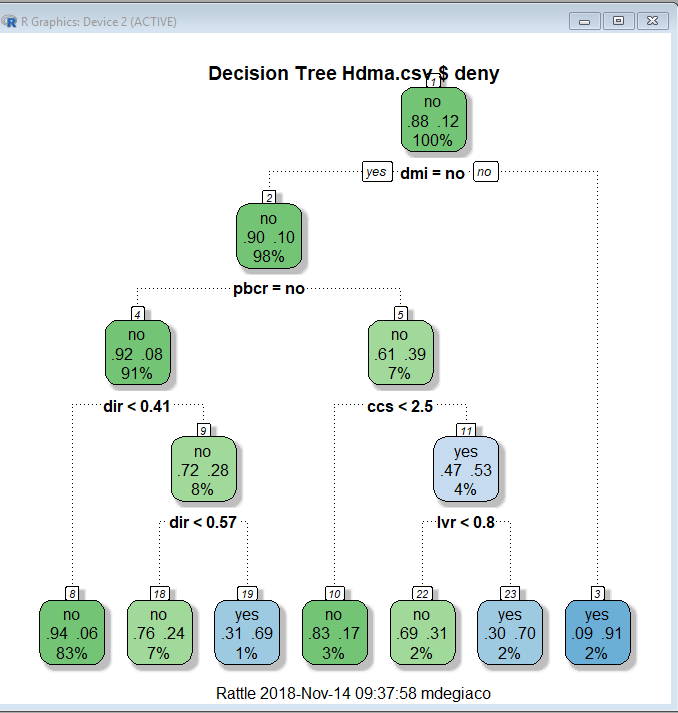
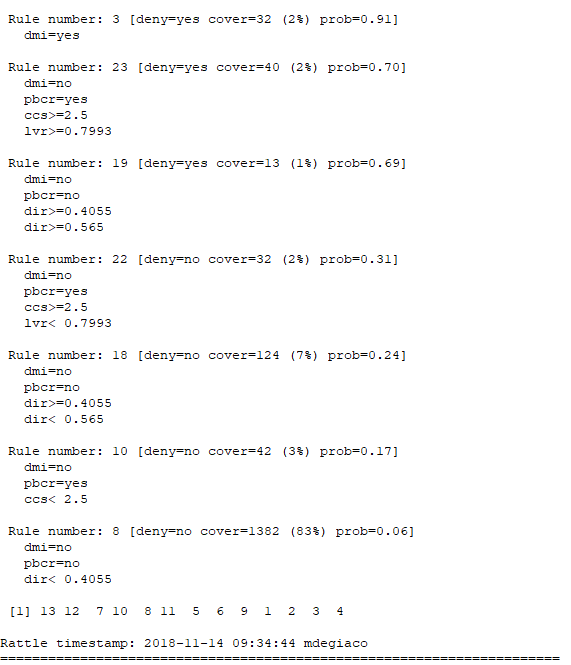
Matthew DeGiacomo

