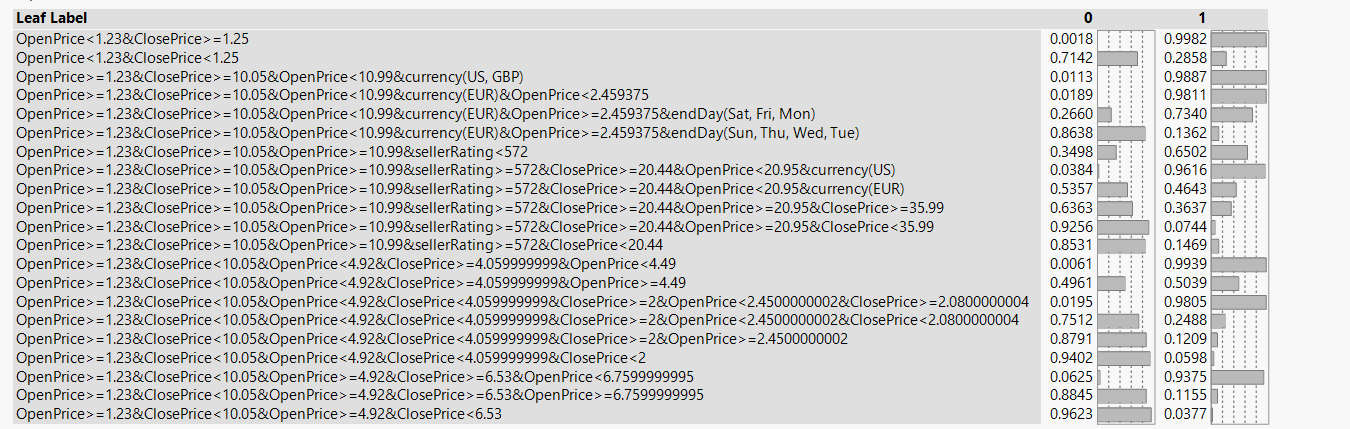
**OPIM 5604 B15 – Predictive Modeling Assignment Meghana Kasula (Net ID=mek15120)**

*“The work contained and presented here is my work and my work alone.”*

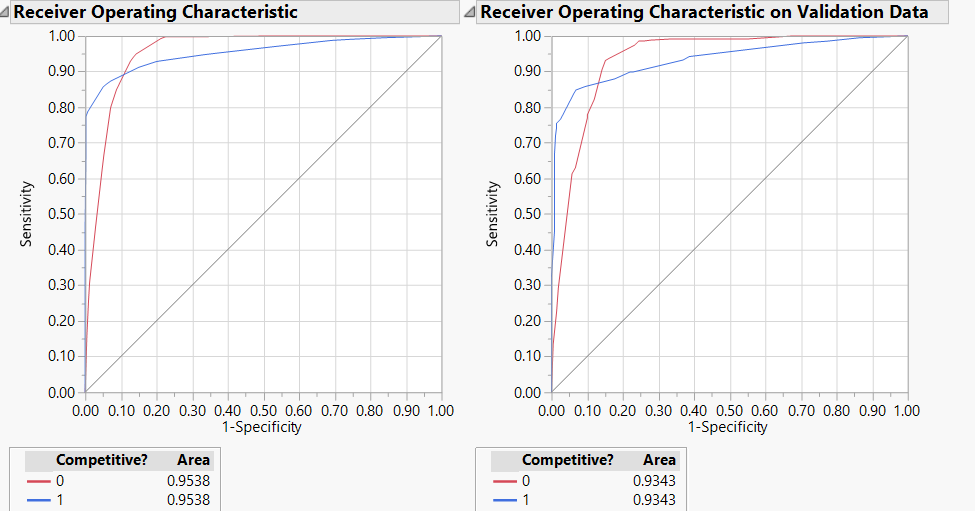
**9.1 Competitive Auctions on eBay.com.**

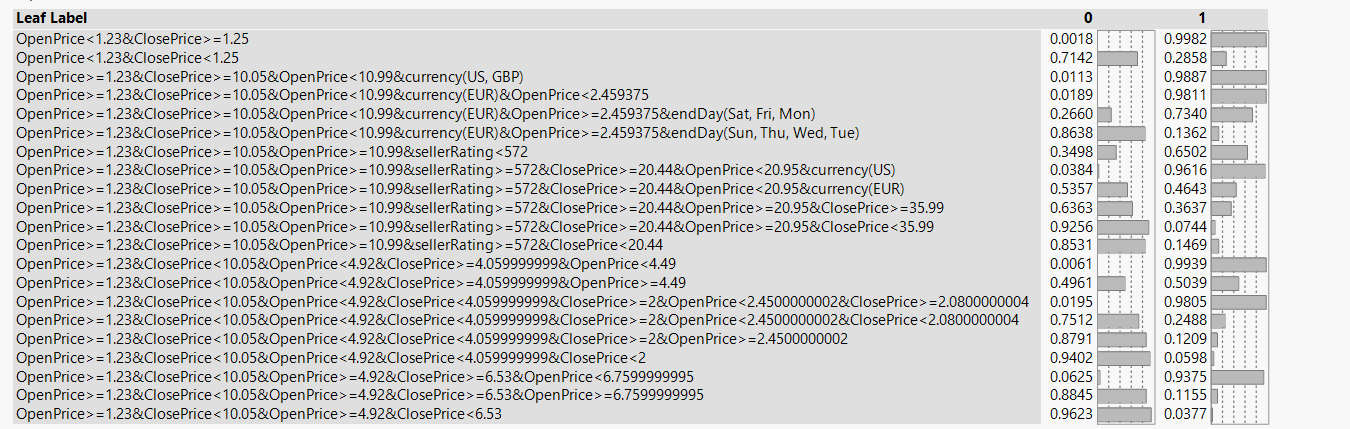
**Data preprocessing. Split the data into training and validation datasets using a 60% : 40 % ratio.**

