Table of Contents

1. Executive Summary
2. Objective
3. Dataset Description
4. Variables Description
5. File Import
6. Data Pre-processing
   1. Replacement
   2. Filter
   3. Data Partition
   4. Interactive Binning
   5. Impute
   6. Transform Variables
   7. Variable Selection
7. Data Mining Techniques
   1. Decision Tree
   2. Neural Network
   3. Logistic Regression
      1. None
      2. Forward Regression
      3. Backward Regression
      4. Stepwise regression
8. Model Comparison
9. Conclusion
10. References

1. **Executive Summary**

Auto dealerships purchase many of their used cars through auto auctions with the intent to buy as many cars as they can in the best condition possible. The biggest challenges that they face is the risk of buying used cars that might have unforeseen issues due to which they cannot be sold. These unfortunate purchases are termed as “kicks”. Kicked cars might have mechanical problems or some serious issues that prevent them from being sold, thus making a loss for dealerships. Dealerships can minimize their losses if they can predict whether the vehicle purchased at the auction is a kicked one or not. This will also help them to present the best inventory selection to the customers. The goal of this project is to design a prediction model that would help the dealerships when shopping for cars at auctions. This model would help them to minimize the risk of investing in the kicked cars, thus preventing any additional losses.

1. **Objective**

Our goal is to mine the dealerships’ data i.e., data of cars purchased at an action. Consequently provide the dealership with a model that will help them predict if the car purchased at the auction was a good buy or a bad buy (kick). We use SAS Enterprise miner as our mining tool.

1. **Dataset description**

The dataset that we utilized, for data mining, is a second hand data, obtained from an online source *kaggle.com*. This data stores details of cars purchased from 2009 to 2010, at various auctions, by the auto community. There are 34 attributes which gives us information about the specifications of the car, purchase details etc.

1. **Variables Description**

Following is the list of variables we have.

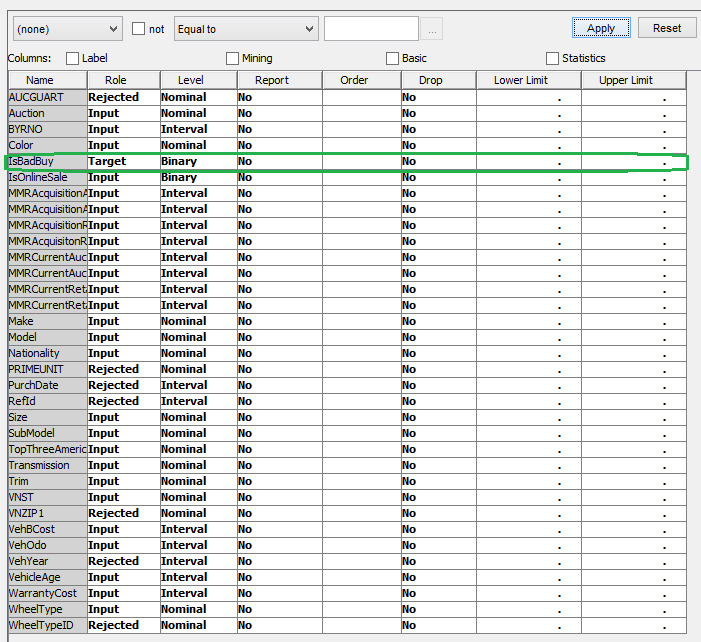
|  |  |
| --- | --- |
| Field Name | Definition |
| RefID | Unique (sequential) number assigned to vehicles |
| IsBadBuy | Identifies if the kicked vehicle was an avoidable purchase |
| PurchDate | The Date the vehicle was Purchased at Auction |
| Auction | Auction provider at which the vehicle was purchased |
| VehYear | The manufacturer's year of the vehicle |
| VehicleAge | The Years elapsed since the manufacturer's year |
| Make | Vehicle Manufacturer |
| Model | Vehicle Model |
| Trim | Vehicle Trim Level |
| SubModel | Vehicle Submodel |
| Color | Vehicle Color |
| Transmission | Vehicles transmission type (Automatic, Manual) |
| WheelTypeID | The type id of the vehicle wheel |
| WheelType | The vehicle wheel type description (Alloy, Covers) |
| VehOdo | The vehicles odometer reading |
| Nationality | The Manufacturer's country |
| Size | The size category of the vehicle (Compact, SUV, etc.) |
| TopThreeAmericanName | Identifies if the manufacturer is one of the top three American manufacturers |
| MMRAcquisitionAuctionAveragePrice | Acquisition price for this vehicle in average condition at time of purchase |
| MMRAcquisitionAuctionCleanPrice | Acquisition price for this vehicle in the above Average condition at time of purchase |
| MMRAcquisitionRetailAveragePrice | Acquisition price for this vehicle in the retail market in average condition at time of purchase |
| MMRAcquisitonRetailCleanPrice | Acquisition price for this vehicle in the retail market in above average condition at time of purchase |
| MMRCurrentAuctionAveragePrice | Acquisition price for this vehicle in average condition as of current day |
| MMRCurrentAuctionCleanPrice | Acquisition price for this vehicle in the above condition as of current day |
| MMRCurrentRetailAveragePrice | Acquisition price for this vehicle in the retail market in average condition as of current day |
| MMRCurrentRetailCleanPrice | Acquisition price for this vehicle in the retail market in above average condition at time of purchase as of current date |
| PRIMEUNIT | Identifies if the vehicle would have a higher demand than a standard purchase |
| AUCGUART | The level guarntee provided by auction for the vehicle (Green light - Guaranteed /arbitratable, Yellow Light - caution/issue, red light - sold as is) |
| BYRNO | Unique number assigned to the buyer that purchased the vehicle |
| VNZIP | Zipcode where the car was purchased |
| VNST | State where the the car was purchased |
| VehBCost | Acquisition cost paid for the vehicle at time of purchase |
| IsOnlineSale | Identifies if the vehicle was originally purchased online |
| WarrantyCost | Warranty price (term=36month and millage=36K) |

**Mining using SAS**

1. **File Import**

We use file import node to import our data which is a .csv file. In this node we have assign ‘IsBadBuy’ as the target variable and reject the irrelevant variables for our analysis.

