Damon Quire

CIS 445

Project 2

Some of the assumptions made in this project are that there are only two outcomes which determines if the target is in fact binary or not. In this case, we assume that there are only two options available, “Yes” and “No”. In some similar cases there may be more than two outcomes and the sample just didn’t have a record with that outcome. For instance, one of the options for a loan decision tree could be “delay loan request for further inspection”. If the data set you had for this hypothetical decision tree did not include this outcome, the computer would have no way of knowing it exists. Therefor, the largest assumption you make when doing these types of things is that your sample is an accurate representation of the entire population. If not, key points or outcomes may go unnoticed. Had all of these data records been outliers, the results could blur or blind you from the correct outcome classification of the population itself. If these were situations that normally wouldn’t be assigned the outcome that they were assigned, it could cause the rules of the decision tree to be wrong, the variable importance to be wrong, and many other things to be incorrectly examined and the wrong hypothesis to be made. Another assumption that is made is that all of the nominal variables like residence are included. There are none from the population that were left out that could be better at determining outcomes, had their been a specific location of residence that was left out that was a stronger predictor of outcome, it could easily throw Residence’s variable importance severely off. With data sets this small, it is easy to get a sample that is indeed not a great representation of the population. It’s similar to many things in every day life, for instance this most recent election, every poll take on social media or other sites said Hillary would win in a landslide victory, however, the population is the United States a s a whole, and what they received were input from active social media persons which are typically democrats. This caused the predictions to be incorrect and skewed because the sample which they used to try and predict the outcome of the election based on the whole population only had or typically had people that were not an accurate distribution off the differing opinions of those in the united states. It is very easy to run all the correct prediction models and rule models but simply have an inaccurate sample that will horribly misinterpret the results of the predicted population.

The most important variable seems to be income. This is gathered from the variable importance being tremendously higher than all of the other variables. This means that this variable plays the largest role in determining the outcome of the data record itself. It alone plays a majority role in determining the “Yes” or “No” outcome that is discovered. It’s also discovered by the fact that the decision tree is split by income first, meaning this is the largest separator of outcomes. This is also relatively self-evident in any kind of purchasing or loan decision, income always plays a huge factor since it’s about money in the long run. This was true when we calculated gain from splitting income first in the extra credit portion of our homework when we calculated gain from this split vs the age split and the residence split. It is a pretty safe bet for income to be a determining factor in most cases like this one. This is why on all loan applications, credit card applications, and other requests to borrow money or secure money, the most important thing on the app is the income field. This is what supplies the basis for a lot of money oriented decisions.

