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.

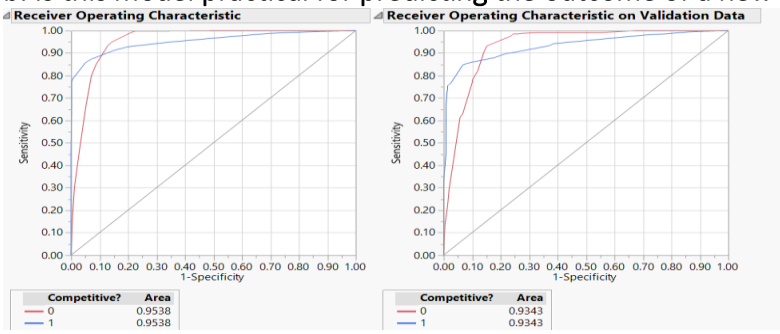
Data preprocessing. Split the data into training and validation datasets using a 60% : 40 % ratio.

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.

Leaf Label	0	1
OpenPrice<1.23&ClosePrice>=1.25	0.0018	0.9982
OpenPrice<1.23&ClosePrice<1.25	0.7142	0.2858
OpenPrice>=1.23&ClosePrice>=10.05&OpenPrice<10.99&currency(US, GBP)	0.0113	0.9887
OpenPrice>=1.23&ClosePrice>=10.05&OpenPrice<10.99&currency(EUR)&OpenPrice<2.459375	0.0189	0.9811
OpenPrice>=1.23&ClosePrice>=10.05&OpenPrice<10.99&currency(EUR)&OpenPrice>=2.459375&endDay(Sat, Fri, Mon)	0.2660	0.7340
OpenPrice>=1.23&ClosePrice>=10.05&OpenPrice<10.99&currency(EUR)&OpenPrice>=2.459375&endDay(Sun, Thu, Wed, Tue)	0.8638	0.1362
OpenPrice>=1.23&ClosePrice>=10.05&OpenPrice>=10.99&sellerRating<572	0.3498	0.6502
OpenPrice>=1.23&ClosePrice>=10.05&OpenPrice>=10.99&sellerRating>=572&ClosePrice>=20.44&OpenPrice<20.95&currency(US)	0.0384	0.9616
OpenPrice>=1.23&ClosePrice>=10.05&OpenPrice>=10.99&sellerRating>=572&ClosePrice>=20.44&OpenPrice<20.95&currency(EUR)	0.5357	0.4643
OpenPrice>=1.23&ClosePrice>=10.05&OpenPrice>=10.99&sellerRating>=572&ClosePrice>=20.44&OpenPrice>=20.95&ClosePrice>=35.99	0.6363	0.3637
OpenPrice>=1.23&ClosePrice>=10.05&OpenPrice>=10.99&sellerRating>=572&ClosePrice>=20.44&OpenPrice>=20.95&ClosePrice<35.99	0.9256	0.0744
OpenPrice>=1.23&ClosePrice>=10.05&OpenPrice>=10.99&sellerRating>=572&ClosePrice<20.44	0.8531	0.1469
OpenPrice>=1.23&ClosePrice<10.05&OpenPrice<4.92&ClosePrice>=4.0599999998&OpenPrice<4.49	0.0061	0.9939
OpenPrice>=1.23&ClosePrice<10.05&OpenPrice<4.92&ClosePrice>=4.0599999998&OpenPrice>=4.49	0.4961	0.5039
OpenPrice>=1.23&ClosePrice<10.05&OpenPrice<4.92&ClosePrice<4.0599999998&ClosePrice>=2&OpenPrice<2.4500000002&ClosePrice>=2.0800000004	0.0195	0.9805
OpenPrice>=1.23&ClosePrice<10.05&OpenPrice<4.92&ClosePrice<4.0599999998&ClosePrice>=2&OpenPrice<2.4500000002&ClosePrice<2.0800000004	0.7512	0.2488
OpenPrice>=1.23&ClosePrice<10.05&OpenPrice<4.92&ClosePrice<4.0599999998&ClosePrice>=2&OpenPrice>=2.4500000002	0.8791	0.1209
OpenPrice>=1.23&ClosePrice<10.05&OpenPrice<4.92&ClosePrice<4.0599999998&ClosePrice<2	0.9402	0.0598
OpenPrice>=1.23&ClosePrice<10.05&OpenPrice>=4.92&ClosePrice>=6.53&OpenPrice<6.7599999995	0.0625	0.9375
OpenPrice>=1.23&ClosePrice<10.05&OpenPrice>=4.92&ClosePrice>=6.53&OpenPrice>=6.7599999995	0.8845	0.1155
OpenPrice>=1.23&ClosePrice<10.05&OpenPrice>=4.92&ClosePrice<6.53	0.9623	0.0377