IsBadBuy is a binary variable with values either 0 or 1. When the value is 1, it states that the car purchased at the auction is a kick.



We explicitly reject the following few variables as they do not have any importance in determining whether the car purchased is a kick or not

Redundant Data: The data set contains the following redundant variables.

“VehYear” and “VehicleAge” are two variables with almost the same meaning. We mark “VehYear” as rejected.

“VNZIP” and “VNST” also signify the same thing. We set “VNZIP” as rejected

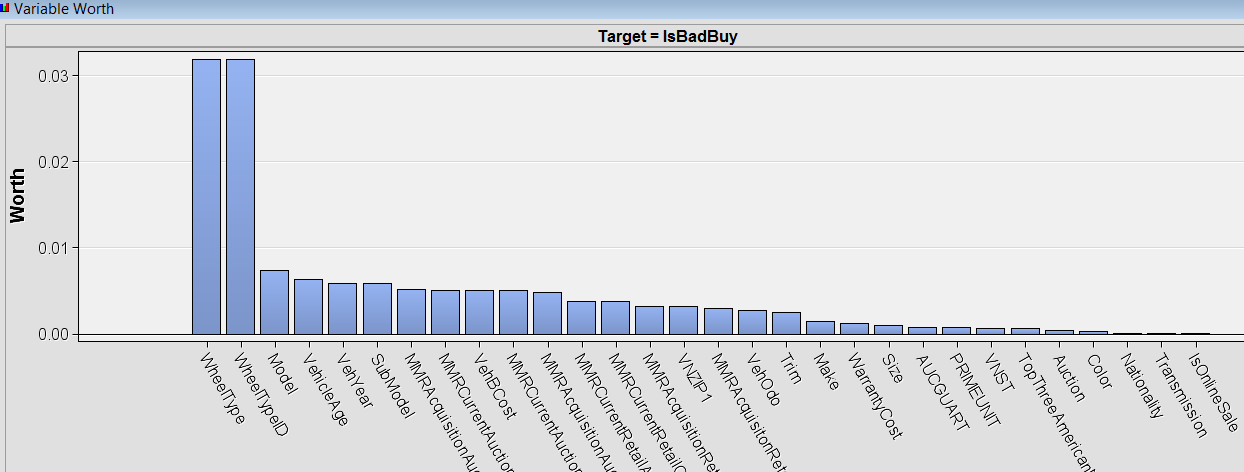
“WheelType” and “WheelTypeId” represent the same thing. So we reject ‘WheelTypeId’

Insignificant variables: Following variables do not help us in our prediction model

RefId -This is the unique number assigned to every vehicle purchased at the auction.

PurchDate - The Date the vehicle was purchased at Auction. It has dates from 2009 to 2010. Also, more than 50% of the records do not have value for this variable

We run StatExplore node on our data to examine variable distributions and it generates summarization statistics. It gives us a summary of the number of classes for each of the nominal variables, a count of the missing values for each attribute. It also shows the importance of each variable once it is imported in SAS. The results are as below



Results show that the following variables have missing values

Trim : Vehicle Trim Level

Transmission : Vehicles transmission type (Automatic, Manual)

MMRAcquisitionAuctionAveragePric: Acquisition price for this vehicle in average condition at time of purchase

MMRAcquisitionAuctionCleanPrice: Acquisition price for this vehicle in the above Average condition at time of purchase

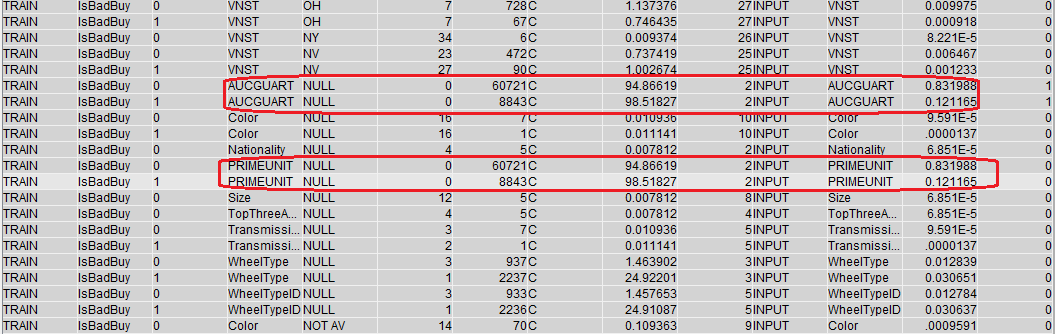
MMRAcquisitionRetailAveragePrice: Acquisition price for this vehicle in the retail market in average condition at time of purchase

MMRAcquisitonRetailCleanPrice: Acquisition price for this vehicle in the retail market in above average condition at time of purchase

From the below figure we see that we can reject following mentioned variables because most of the values are NULL (close to 95%)

AUCGUART - The level guarantee provided by auction for the vehicle

PRIMEUNIT - Identifies if the vehicle would have a higher demand than a standard purchase