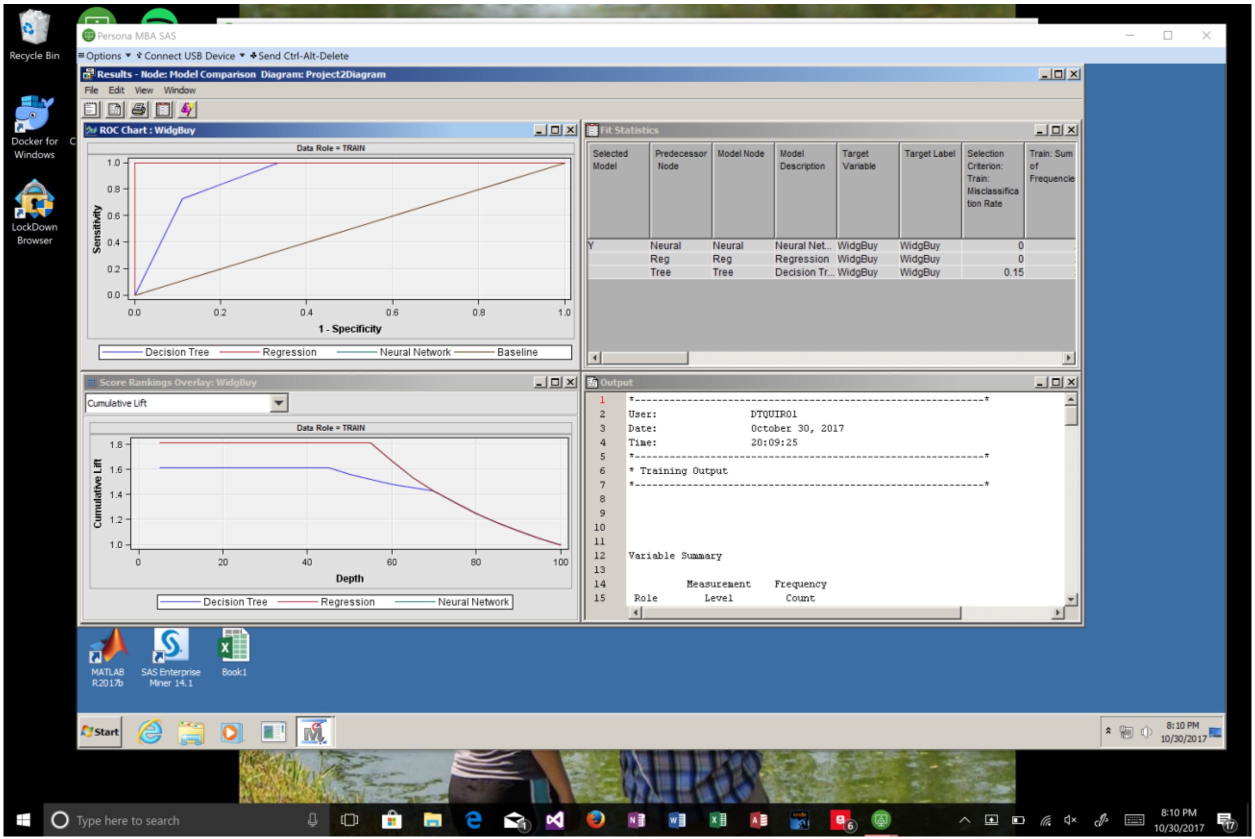
The rules of the decision tree are as follows: if income is high or missing, 72.73% of the time the outcome is “No”. If income is low the outcome is “Yes” 88.89% of the time. If the income is high or missing AND the age is below 30.5, the outcome is “YES” 60% of the time. If the income is high AND the age is above of equal to 30.5 the outcome is “No” 100% of the time, meaning this is the only path that is a clear decider of outcomes, and the only time the outcome is “Yes” with a high or missing income is when their age is under 30.5, otherwise, the answer is always “No”. So there is only one path with a guaranteed outcome which is interesting. The other path(s) would need more variables or factors to be able to guarantee the outcome. The variables that are available are not capable of doing that for this data set in all paths.

The weight is the highest for Residence=CHI. I’m assuming that this is due to the fact that every single instance of CHI for residence has the same outcome. Meaning that this is not a determining factor per say but is an interesting result nonetheless. Income being high is also weighted very heavily which means that it is usually a pointer to a certain outcome. So this variable income is the most important variable, and income being high usually points to a certain outcome. Variables X2 and Residence=LA are not good at all at pointing to a certain outcome because when they appear, the outcome is hard to distinguish from those alone. The darker the red on the chart, the more heavily weighted that variable instance is.

From the ROC chart it looks like the regression model is the best model to use because it curves straight and to the left, leaving it in the top left corner longer than the other two models. This also remains true for the Lift chart, the regression model has a higher reach point on the graph and is the better model to predict and visualize the data. This means the regression model is the best model to use for this specific data set. Logistic regression to be specific.