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.

Leaf Label	0	1
OpenPrice<1.23&ClosePrice>=1.25	0.0018	0.9982
OpenPrice<1.23&ClosePrice<1.25	0.7142	0.2858
OpenPrice>=1.23&ClosePrice>=10.05&OpenPrice<10.99&currency(US, GBP)	0.0113	0.9887
OpenPrice>=1.23&ClosePrice>=10.05&OpenPrice<10.99&currency(EUR)&OpenPrice<2.459375	0.0189	0.9811
OpenPrice>=1.23&ClosePrice>=10.05&OpenPrice<10.99&currency(EUR)&OpenPrice>=2.459375&endDay(Sat, Fri, Mon)	0.2660	0.7340
OpenPrice>=1.23&ClosePrice>=10.05&OpenPrice<10.99&currency(EUR)&OpenPrice>=2.459375&endDay(Sun, Thu, Wed, Tue)	0.8638	0.1362
OpenPrice>=1.23&ClosePrice>=10.05&OpenPrice>=10.99&sellerRating<572	0.3498	0.6502
OpenPrice>=1.23&ClosePrice>=10.05&OpenPrice>=10.99&sellerRating>=572&ClosePrice>=20.44&OpenPrice<20.95&currency(US)	0.0384	0.9616
OpenPrice>=1.23&ClosePrice>=10.05&OpenPrice>=10.99&sellerRating>=572&ClosePrice>=20.44&OpenPrice<20.95&currency(EUR)	0.5357	0.4643
OpenPrice>=1.23&ClosePrice>=10.05&OpenPrice>=10.99&sellerRating>=572&ClosePrice>=20.44&OpenPrice>=20.95&ClosePrice>=35.99	0.6363	0.3637
OpenPrice>=1.23&ClosePrice>=10.05&OpenPrice>=10.99&sellerRating>=572&ClosePrice>=20.44&OpenPrice>=20.95&ClosePrice<35.99	0.9256	0.0744
OpenPrice>=1.23&ClosePrice>=10.05&OpenPrice>=10.99&sellerRating>=572&ClosePrice<20.44	0.8531	0.1469
OpenPrice>=1.23&ClosePrice<10.05&OpenPrice<4.92&ClosePrice>=4.0599999998&OpenPrice<4.49	0.0061	0.9939
OpenPrice>=1.23&ClosePrice<10.05&OpenPrice<4.92&ClosePrice>=4.0599999998&OpenPrice>=4.49	0.4961	0.5039
OpenPrice>=1.23&ClosePrice<10.05&OpenPrice<4.92&ClosePrice<4.0599999998&ClosePrice>=2&OpenPrice<2.4500000002&ClosePrice>=2.0800000004	0.0195	0.9805
OpenPrice>=1.23&ClosePrice<10.05&OpenPrice<4.92&ClosePrice<4.0599999998&ClosePrice>=2&OpenPrice<2.4500000002&ClosePrice<2.0800000004	0.7512	0.2488
OpenPrice>=1.23&ClosePrice<10.05&OpenPrice<4.92&ClosePrice<4.0599999998&ClosePrice>=2&OpenPrice>=2.4500000002	0.8791	0.1209
OpenPrice>=1.23&ClosePrice<10.05&OpenPrice<4.92&ClosePrice<4.0599999998&ClosePrice<2	0.9402	0.0598
OpenPrice>=1.23&ClosePrice<10.05&OpenPrice>=4.92&ClosePrice>=6.53&OpenPrice<6.7599999995	0.0625	0.9375
OpenPrice>=1.23&ClosePrice<10.05&OpenPrice>=4.92&ClosePrice>=6.53&OpenPrice>=6.7599999995	0.8845	0.1155
OpenPrice>=1.23&ClosePrice<10.05&OpenPrice>=4.92&ClosePrice<6.53	0.9623	0.0377