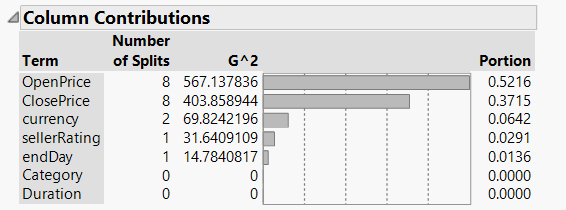
**a. Fit a classification tree using all predictors, using the Go button. Display the leaf report, and write down the first four branches in the leaf report in terms of rules.**

The four first branches in the leaf report are covered in the red box.

**b. Is this model practical for predicting the outcome of a new auction?**

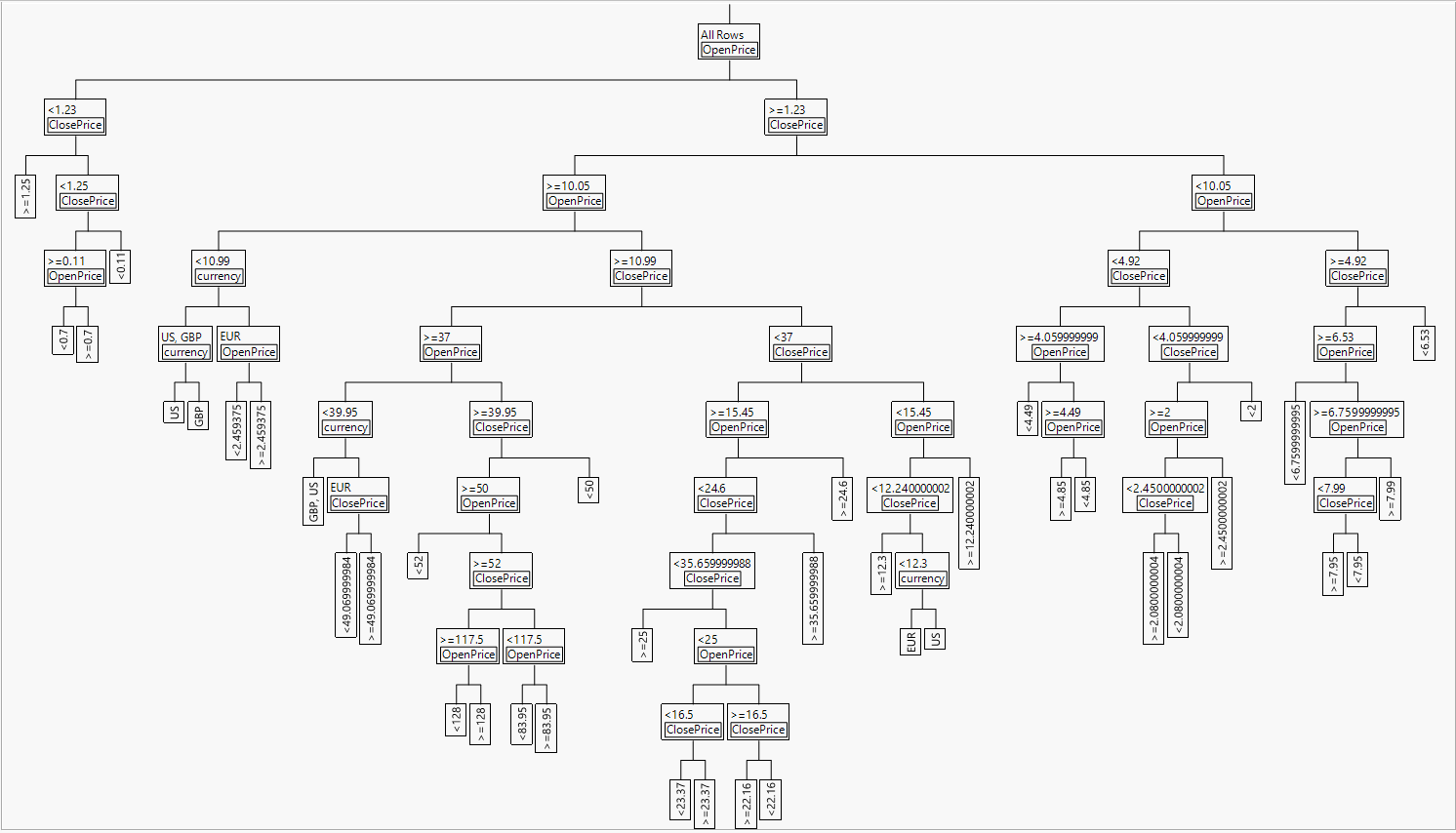
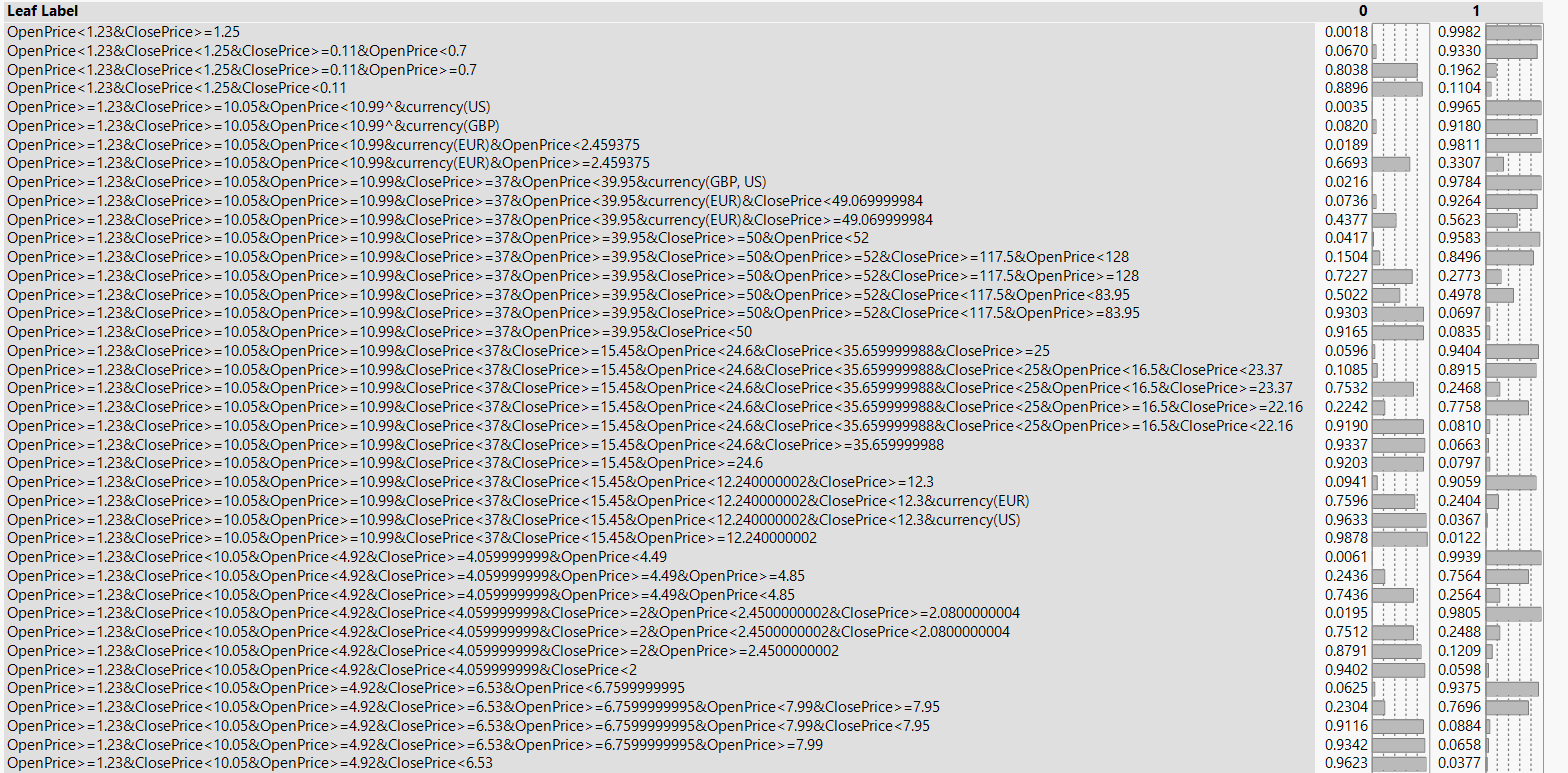
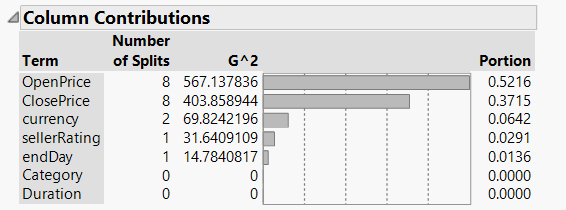
As we can see, the Area under the curve is large. It is approximately 0.90, which is almost 90% of the total area. Since, it is large, it is a practical model to predict outcome of new auction.