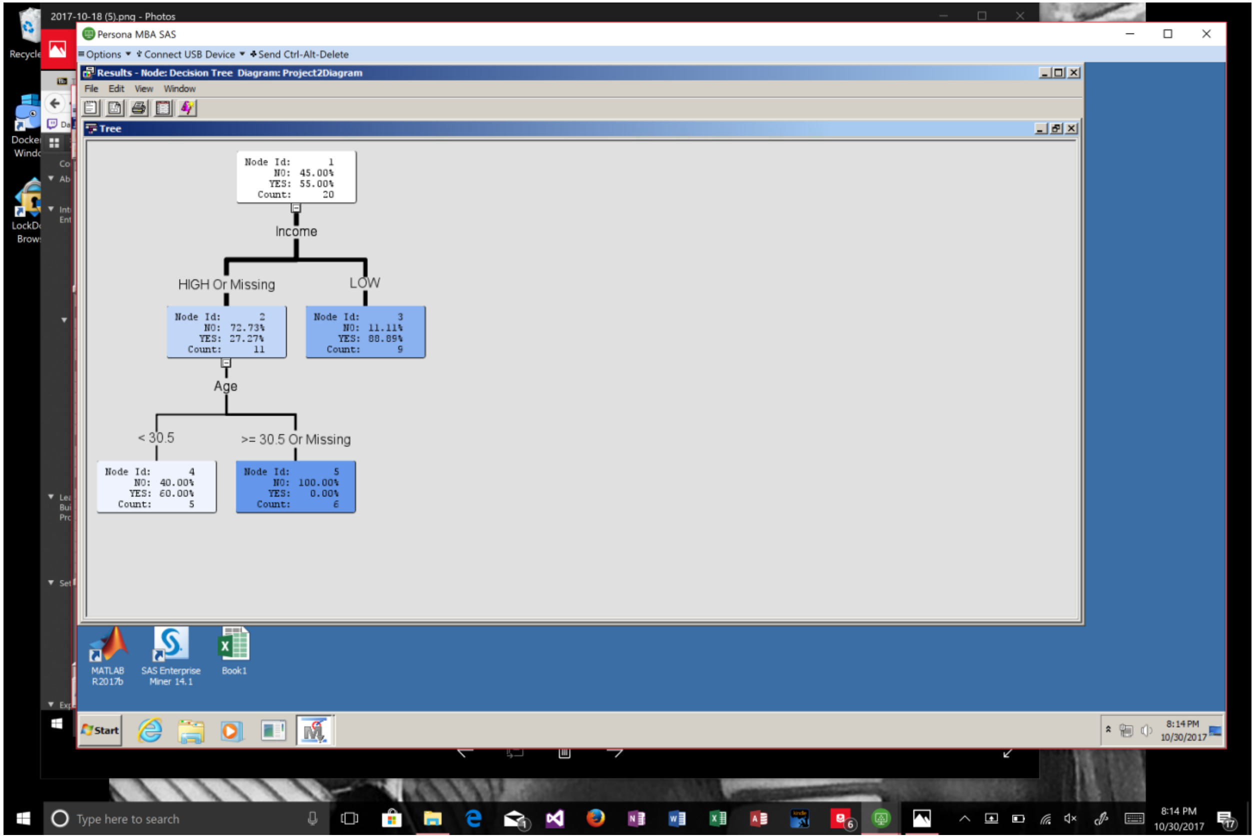
For me the decision tree model is the easiest to understand based on the clearly defined rules and the percentages of outcome occurrences within those rules. It is easy to predict and determine outcomes based on this visual representation of the decision rules or criteria. The other models are a little hard to follow without knowing off hand what everything represents that you’re given. However, this doesn’t mean that this model is the best fit for the data just because it’s the easiest for me or for whatever user to understand. Sometimes (like this time) it is not the best fit or predictor for the data.

This project was challenging due to the fact that I had to know where things were and how to use them without being explicitly stated within a tutorial. It made it easier to understand how exactly the models work and exactly what the model comparison is doing. Google is not very helpful when it comes to SAS, I guess because of the massive amount of capabilities that SAS has. Once I found all the correct nodes and truly engaged in the results windows, I could somewhat establish what I was looking at and it’s significance. Vmware was slower than ever this time but its nothing I’m not expecting at this point, it made running the nodes very irritating but patience is an important virtue they say. Once everything worked I was able to clearly define the rules the decision tree gave me, the regression model's statistics, and the neural network. I was however confused on the confusion matrix, maybe that's why it's a confusion matrix. I couldn't find the location of this table, I found other tables and examined them but couldn't find anything labeled confusion matric.



Selected Model	Predecessor Node	Model Node	Model Description	Target Variable	Target Label	Selection Criterion: Train: Misclassification Rate	Train: Sum of Frequencie
Y	Neural	Neural	Neural Net...	WidgBuy	WidgBuy	0	
	Reg	Reg	Regression	WidgBuy	WidgBuy	0	
	Tree	Tree	Decision Tr...	WidgBuy	WidgBuy	0.15	

```
*------------------------------------------------------------*
User:                 DTQUIR01
Date:                 October 30, 2017
Time:                 20:09:25
*------------------------------------------------------------*
* Training Output
*------------------------------------------------------------*




Variable Summary

          Measurement    Frequency
 Role        Level         Count
```