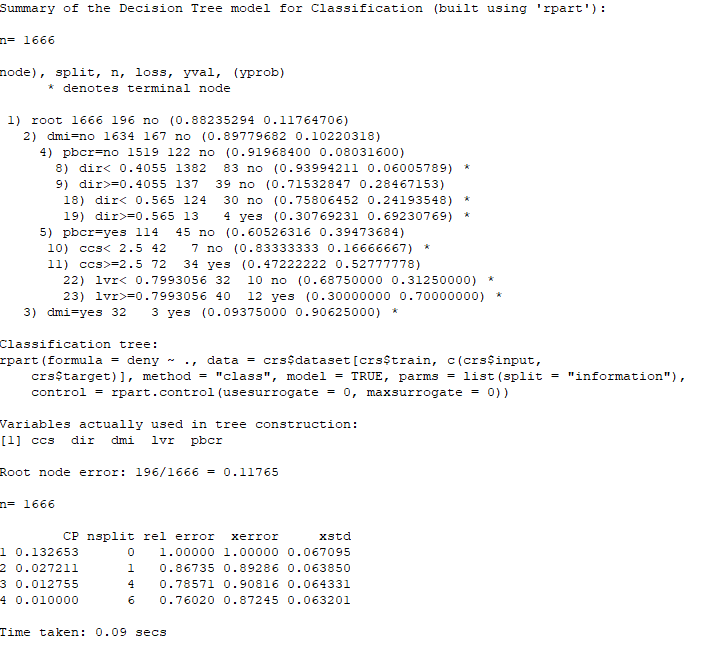
Mod 7 R Activity

11/14/2018

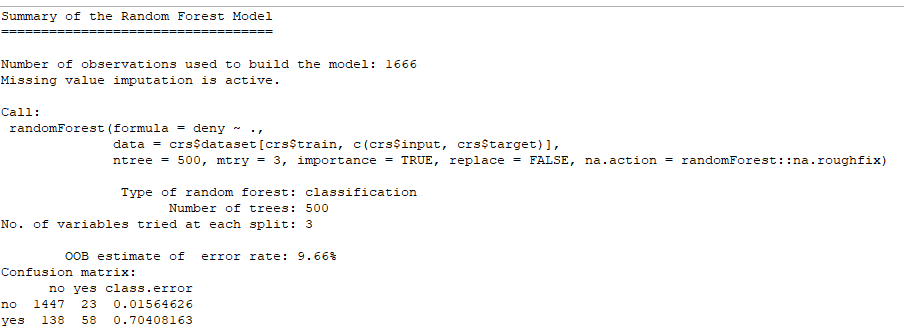
For my decision tree and random forest I used the Ecdat HDMA data set where my target variable was “deny: if a person got denied a mortgage or not”. I started with a decision tree.

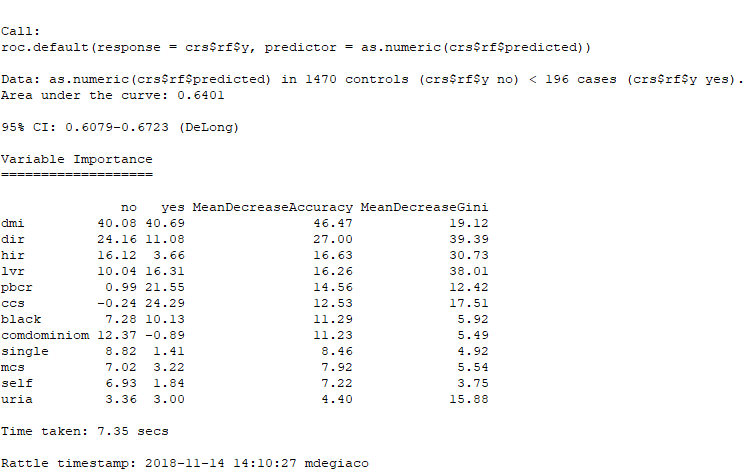


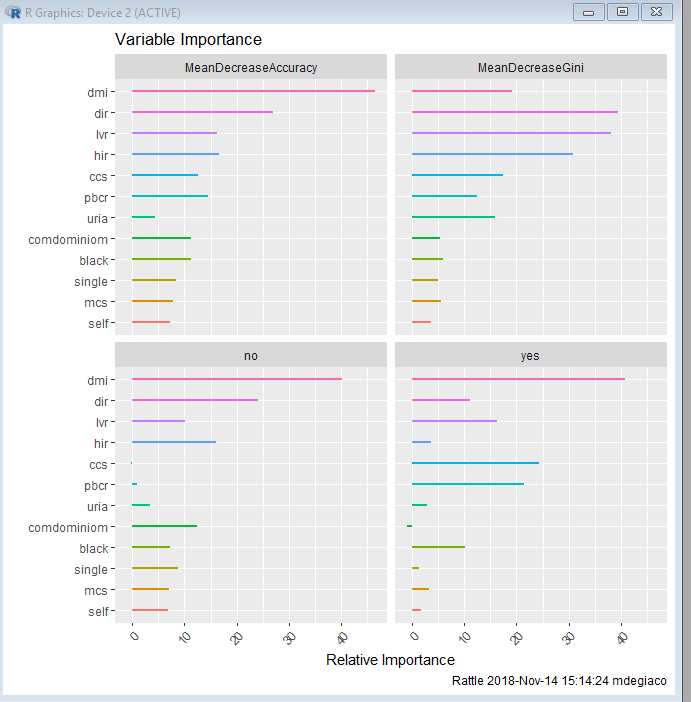
I went with the traditional algorithm in rattle and went with the default parameters. To summarize this tree a couple of things stuck out to me that I noticed. First, if you see rule 8 below, it indicates that this rule covers 1382 observations out 1666 so 83% of the data set follows this rule, that 6% of the observations where they were not denied mortgage insurance (dmi), the did not have a bad public credit record, and the debt to income ratio was less than 40% there mortgage application was not denied. As you can see below as well, the error rate of this tree is 11% so 89% accuracy which is really good. Out of 13 variables, only 5 were used in the tree construction excluding the target variable and ident, the 5 were: consumer credit score, debt to income ratio, denied mortgage insurance?, ratio of size of loan to assessed value of property, and if they had a public bad credit record. Looking at the rules and adding up how much of the data set they account for, between 3 rules (#3, #23, and #19) only 5% of the dataset was denied on their mortgage application. Another thing I found interesting was comparing rules 23 and 10 was that rule 23 they were denied and rule 10 they were approved but within those rules, they both were not denied mortgage insurance, they both had bad public credit, but the difference was the people who got denied had a credit score greater than 2.5 and the ones who didn’t had one under 2.5. Also Rule 23 they got denied 70% of the time with those variables but of rule 10 they only got approved 17% of the time, so those two rules are very close.