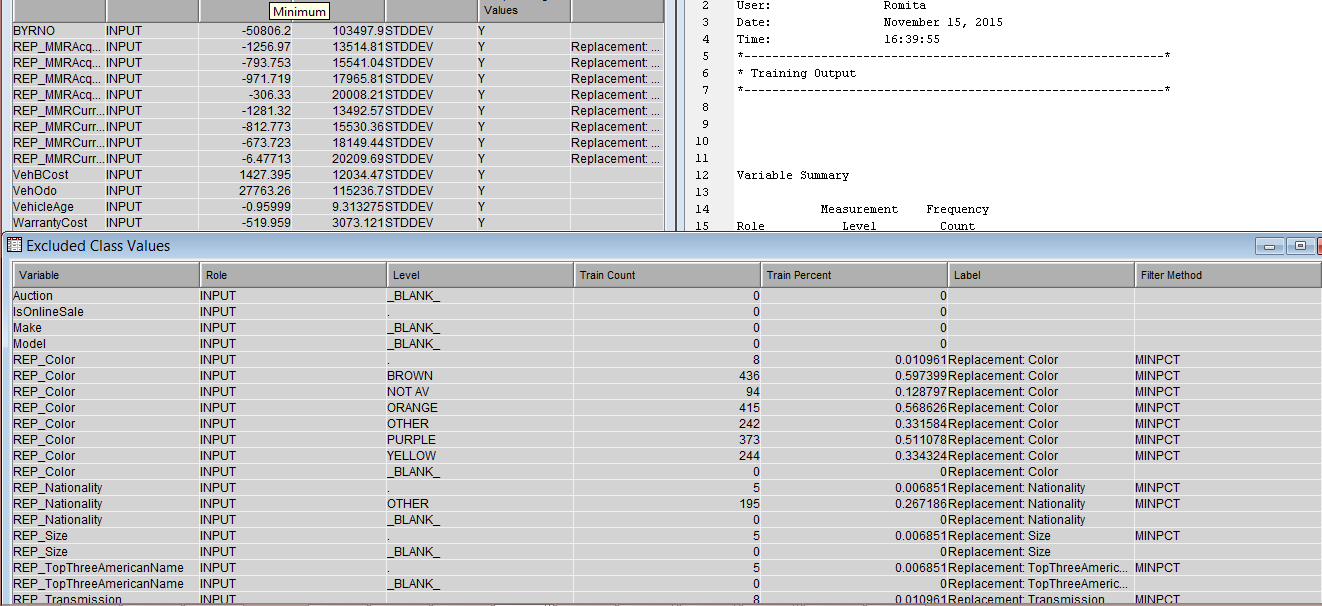


1. **Data Pre-Processing**
   1. **Replacement**   
      We ran the Replacement node on our data to replace all our missing/null values with ‘.’. Variables storing vehicles price values have a few values which are 0. As none of the prices can have a value of 0$, we consider 0 as missing value for such variables. We take 1 as the lower threshold limit for all the price variables and replace them, as well, with ‘.’. Following are the variables storing price values

|  |  |
| --- | --- |
| Variable | Description |
| MMRAcquisitionAuctionAveragePrice | Acquisition price for this vehicle in average condition at time of purchase |
| MMRAcquisitionAuctionCleanPrice | Acquisition price for this vehicle in the above Average condition at time of purchase |
| MMRAcquisitionRetailAveragePrice | Acquisition price for this vehicle in the retail market in average condition at time of purchase |
| MMRAcquisitonRetailCleanPrice | Acquisition price for this vehicle in the retail market in above average condition at time of purchase |
| MMRCurrentAuctionAveragePrice | Acquisition price for this vehicle in average condition as of current day |
| MMRCurrentAuctionCleanPrice | Acquisition price for this vehicle in the above condition as of current day |
| MMRCurrentRetailAveragePrice | Acquisition price for this vehicle in the retail market in average condition as of current day |
| MMRCurrentRetailCleanPrice | Acquisition price for this vehicle in the retail market in above average condition as of current day |

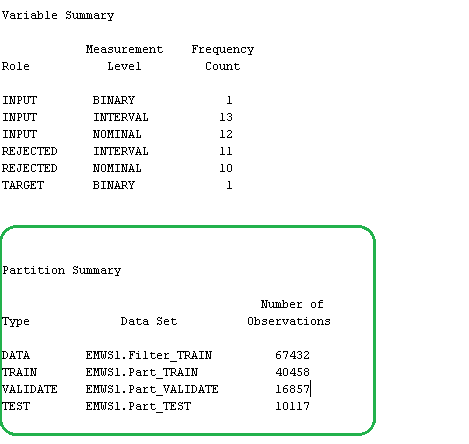
* 1. **Filter**

The Filter node helps us exclude outliers or other observations that is not significant in our data mining analysis. Default filtering method was kept as none. Following screenshot is a result of the filtering applied on the data.



* 1. **Data Partition**

After a few pre-processing steps we partitioned the data using data partitioning node. The data was divided into Training, Validation & Test data in the ratio 60:25:15 respectively. Following is the result after partitioning



* 1. **Interactive Binning**

We applied this technique to bin the variables which had a high class level count. We chose attributes ‘Model’, ‘Sub Model’, ‘Trim’, ‘Color’ and ‘Transmission’ for interactive binning. Model and SubModel had initial class levels count of more than 500. After the applying this method we were able to reduce the class count to 51 and 40 for model and submodel respectively.

* 1. **Impute**

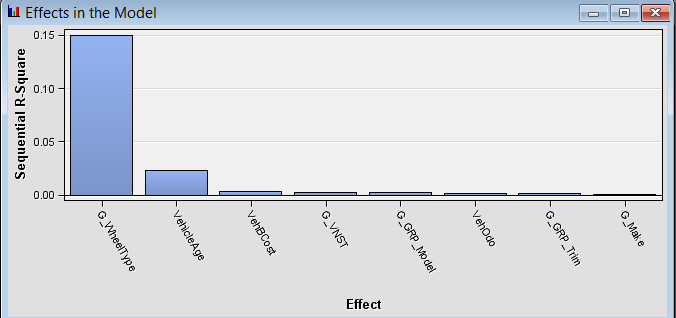
This node was used to replace the missing values. Missing values of interval data was replaced with the m*ean* and those for Nominal/Binary data was replaced with *the most frequently occurring* value.