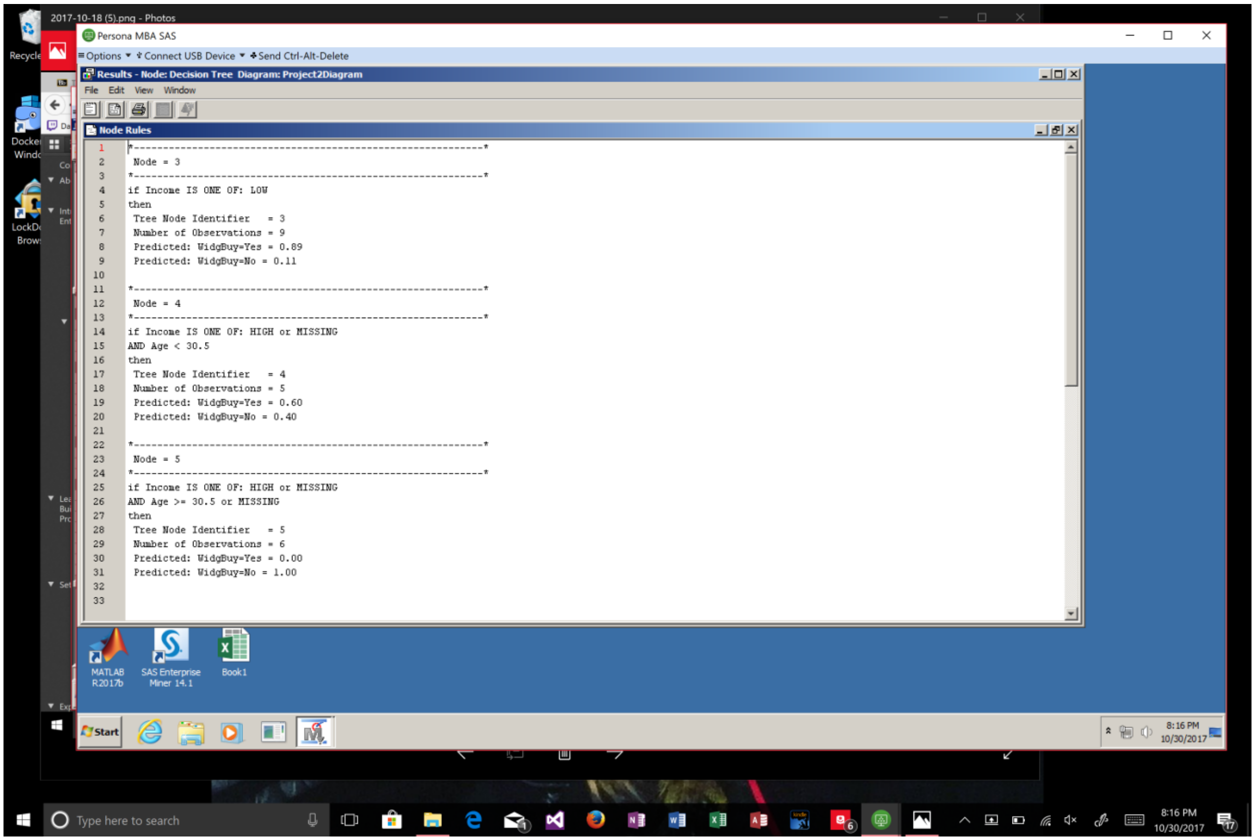
```
Node Id:      1
    NO:  45.00%
   YES:  55.00%
 Count:      20
```

```
Node Id:      2
    NO:  72.73%
   YES:  27.27%
 Count:      11
```