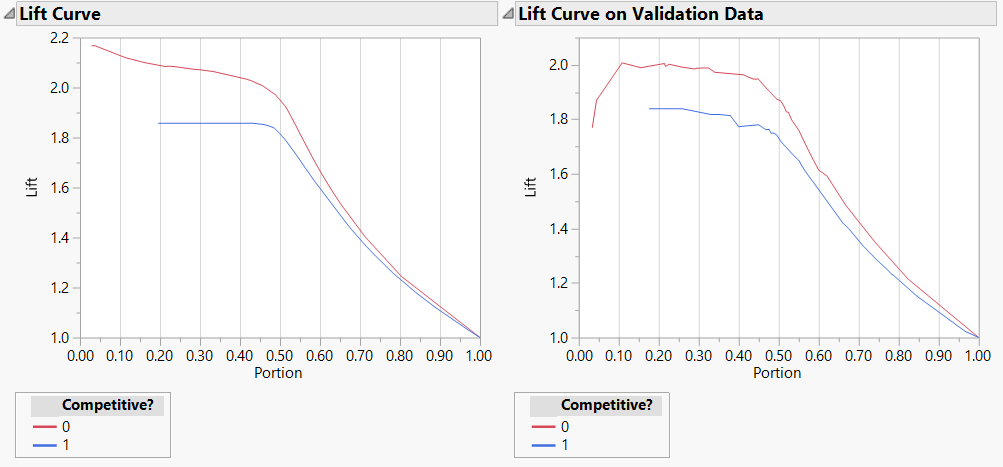
**c. Describe the interesting and uninteresting information that these rules provide.**

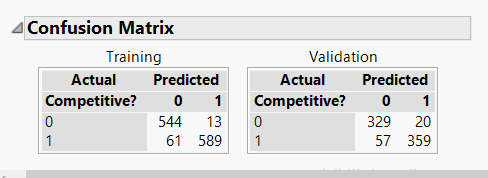
From both tables, we can see that open price has most contribution which is interesting. Also, when it is specifically less than 1.23 and close price is greater than or equal to 1.25, the outcome is 1. Also, the outcome has at least some percentage of 1 if open price is less than 1.23.

What is not interesting is that Duration and Category has exact zero contribution and also seller rating and currency has very less say in the contribution and predicting the outcome.

**d. Fit another classification tree, this time only with predictors that can be used for predicting the outcome of a new auction. Describe the resulting tree in terms of rules.**

Removed the seller rating, end day, category, duration since their contribution is nil or very less. We observe that the tree has increased in size. Also, the contrition of open price and close price is same as before and expected. Also, the leaf report has more rules now. The rules of open price being greater than 1.23 is also in action here as before.

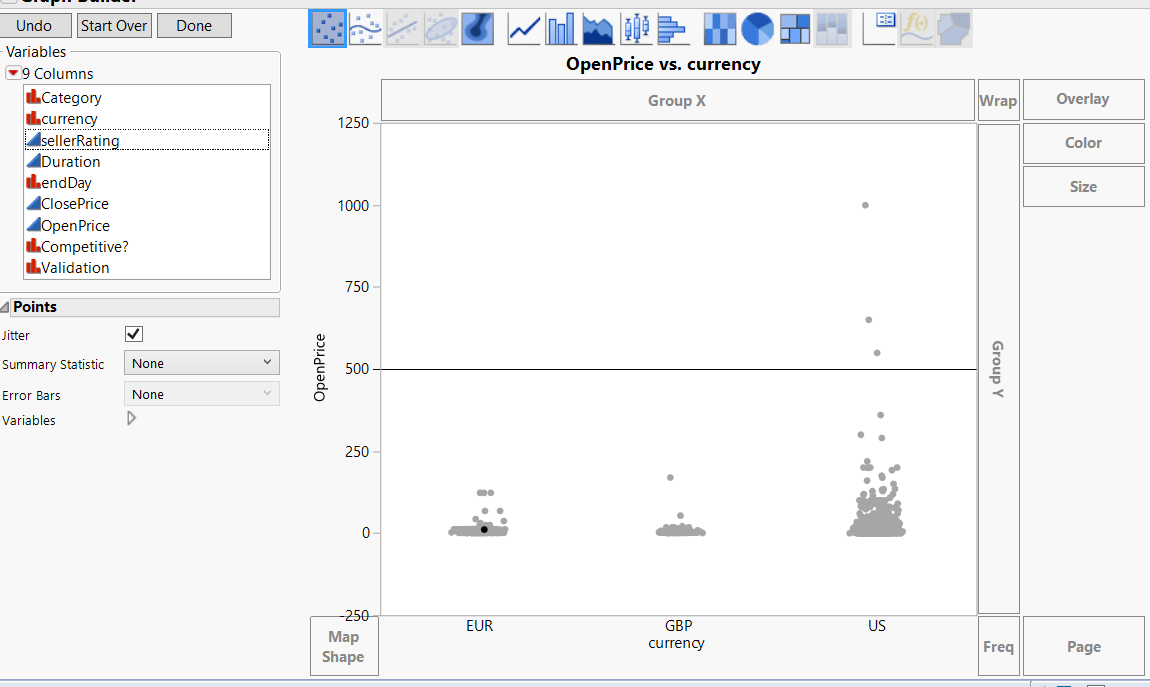
**e. Examine the lift curve and the classification matrix for the tree. What can you say about the predictive performance of this model?**