* 1. **Transform Variables**

We applied transform as a part of pre-processing to reduce the skewness of variables. Logarithmic transformation was applied on the attribute ‘WarrantyCost’. We tried to create a few more formulae such as creating binary variables for VehicleAge and WheelType. But that ultimately resulted in an increase in the Misclassification Rate. Hence we deleted the newly created variables.

* 1. **Variable Selection**

This node helps us in reducing the variables needed for analysis. Variables selected for analysis in the Variable Selection node are available to the subsequent nodes. This node was applied after Transform variables node. Following chart shows the effect of the variables with corresponding R-square values.

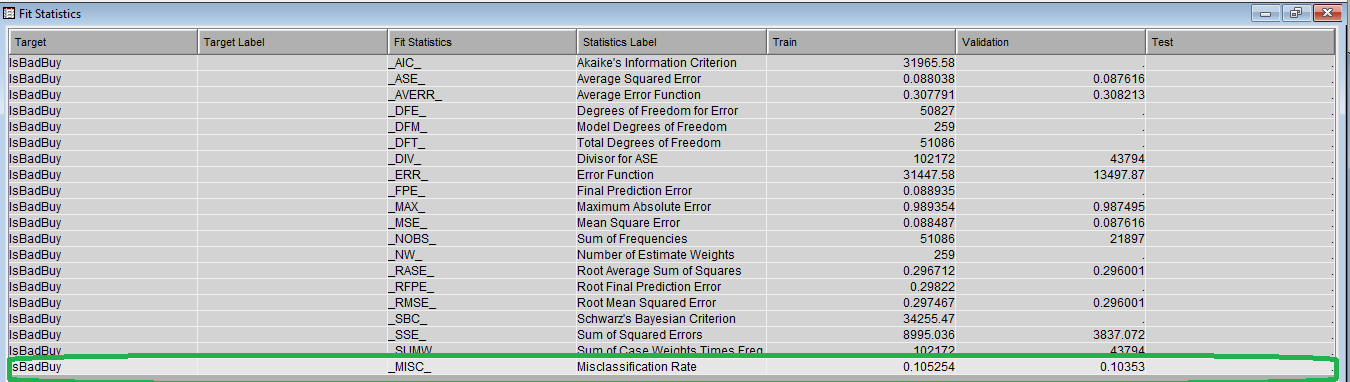


1. **Data Mining Techniques**

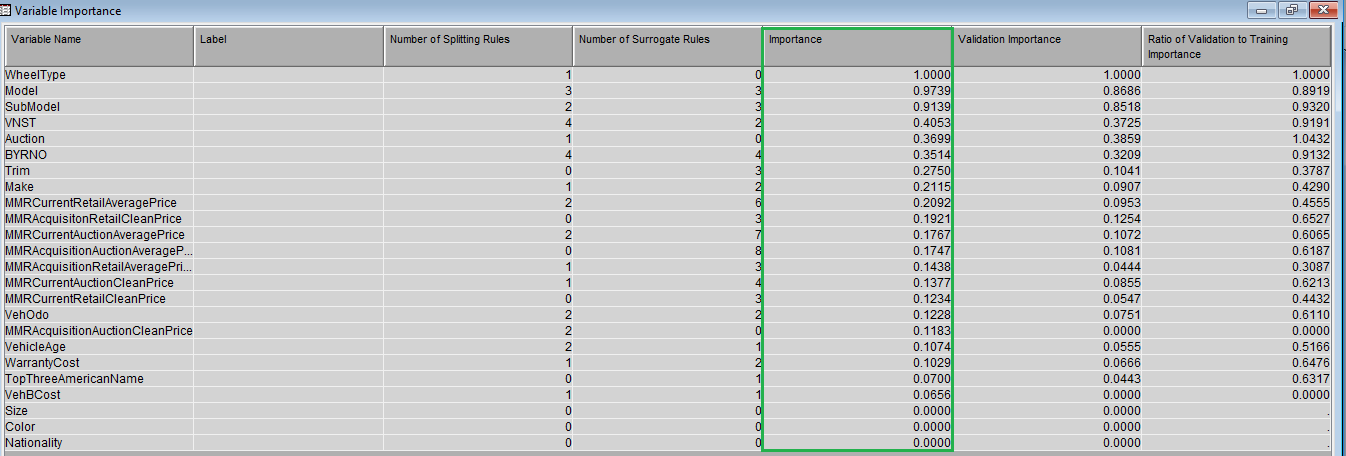
We explored below mentioned data mining techniques and based upon the findings of the various models we chose the best one

* 1. **Decision Tree**

Decision Tree helps to classify data based on certain significant variables. A decision tree is a decision support tool that uses a tree-like graph or model of decisions and their possible consequences, including chance event outcomes, resource costs, and utility. The findings of the decision tree, on our dataset, are as follow

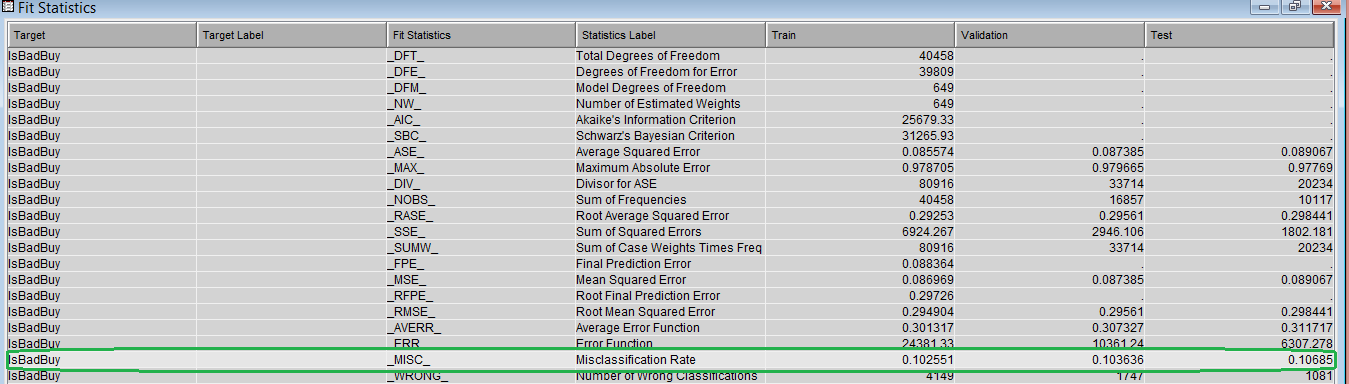
Here the miscalculation rate is 10.35% which is comparatively higher than the regression models. Hence we do not use this model for our analysis.