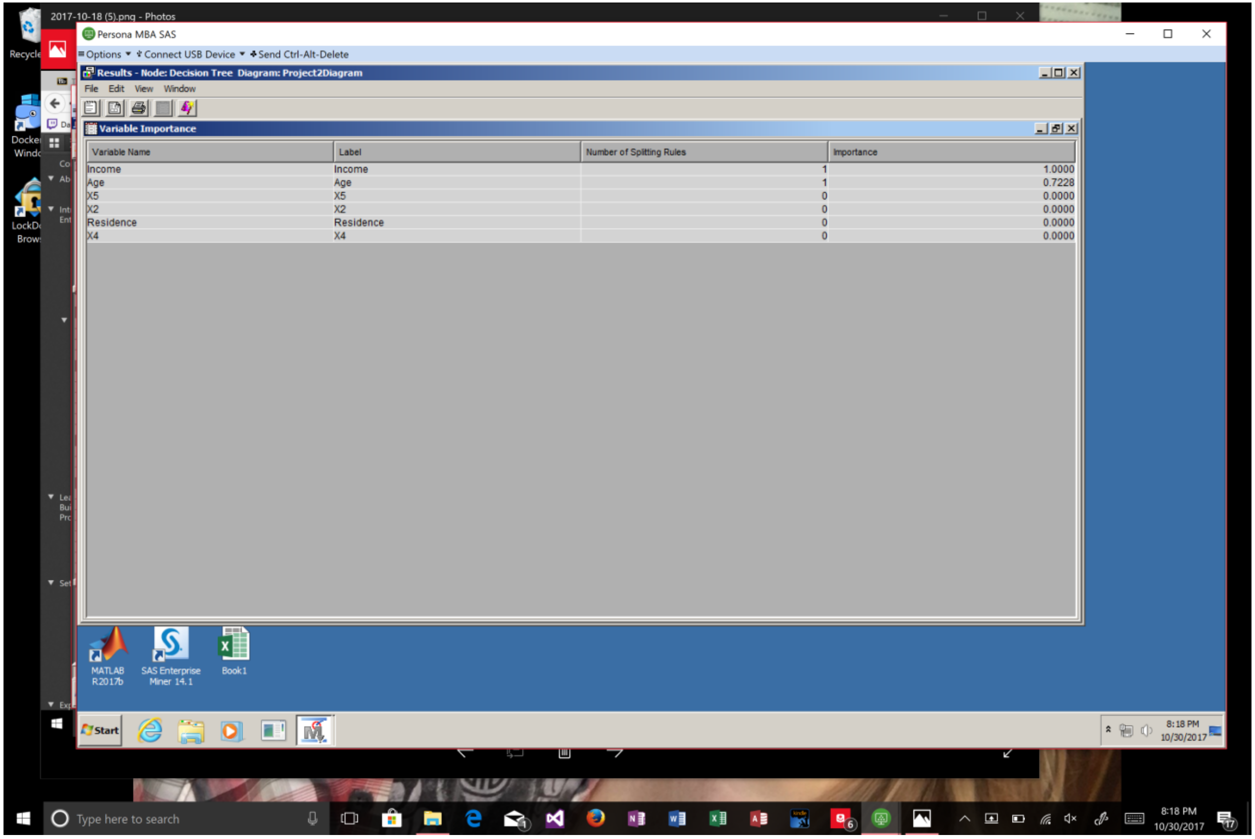
```
Node Id:      3
    NO:  11.11%
   YES:  88.89%
 Count:       9
```

```
Node Id:      4
    NO:  40.00%
   YES:  60.00%
 Count:       5
```