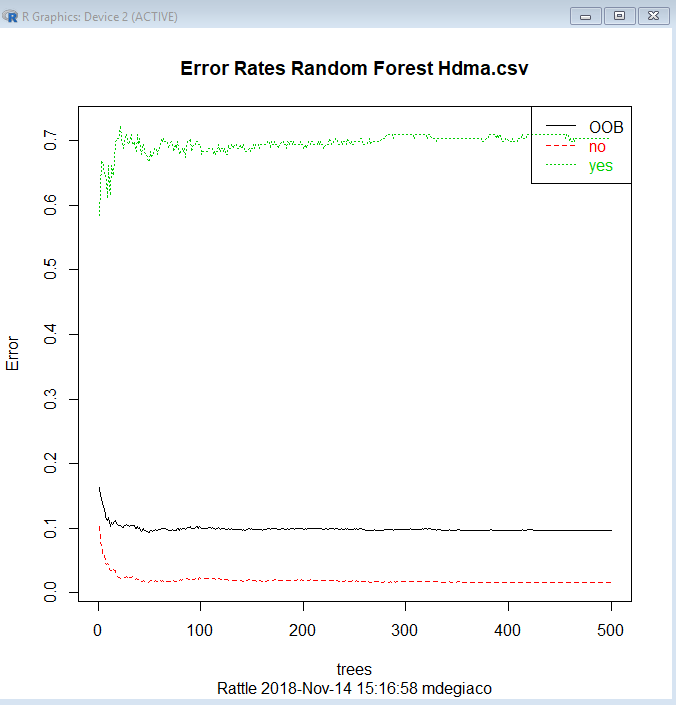


For my next model, I did a random forest with the same data in a traditional algorism form with 500 trees and 3 variables. Below are the text view of the forest, Variable importance chart, and Error rate chart from the Random Forest Model.









What these snapshots tell you about the data in a Random Forest Model, is first the model has an OOB error rate of 9.66% which means it 90% accurate which is fantastic for a model to be 90% accurate. Then you take a look at the confusion matrix, 1447/1470 observations it was predicted that they were not going to get denied on their mortgage application and did not which is 0.015 error rate and 98% accuracy on the flipside though, 138/196 times it was predicted that a person was going to get denied but they did not end up getting denied which is 0.70 error rate and 30% accuracy. Then you can see the Variable importance chart in just chart form and then in graph form, the most important variables in this set are: denied mortgage insurance, debt to income ratio, housing expenses to income ratio, ratio size of the loan in proportion to the property value, public credit score, and consumer credit score. The first section “no” is how important that variable is to not getting denied on your application, the “yes” is how important that variable is the getting denied. As you can see the consumer credit score is negative with the no but then has the second highest score with the yes variable. Third column is a scaled average of the prediction accuracy of each variable, as you can see the variable denied mortgage insurance has the highest score, and then a drop of half for the second highest score, so the DMI variable is the most accurate one. The last row, the gini score is the total decrease in the nodes impurity when being split, debt to income ratio has the highest score for this row. The last chart is the error matrix chart and as you can see after about 50 tree’s in the random forest (we did 500) there is very little no change in the error rate of the random forest model, this chart helps us decide how many tree’s to add to our model when doing readjustments. In reality, compared to my discussion post they both ended up being easy to read models, the random forest is 1% more accurate than the tree, I noticed for variables that have more importance in my forest than what was shown in my tree, and overall if I don’t have trouble reading the forest, I would choose the forest but I don’t like not having a visual tree like a decision tree gives you.

REFERENCES:

* Williams, G. J. (2013). Data mining with Rattle and R: The art of excavating data for knowledge discovery. New York: Springer.