Following table shows us the importance of variables used for decision tree

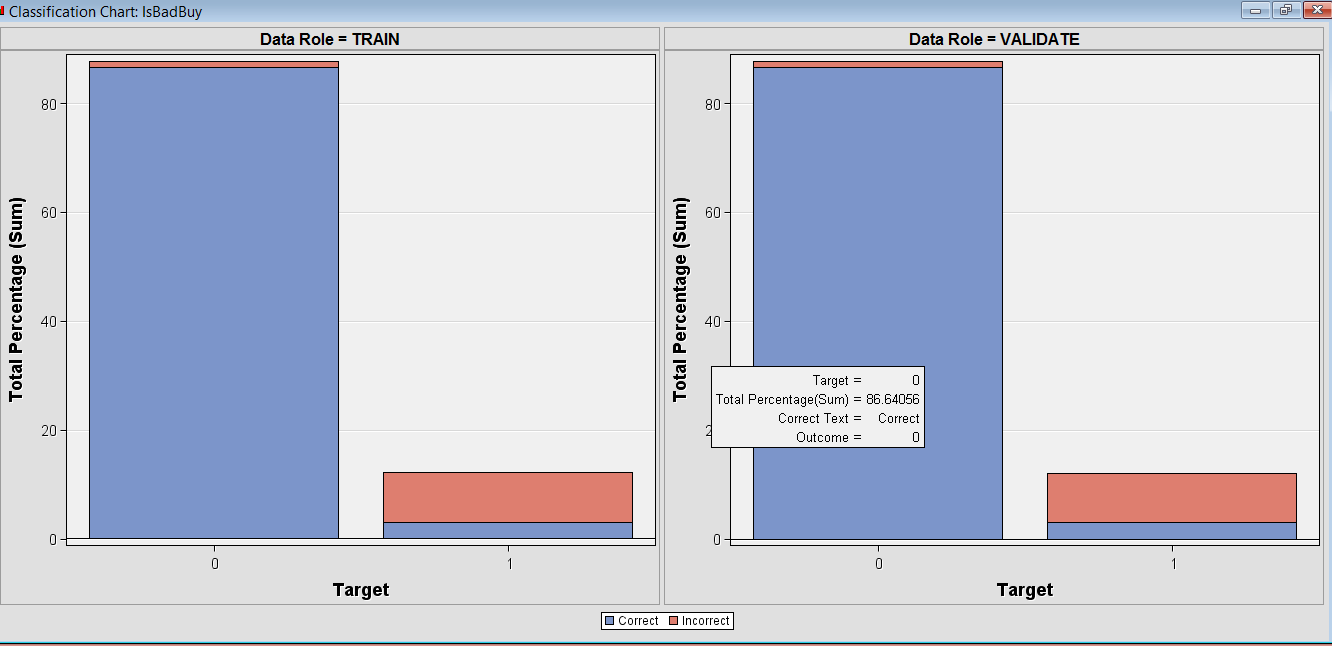


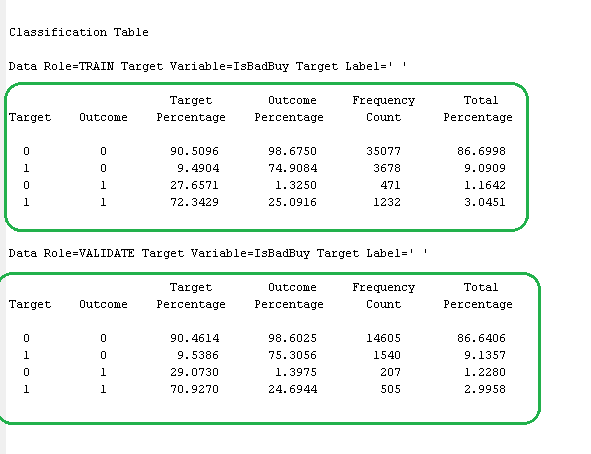
* 1. **Neural Network**

Following shown is the result after using Neural Network node as the model



Here the Misclassification rate **is 10.3638%.**

In test result, we take a closer look to misclassification rate in the Classification Chart as shown below

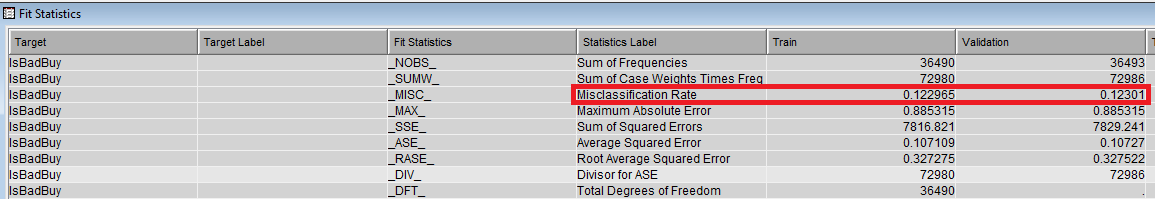


We investigate the above classification table in Output. The 10.3638% misclassification rate is consist of 9.14% false positive, and 1.23% false negative. This means 9.14% of the predicted bad buys in the modal are in fact good buys; 1.23% of the predicted good buys predicted in the modal are bad buys.

