurate model. Lastly, it highlighted the need to analyze the data for potential patterns that could be used to further improve the accuracy of the model.

Few things from technical front:

* Different functionality and use of Tableau
* Use of SAS Miner
* Used Python, PyCharm IDE for data validation and cleansing

# Future Extension

As part of future extension for this project, need to learn more about data and try to come up with more highly distributed data so that we can create our model accordingly.

Try some more boosting technique or neural network if required to get better results and as part of extension to the project will be adding few more visualization to understand the data better.

# References

McGuire, G., Taylor, G. and Miller, H. (2018). Self-Assembling Insurance Claim Models Using Regularized Regression and Machine Learning. SSRN Electronic Journal. doi:10.2139/ssrn.3241906.

["Sensitivity vs Specificity"](https://www.technologynetworks.com/analysis/articles/sensitivity-vs-specificity-318222). Analysis & Separations from Technology Networks. Retrieved 10 December 2021.

R. Quinlan, "Learning efficient classification procedures", Machine Learning: an artificial intelligence approach, Michalski, Carbonell & Mitchell (eds.), Morgan Kaufmann, 1983, p. 463–482. [doi](https://en.wikipedia.org/wiki/Doi_(identifier)):10.1007/978-3-662-12405-5\_15