Column Contributions

Term	Number of Splits	G^2	Portion
OpenPrice	8	567.137836	0.5216
ClosePrice	8	403.858944	0.3715
currency	2	69.8242196	0.0642
sellerRating	1	31.6409109	0.0291
endDay	1	14.7840817	0.0136
Category	0	0	0.0000
Duration	0	0	0.0000

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

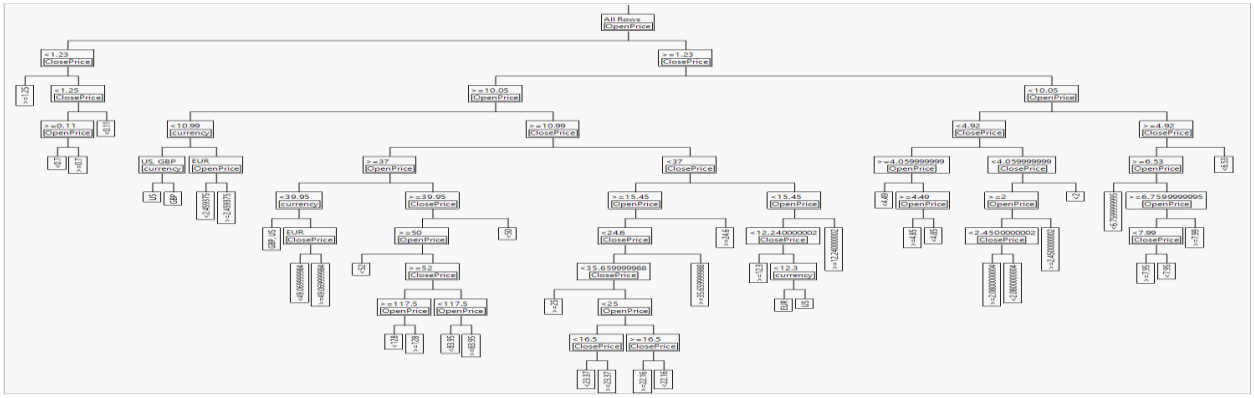
outcome has at least some percentage of 1 if open price is less than 1.23.
What is not interesting is that Duration and Category has exact zero contribution and also seller rating and currency has very less say in the contribution and predicting the outcome.

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.

Column Contributions

Term	Number of Splits	G^2	Portion
OpenPrice	8	567.137836	0.5216
ClosePrice	8	403.858944	0.3715
currency	2	69.8242196	0.0642
sellerRating	1	31.6409109	0.0291
endDay	1	14.7840817	0.0136
Category	0	0	0.0000
Duration	0	0	0.0000

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.