* 1. **Gradient Boosting**

This Model creates ‘the series of Decision Trees’ by fitting the residual of the prediction from the earlier tree in the series.

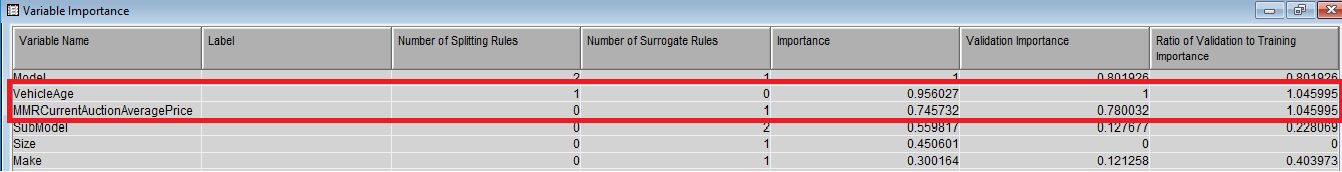
Validation Misclassification Rate = 12.3%



According to the Gradient Boosting, the most important variables that contribute in predicting the dependent variable (IsBadBuy) are –

1 – VehicleAge

2 – MMRCurrentAuctionAveragePrice



* 1. **Logistic Regression**

We performed regression analysis on our dataset to find the significance of the attributes which drive the target variable, IsBadBuy

As our target variable is a binary variable we performed logistic regression on our dataset. Following are the results for logistic regression, with different selection models.

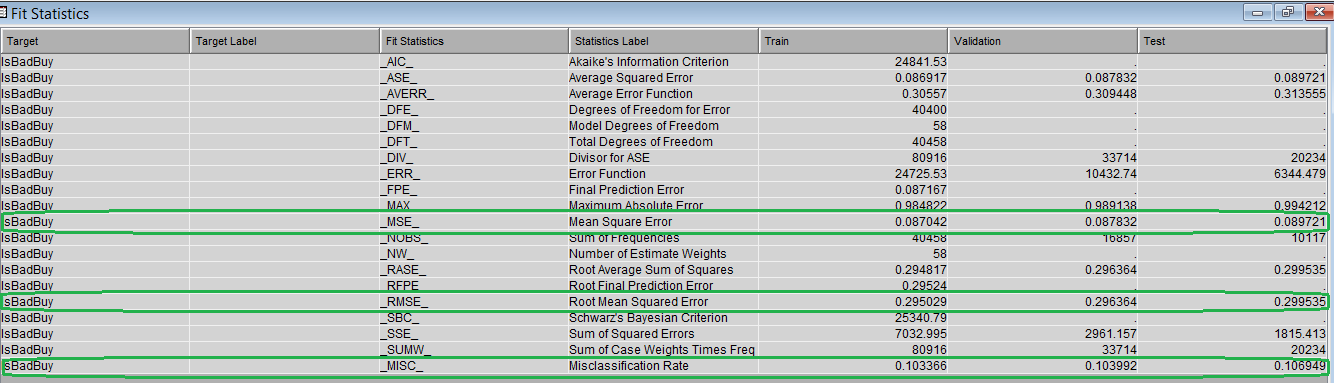
* + 1. **None**

We tried this selection model at various steps in our process as mentioned below

First we tried it directly after interactive binning from which we got the misclassification rate as 10.3399%

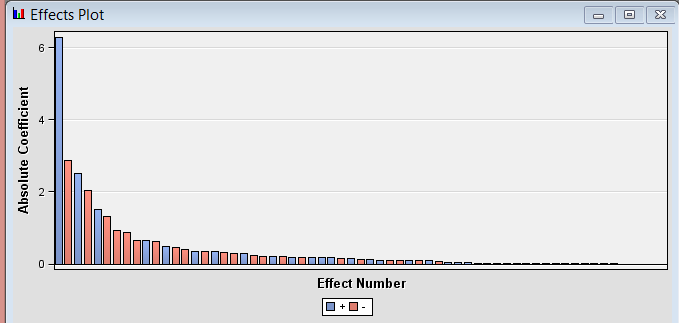
Then we performed logistic regression after transforming variables, for skewness. Here the misclassification rate came out to be 10.33992% which was same as that for the first one.

**Fit Statistics**



The above statistics tell us about the accuracy of the model. Misclassification Rate is the fraction of cases assigned to the wrong class, which in our case is close to 10.39% for the validation dataset.

**Effects Plot**



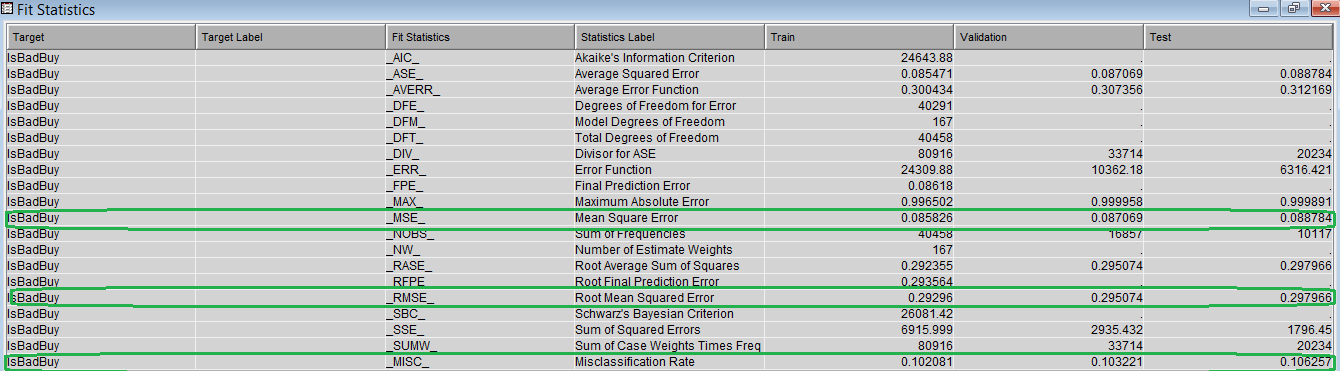
From the effects plot window we can see greater the absolute value of variable, the more important that variable is to the regression model.