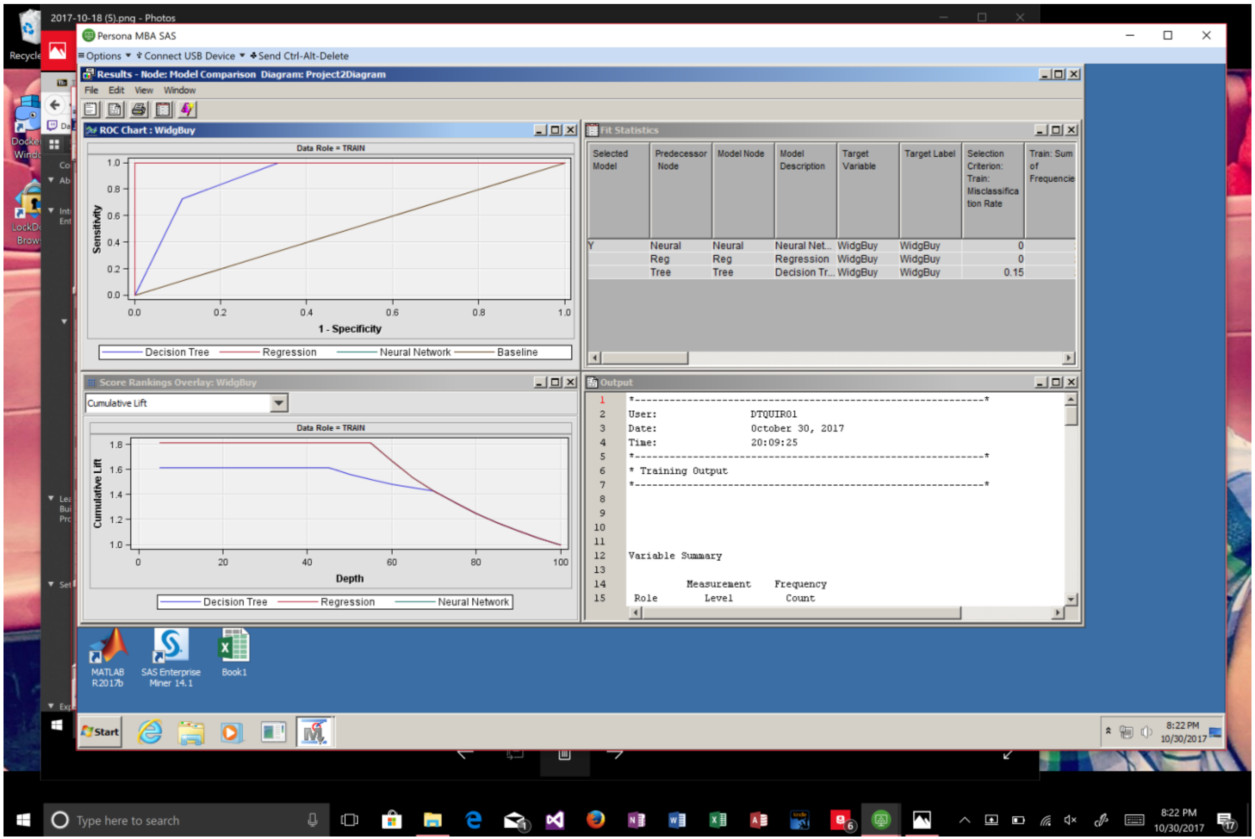
```
Node Id:      5
    NO: 100.00%
   YES:   0.00%
 Count:       6
```



```
*------------------------------------------------------------*
 Node = 3
*------------------------------------------------------------*
if Income IS ONE OF: LOW
then
 Tree Node Identifier   = 3
 Number of Observations = 9
 Predicted: WidgBuy=Yes = 0.89
 Predicted: WidgBuy=No = 0.11

*------------------------------------------------------------*
 Node = 4
*------------------------------------------------------------*
if Income IS ONE OF: HIGH or MISSING
AND Age < 30.5
then
 Tree Node Identifier   = 4
 Number of Observations = 5
 Predicted: WidgBuy=Yes = 0.60
 Predicted: WidgBuy=No = 0.40

*------------------------------------------------------------*
 Node = 5
*------------------------------------------------------------*
if Income IS ONE OF: HIGH or MISSING
AND Age >= 30.5 or MISSING
then
 Tree Node Identifier   = 5
 Number of Observations = 6
 Predicted: WidgBuy=Yes = 0.00
 Predicted: WidgBuy=No = 1.00
```



Variable Name	Label	Number of Splitting Rules	Importance
Income	Income	1	1.0000
Age	Age	1	0.7228
X5	X5	0	0.0000
X2	X2	0	0.0000
Residence	Residence	0	0.0000
X4	X4	0	0.0000