Leaf Label	0	1
OpenPrice<1.23&ClosePrice>=1.25	0.0018	0.9982
OpenPrice<1.23&ClosePrice<1.25&ClosePrice>=0.11&OpenPrice<0.7	0.0670	0.9330
OpenPrice<1.23&ClosePrice<1.25&ClosePrice>=0.11&OpenPrice>=0.7	0.8038	0.1962
OpenPrice<1.23&ClosePrice<1.25&ClosePrice<0.11	0.8896	0.1104
OpenPrice>=1.23&ClosePrice>=10.05&OpenPrice<10.99^&currency(US)	0.0035	0.9965
OpenPrice>=1.23&ClosePrice>=10.05&OpenPrice<10.99^&currency(GBP)	0.0820	0.9180
OpenPrice>=1.23&ClosePrice>=10.05&OpenPrice<10.99&currency(EUR)&OpenPrice<2.459375	0.0189	0.9811
OpenPrice>=1.23&ClosePrice>=10.05&OpenPrice<10.99&currency(EUR)&OpenPrice>=2.459375	0.6693	0.3307
OpenPrice>=1.23&ClosePrice>=10.05&OpenPrice>=10.99&ClosePrice>=37&OpenPrice<39.95&currency(GBP, US)	0.0216	0.9784
OpenPrice>=1.23&ClosePrice>=10.05&OpenPrice>=10.99&ClosePrice>=37&OpenPrice<39.95&currency(EUR)&ClosePrice<49.069999984	0.0736	0.9264
OpenPrice>=1.23&ClosePrice>=10.05&OpenPrice>=10.99&ClosePrice>=37&OpenPrice<39.95&currency(EUR)&ClosePrice>=49.069999984	0.4377	0.5623
OpenPrice>=1.23&ClosePrice>=10.05&OpenPrice>=10.99&ClosePrice>=37&OpenPrice>=39.95&ClosePrice>=50&OpenPrice<52	0.0417	0.9583
OpenPrice>=1.23&ClosePrice>=10.05&OpenPrice>=10.99&ClosePrice>=37&OpenPrice>=39.95&ClosePrice>=50&OpenPrice>=52&ClosePrice>=117.5&OpenPrice<128	0.1504	0.8496
OpenPrice>=1.23&ClosePrice>=10.05&OpenPrice>=10.99&ClosePrice>=37&OpenPrice>=39.95&ClosePrice>=50&OpenPrice>=52&ClosePrice>=117.5&OpenPrice>=128	0.7227	0.2773
OpenPrice>=1.23&ClosePrice>=10.05&OpenPrice>=10.99&ClosePrice>=37&OpenPrice>=39.95&ClosePrice>=50&OpenPrice>=52&ClosePrice<117.5&OpenPrice<83.95	0.5022	0.4978
OpenPrice>=1.23&ClosePrice>=10.05&OpenPrice>=10.99&ClosePrice>=37&OpenPrice>=39.95&ClosePrice>=50&OpenPrice>=52&ClosePrice<117.5&OpenPrice>=83.95	0.9303	0.0697
OpenPrice>=1.23&ClosePrice>=10.05&OpenPrice>=10.99&ClosePrice>=37&OpenPrice>=39.95&ClosePrice<50	0.9165	0.0835
OpenPrice>=1.23&ClosePrice>=10.05&OpenPrice>=10.99&ClosePrice<37&ClosePrice>=15.45&OpenPrice<24.6&ClosePrice<35.659999988&ClosePrice>=25	0.0596	0.9404
OpenPrice>=1.23&ClosePrice>=10.05&OpenPrice>=10.99&ClosePrice<37&ClosePrice>=15.45&OpenPrice<24.6&ClosePrice<35.659999988&ClosePrice<25&OpenPrice<16.5&ClosePrice<23.37	0.1085	0.8915
OpenPrice>=1.23&ClosePrice>=10.05&OpenPrice>=10.99&ClosePrice<37&ClosePrice>=15.45&OpenPrice<24.6&ClosePrice<35.659999988&ClosePrice<25&OpenPrice<16.5&ClosePrice>=23.37	0.7532	0.2468
OpenPrice>=1.23&ClosePrice>=10.05&OpenPrice>=10.99&ClosePrice<37&ClosePrice>=15.45&OpenPrice<24.6&ClosePrice<35.659999988&ClosePrice<25&OpenPrice>=16.5&ClosePrice>=22.16	0.2242	0.7758