The above results talk about the importance of the variables in the regression model.

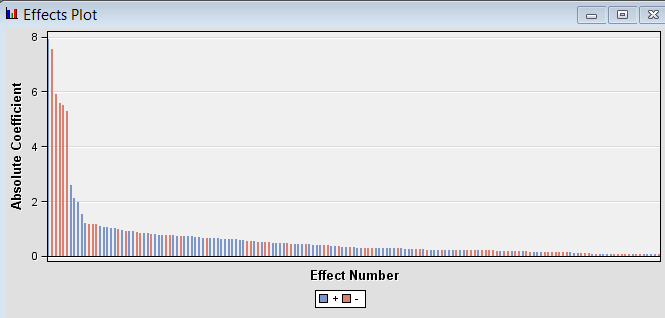
* + 1. **Forward Selection**

**Fit Statistics**



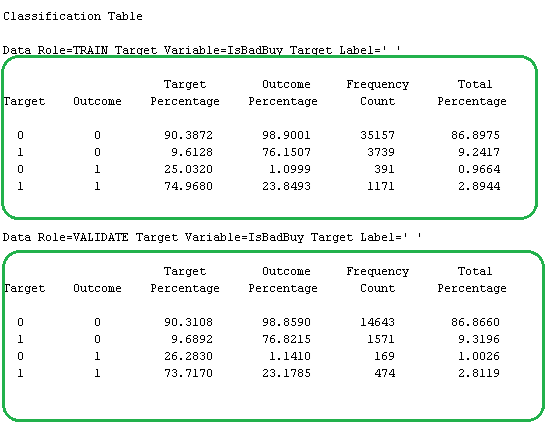
From the above results we can see that the Misclassification Rate is **10.322%** for the validation dataset, which seems to be a good result.

**Effects Plot**

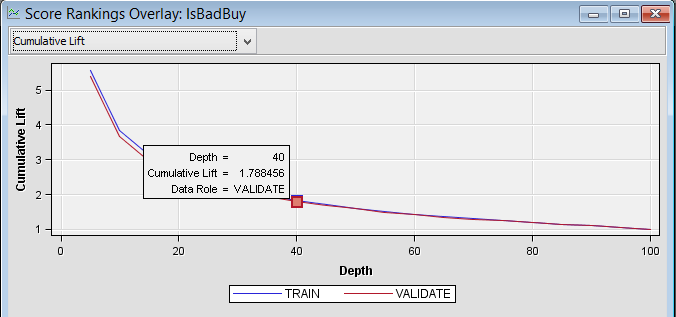


From the effects plot window we can see greater the absolute value of variable, the more important that variable is to the regression model.

The results of forward regression show that the analysis starts with the attribute ‘WheelType’, then ‘VehicleAge and so on. It goes on adding variables till the point where it sees that addition of more variables do not make any further significant improvement to the model.



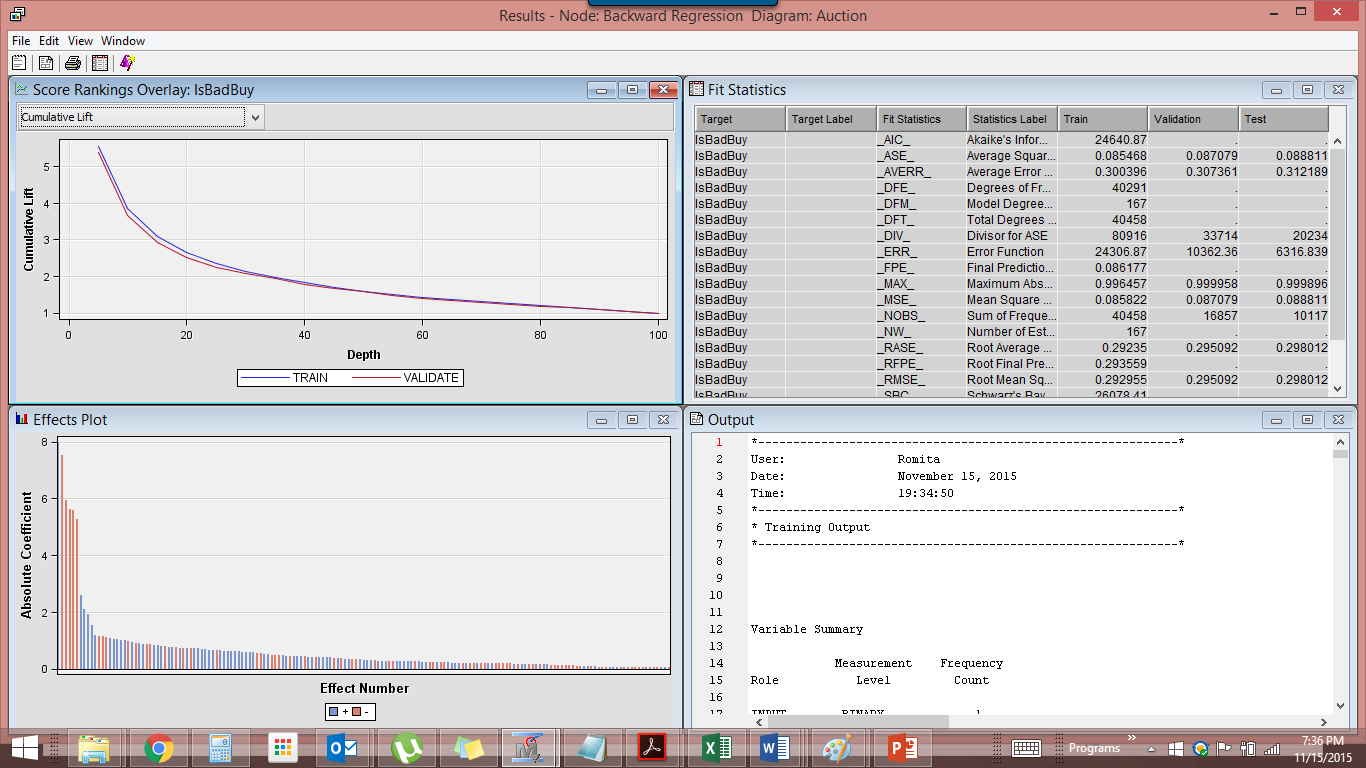
The above classification table shows that the misclassification rate of 10.32% in the forward regression model is consist of 9.32% false positive and 1% false negative. The result is very similar to that of backward regression. This means 9.32% of the predicted bad buys are in reality good buys, and 1% of the predicted good buys are in reality bad buys.