As we can see, the Lift curves, the graph is almost similar from 0.50 to 1. Which is 50% of the total graph. Also on the validation set, the lift tends to start from the same point for both the competitiveness but then there is a sudden lift and drop in terms of 0 competitiveness.

From the matrix, we can see that the flase negatives and false positives are usually similar and comparatively high values. And vice versa with 1-0,0-1 in both training and validation set.

**f. Plot the resulting tree on a scatterplot: Use the two axes for the two best (quantitative) predictors. Each auction will appear as a point, with coordinates corresponding to its values on those two predictors. Use different colors or symbols to separate competitive and noncompetitive auctions. Draw lines (you can use the line tool in JMP or an axis reference line) at the values that create splits. Does this splitting seem reasonable with respect to the meaning of the two predictors? Does it seem to do a good job of separating the two classes?**



The default axis here is at 500 , which was the default one. Also, it seems to separate the outliers and the class quite well. We can also observe that already any values are above 500. And values in USD are more than any other.

**g. Based on this last tree (d), what can you conclude from these data about the chances of an auction obtaining at least two bids and its relationship to the auction settings set by the seller (duration, opening price, ending day, currency)? What would you recommend for a seller as the strategy that will most likely lead to a competitive auction?**

Following are my recommendations:

1. If the open price is less than 1.23, there is some probability of getting the outcome of 1. So, keep the open price less than 1.23.
2. Close price has shown the outcome 0 very successfully between 4.92 to 6.09. Hence this should be avoided.
3. The combination of close price being less than 10 and open price and currency has yielded a good number of outcomes being 1.
4. The combination of open price lesser than 1.23 and currency is EUR and close price should be more than 1.25 and less than 4.92

**9.3 Predicting Prices of Used Cars (Regression Trees).**

**Data preprocessing. Split the data into training (50%), validation (30%), and**

**test (20%) datasets.**

**a. Run a regression tree with the output variable Price and input variables Age**

**−08−04, KM, Fuel−Type, HP, Automatic, Doors, Quarterly−Tax, Mfg**

**−Guarantee, Guarantee−Period, Airco, Automatic−Airco, CD−Player,**

**Powered−Windows, Sport−Model, and Tow−Bar. Set the minimum split size**

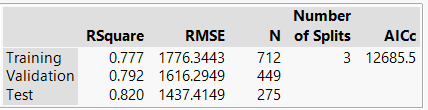
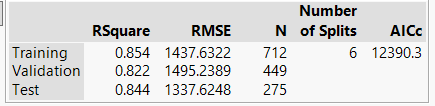
**to 1, and use the split button repeatedly to create a full tree (hint, use the red**

**triangle options to hide the tree and the graph). As you split, keep an eye on**

**RMSE and RSquare for the training, validation and test sets.**

**i. Describe what happens to the RSquare and RMSE for the training,**

**validation and test sets as you continue to split the tree.**

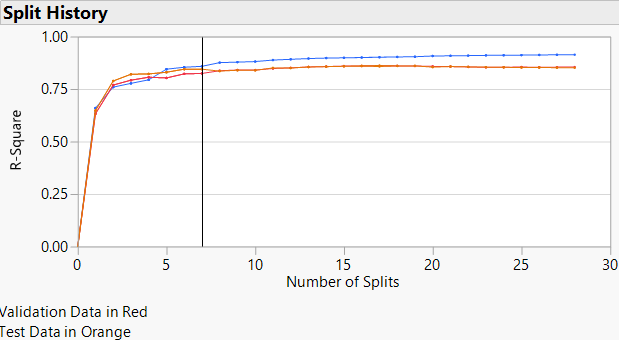


As, I have started to split,

The RSquare value has increased. And RMSE has decresed for all three types of sets.