OpenPrice>=1.23&ClosePrice>=10.05&OpenPrice>=10.99&ClosePrice<37&ClosePrice>=15.45&OpenPrice<24.6&ClosePrice<35.659999988&ClosePrice<25&OpenPrice>=16.5&ClosePrice<22.16	0.9190	0.0810
OpenPrice>=1.23&ClosePrice>=10.05&OpenPrice>=10.99&ClosePrice<37&ClosePrice>=15.45&OpenPrice<24.6&ClosePrice>=35.659999988	0.9337	0.0663
OpenPrice>=1.23&ClosePrice>=10.05&OpenPrice>=10.99&ClosePrice<37&ClosePrice>=15.45&OpenPrice>=24.6	0.9203	0.0797
OpenPrice>=1.23&ClosePrice>=10.05&OpenPrice>=10.99&ClosePrice<37&ClosePrice<15.45&OpenPrice<12.2400000002&ClosePrice>=12.3	0.0941	0.9059
OpenPrice>=1.23&ClosePrice>=10.05&OpenPrice>=10.99&ClosePrice<37&ClosePrice<15.45&OpenPrice<12.2400000002&ClosePrice<12.3&currency(EUR)	0.7596	0.2404
OpenPrice>=1.23&ClosePrice>=10.05&OpenPrice>=10.99&ClosePrice<37&ClosePrice<15.45&OpenPrice<12.2400000002&ClosePrice<12.3&currency(US)	0.9633	0.0367
OpenPrice>=1.23&ClosePrice>=10.05&OpenPrice>=10.99&ClosePrice<37&ClosePrice<15.45&OpenPrice>=12.2400000002	0.9878	0.0122
OpenPrice>=1.23&ClosePrice<10.05&OpenPrice<4.92&ClosePrice>=4.0599999998&OpenPrice<4.49	0.0061	0.9939
OpenPrice>=1.23&ClosePrice<10.05&OpenPrice<4.92&ClosePrice>=4.0599999998&OpenPrice>=4.49&OpenPrice>=4.85	0.2436	0.7564
OpenPrice>=1.23&ClosePrice<10.05&OpenPrice<4.92&ClosePrice>=4.0599999998&OpenPrice>=4.49&OpenPrice<4.85	0.7436	0.2564
OpenPrice>=1.23&ClosePrice<10.05&OpenPrice<4.92&ClosePrice<4.0599999998&ClosePrice>=2&OpenPrice<2.45000000002&ClosePrice>=2.08000000004	0.0195	0.9805
OpenPrice>=1.23&ClosePrice<10.05&OpenPrice<4.92&ClosePrice<4.0599999998&ClosePrice>=2&OpenPrice>=2.45000000002&ClosePrice<2.08000000004	0.7512	0.2488
OpenPrice>=1.23&ClosePrice<10.05&OpenPrice<4.92&ClosePrice<4.0599999998&ClosePrice>=2&OpenPrice>=2.4500000002	0.8791	0.1209
OpenPrice>=1.23&ClosePrice<10.05&OpenPrice<4.92&ClosePrice<4.0599999998&ClosePrice<2	0.9402	0.0598
OpenPrice>=1.23&ClosePrice<10.05&OpenPrice>=4.92&ClosePrice>=6.53&OpenPrice<6.7599999995	0.0625	0.9375
OpenPrice>=1.23&ClosePrice<10.05&OpenPrice>=4.92&ClosePrice>=6.53&OpenPrice>=6.7599999995&OpenPrice<7.99&ClosePrice>=7.95	0.2304	0.7696
OpenPrice>=1.23&ClosePrice<10.05&OpenPrice>=4.92&ClosePrice>=6.53&OpenPrice>=6.7599999995&OpenPrice<7.99&ClosePrice<7.95	0.9116	0.0884
OpenPrice>=1.23&ClosePrice<10.05&OpenPrice>=4.92&ClosePrice>=6.53&OpenPrice>=6.7599999995&OpenPrice>=7.99	0.9342	0.0658
OpenPrice>=1.23&ClosePrice<10.05&OpenPrice>=4.92&ClosePrice<6.53	0.9623	0.0377