****

Above shown is the Cumulative Lift chart. As expected, for a good model, the cumulative lift is exponentially decreasing.

* + 1. **Backward Regression**

Backward regression starts with all available transformed input variable, and drop the variable that has the least impact on the model fit in each step. We run Regression node in backward setting, and results are as follow.



24 variables were used in the final backward regression modal; 10 are nominal variables, and 13 are interval variable, and 1 is binary variable. The most influential variable is wheel type, which is consistent with Decision Tree and other regression models.

* + 1. **Stepwise Regression**

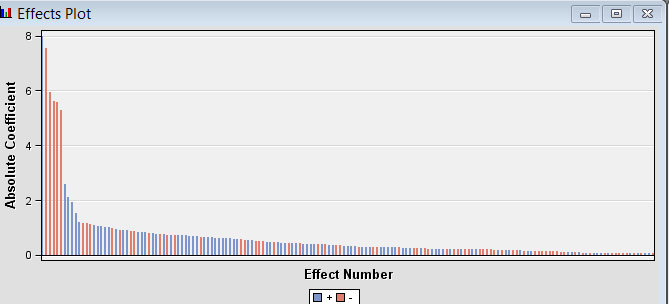
Following are the results obtained after performing Stepwise Regression

**Fit Statistics**



The misclassification rate comes out to be 10.3221%

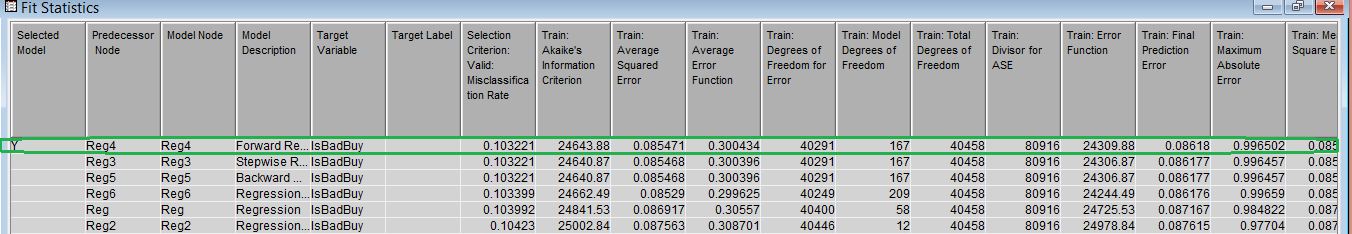
**Effects Plot**

****

Stepwise regression shows that the analysis starts with the attribute ‘WheelType’

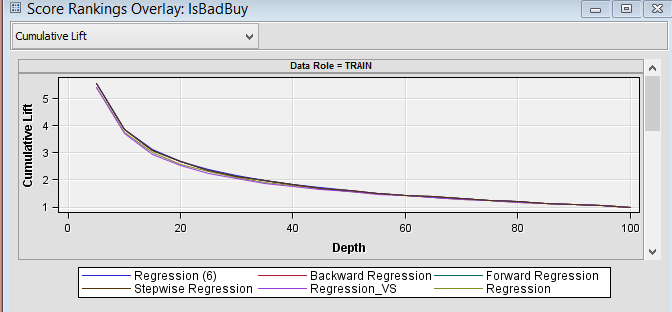
1. **Model Comparison**

We run model comparison on all the regression models we used. Following is the comparison result obtained.

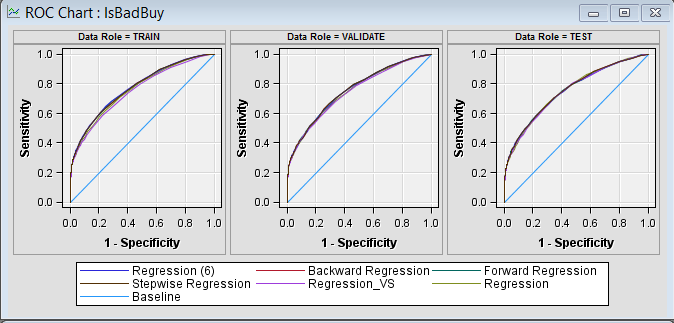


SAS selects forward logistic regression as the best model with a misclassification rate of **10.3221%.**

Below is the cumulative lift chart for all the regression models



Following is the ROC chart for our models with Forward Regression as the closest one which shows the best accuracy.



1. **Conclusion**

Forward Logistic regression was the best model chosen for the prediction of kicked cars. This was because it had the least misclassification rate and a good r-square value. This model also showed slight higher accuracy than the other models we tried.