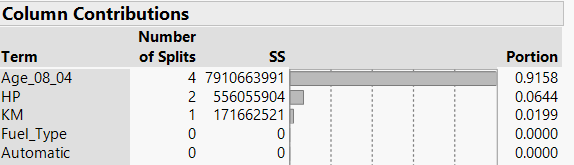
**ii. How does the performance of the test set compare to the training and**

**validation sets on these measures? Why does this occur?**

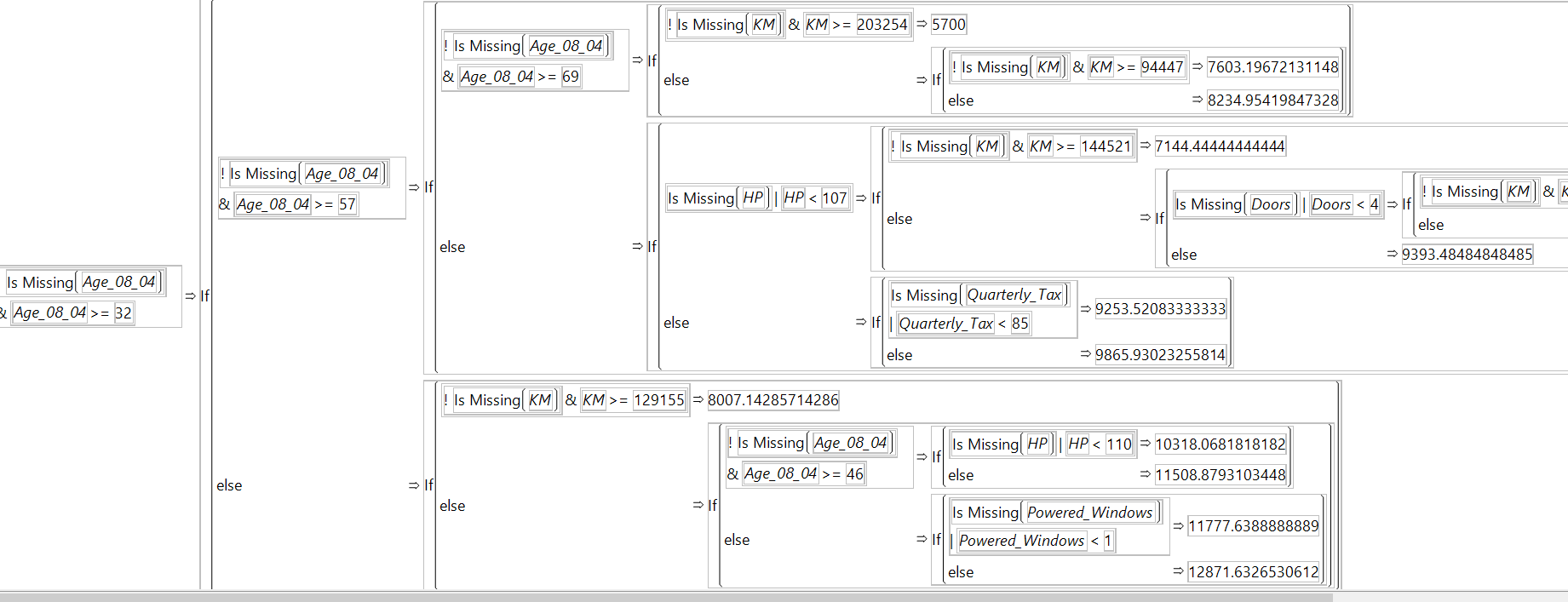


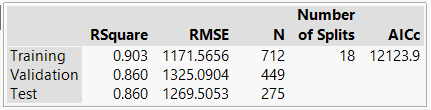
As we can see, the training dataset is nearing R-Square value of 1 which is performing better than the rest. Mostly because the training data has more values and it fits the model well since model is based on training data.

**iii. Based on this tree, which are the most important car specifications for predicting the car's price?**

Age\_08\_04, HP,KM are the most important predictors.

**iv. Refit this model, and use the *Go* button to automatically split and prune the tree based on the validation RSquare. Save the prediction formula for this model to the data table.**

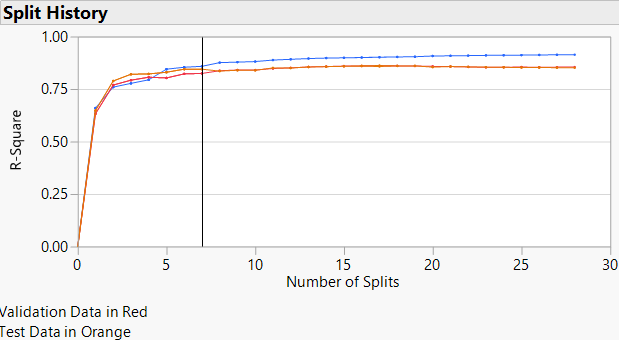


**v. How many splits are in the final tree?**

There are 18 splits in the final tree which is optimal. Since after that, the r-square is quite constant after that.

**vi. Compare RSquare and RMSE for the training, validation and test sets for**

**the reduced model to the full model.**

There is not much difference between RSquare and RMSE, since the reduced model after till 25 splits, the difference is almost negligible. It can be seen in the graph.

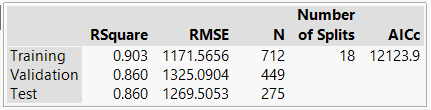
**vii. Which model is better for making predictions? Why?**

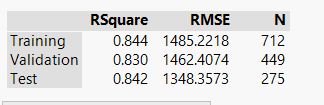
The tree with 18 splits. Since R-square value is highest there and constant almlost afterwards. So less expensive too.