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

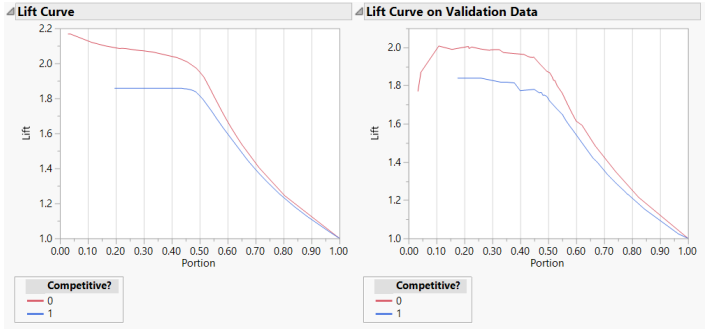
Confusion Matrix

Training

Actual Competitive?	Predicted 0	Predicted 1
0	544	13
1	61	589

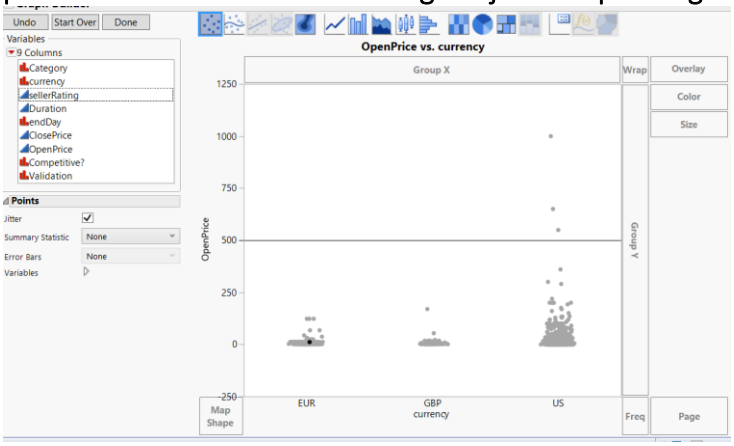
Validation

Actual Competitive?	Predicted 0	Predicted 1
0	329	20
1	57	359



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.

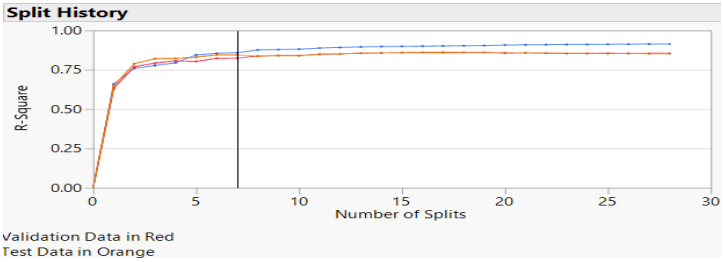
i. Describe what happens to the RSquare and RMSE for the training, validation and test sets as you continue to split the tree.

	RSquare	RMSE	N	Number of Splits	AICc
Training	0.777	1776.3443	712	3	12685.5
Validation	0.792	1616.2949	449		
Test	0.820	1437.4149	275		

	RSquare	RMSE	N	Number of Splits	AICc
Training	0.854	1437.6322	712	6	12390.3
Validation	0.822	1495.2389	449		
Test	0.844	1337.6248	275		

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.

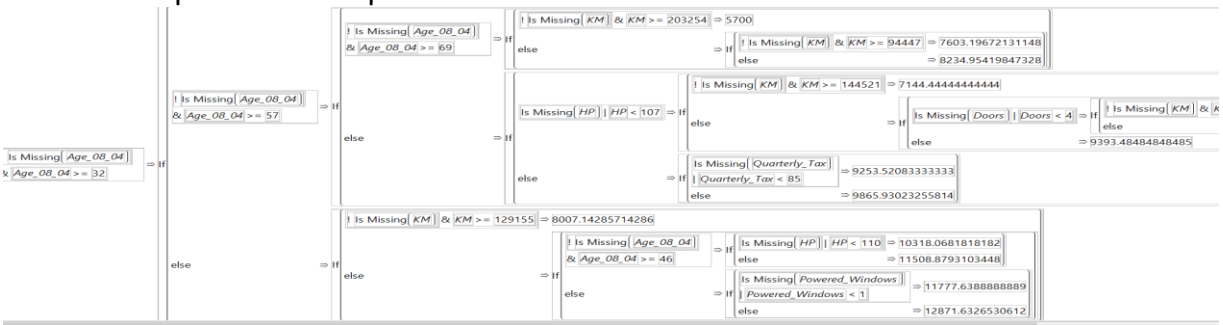
iii. Based on this tree, which are the most important car specifications for predicting the car's price?

Column Contributions

Term	Number of Splits	SS	Portion
Age_08_04	4	7910663991	0.9158
HP	2	556055904	0.0644
KM	1	171662521	0.0199
Fuel_Type	0	0	0.0000
Automatic	0	0	0.0000

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.

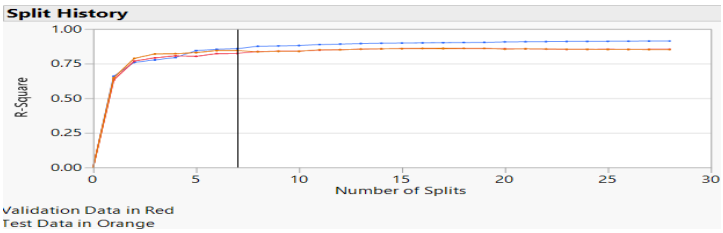


v. How many splits are in the final tree?

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

	RSquare	RMSE	N	Number of Splits	AICc
Training	0.903	1171.5656	712	18	12123.9
Validation	0.860	1325.0904	449		
Test	0.860	1269.5053	275		

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.

	RSquare	RMSE	N
Training	0.844	1485.2218	712
Validation	0.830	1462.4074	449
Test	0.842	1348.3573	275

	RSquare	RMSE	N	Number of Splits	AICc
Training	0.903	1171.5656	712	18	12123.9
Validation	0.860	1325.0904	449		
Test	0.860	1269.5053	275		

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.

Boosted Tree for Price

Specifications

Target Column:	Price	Number o
Validation Column:	Validation 2	Number o
Number of Layers:	50	Number o
Splits Per Tree:	3	
Learning Rate:	0.1	

Overall Statistics

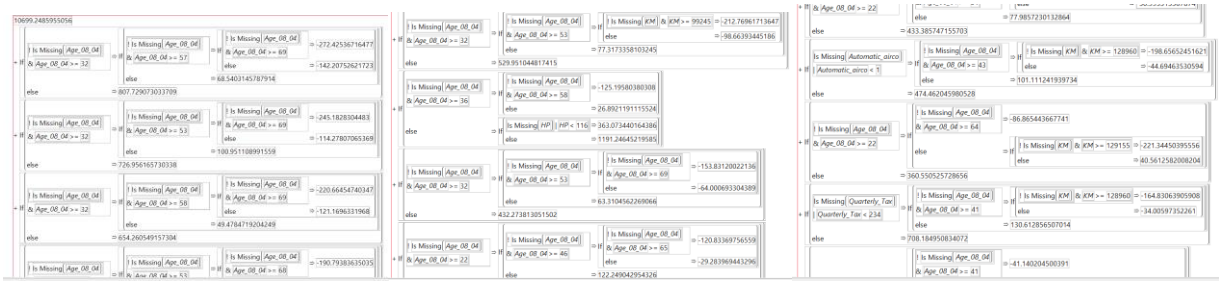
	RSquare	RMSE	N
Training	0.927	1015.1445	712
Validation	0.892	1163.4068	449
Test	0.904	1052.7959	275

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.

Model Comparison

Predictors

Measures of Fit for Price

Validation 2	Predictor	Creator	.2 .4 .6 .8	RSquare	RASE	AAE	Freq
Training	Price Predictor 3	Partition		0.9029	1171.6	835.61	712
Training	Price Predictor	Bootstrap Forest		0.8440	1485.2	1056.8	712
Training	Price Predictor 2	Boosted Tree		0.9271	1015.1	755.14	712
Validation	Price Predictor 3	Partition		0.8603	1325.1	981.92	449
Validation	Price Predictor	Bootstrap Forest		0.8299	1462.4	1136.7	449
Validation	Price Predictor 2	Boosted Tree		0.8923	1163.4	890.41	449
Test	Price Predictor 3	Partition		0.8598	1269.5	959.95	275
Test	Price Predictor	Bootstrap Forest		0.8419	1348.4	1030.7	275
Test	Price Predictor 2	Boosted Tree		0.9036	1052.8	826.63	275

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.