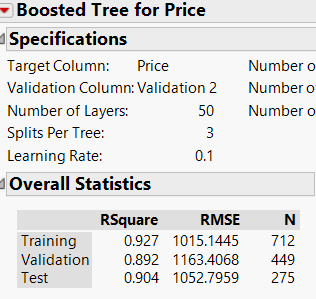
**9.4 Predicting Used Car Prices (Bootstrap Forest and Boosted Trees).**

**Return to the Toyota Corolla data, and refit the partition model. (Hint: Use the recall button in the partition dialog). This time, choose bootstrap forest from the dialog window. Use the default settings.**

**a. Compared to final reduced tree above, how does the bootstrap forest behave**

**in terms of overall error rate on the test set? Save the prediction formula for this model to the data table.**

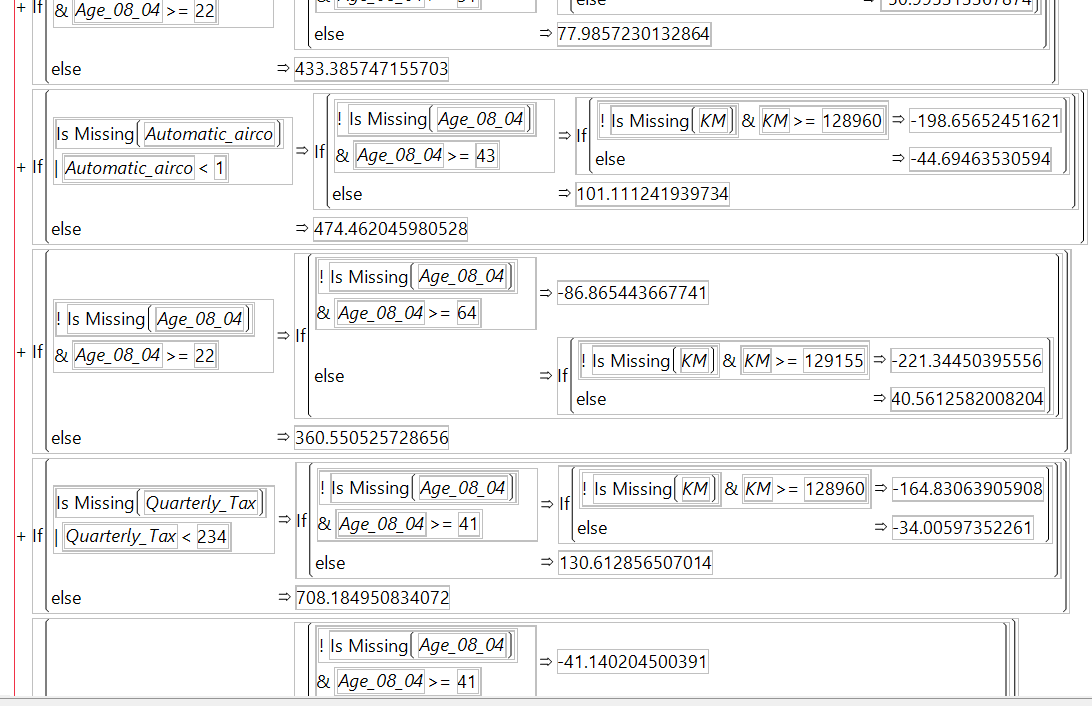
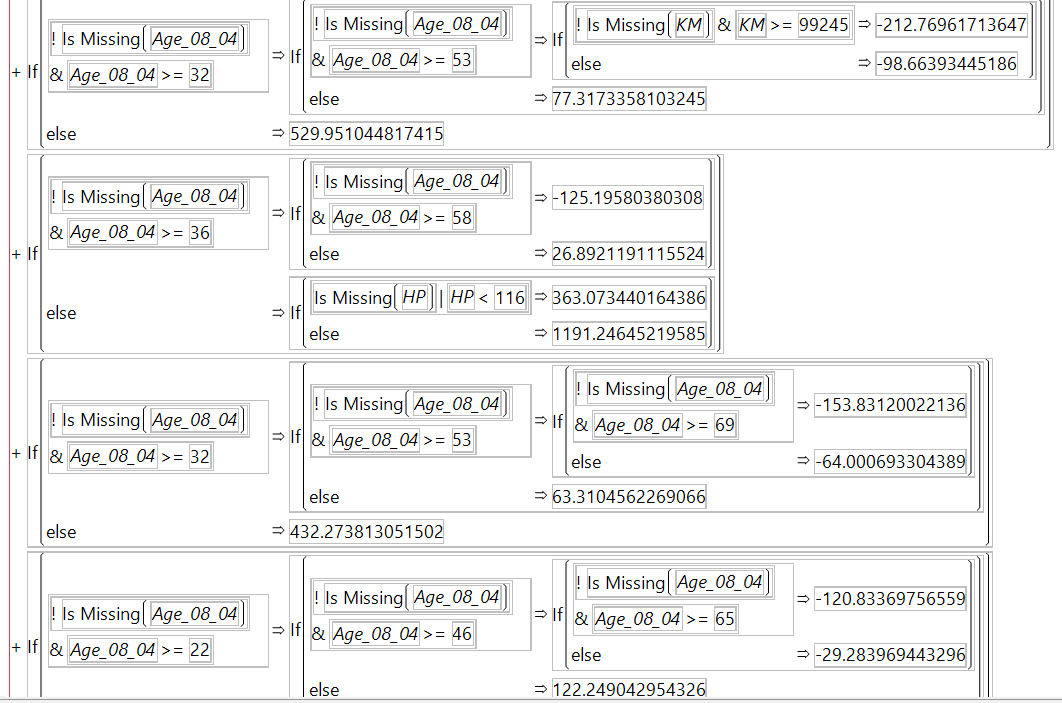
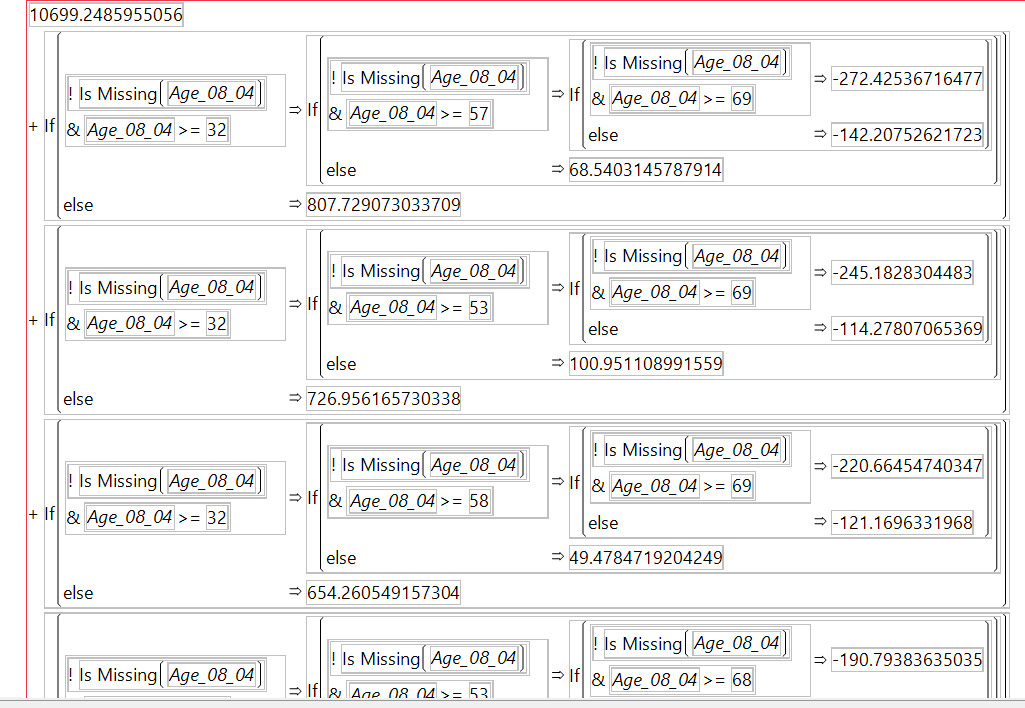
The R-square in Bootstrap is much lesser than the regression tree which shows that tree model is better. The RMSE is much higher in the regression tree which is bad model.

**b. Run the same model again, but this time choose boosted tree from the partition dialog. Use the default settings.**

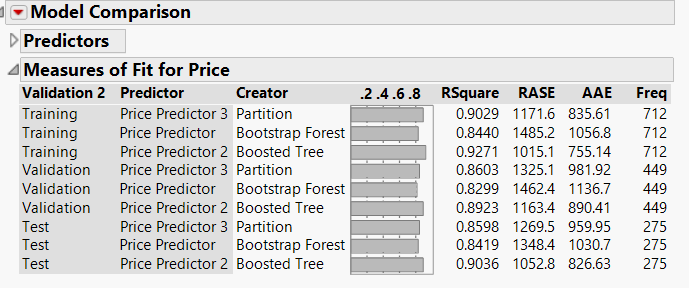
The R-square is more than the bootstrap one but more than the regression tree and the RMSE is much lesser than both. SO this is a good model. Better than the both above so far.

**c. How does the boosted tree behave in terms of the error rate relative to the reduced model and the bootstrap forest. Save the prediction formula for this model to the data table.**

As RMSE value is less, so the error rate is lesser.



d. To facilitate comparison of error rates for the different models, use the *Model Comparison* platform (under *Analyze > Modeling*). To view fit statistics for the different models, put the validation column in the *Group* field in the Model Comparison dialog.



**i. Which model performs best on the test set?**

Boosted tree

**ii. Explain why this model might have the best performance over the other**

**models you fit.**

Because , it might not consider outliers and only succumbs to the most weighted values.

Also, the RMSE Value is much lesser, maybe the error rate is less. Also, the model performs best in this case.