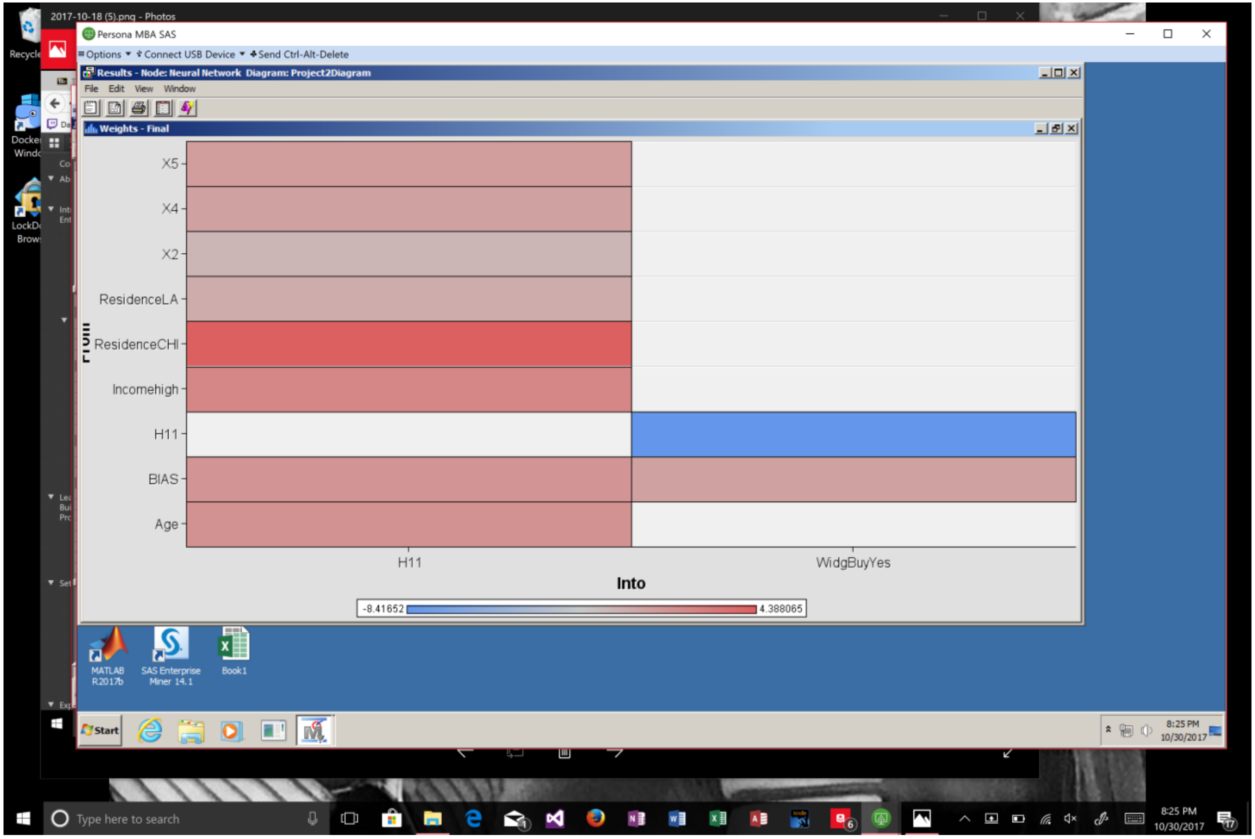


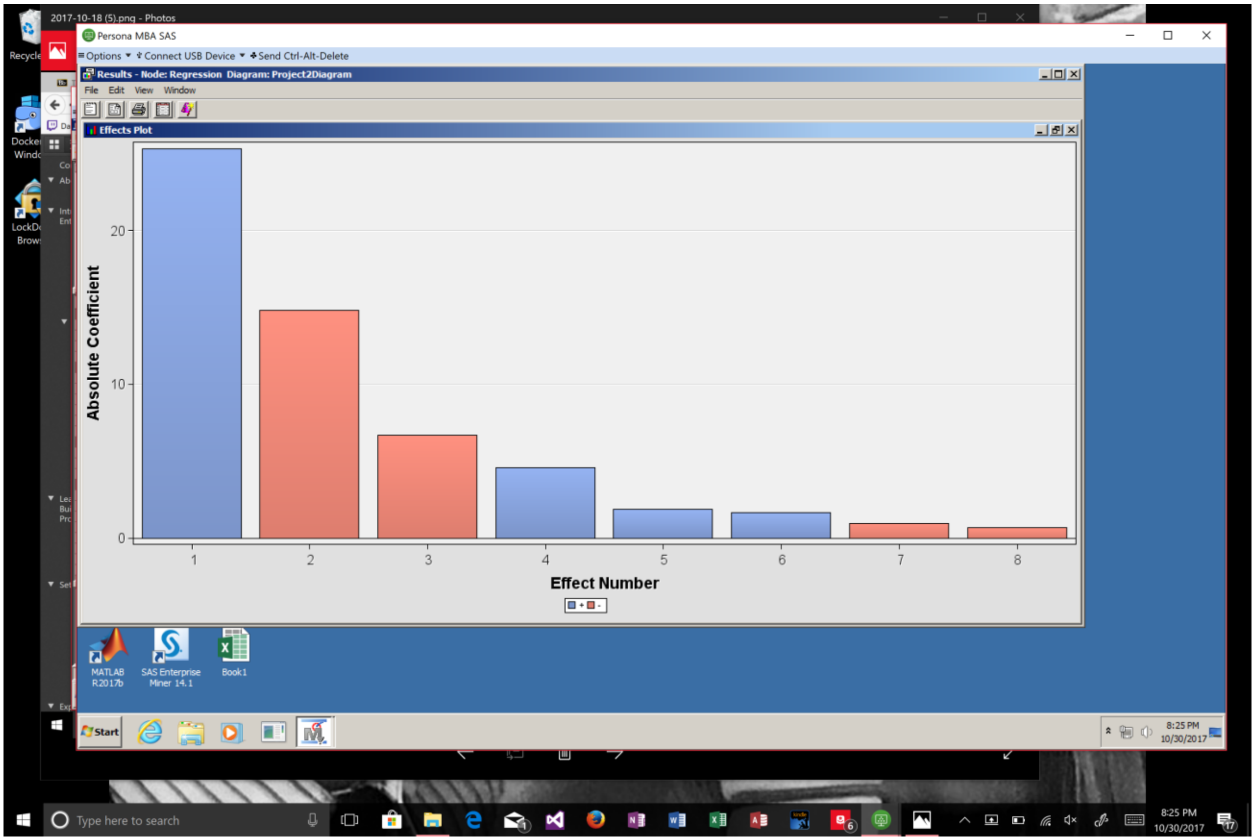
Fit Statistics

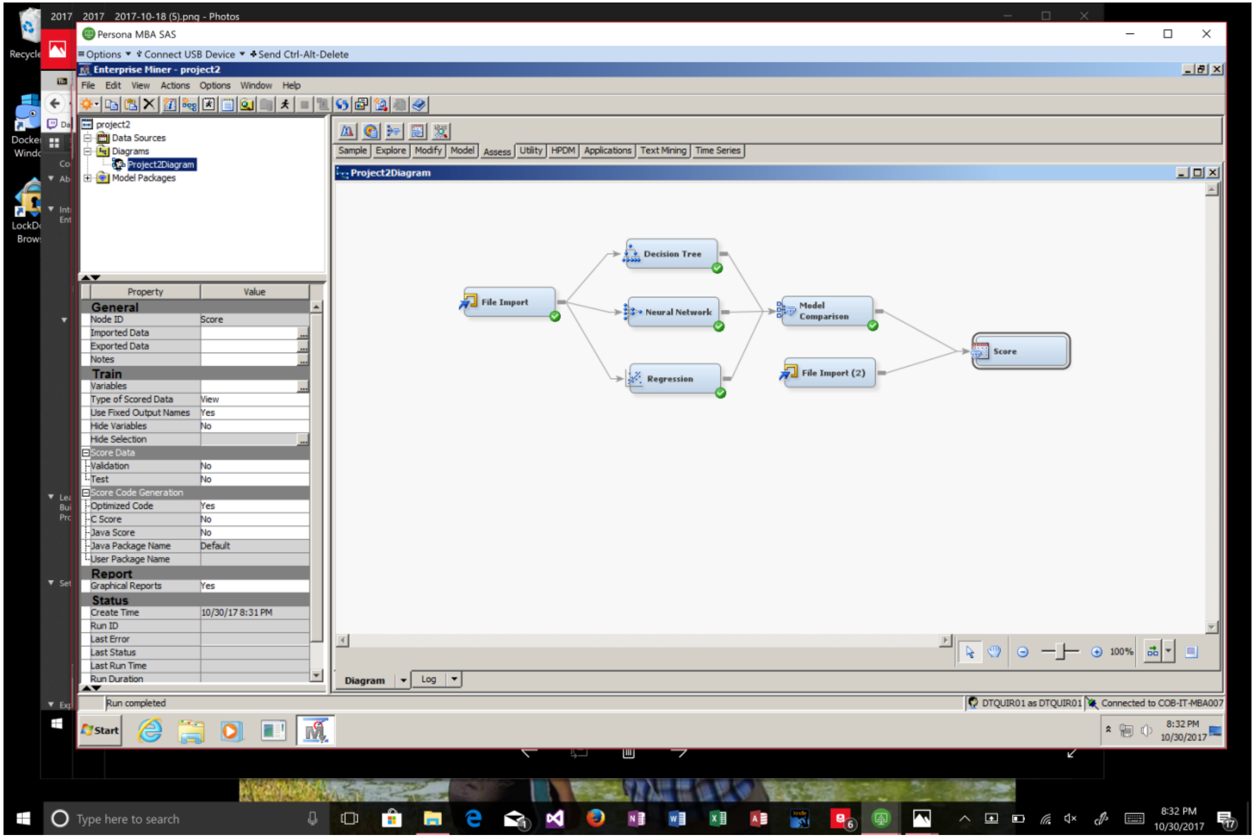
Selected Model	Predecessor Node	Model Node	Model Description	Target Variable	Target Label	Selection Criterion: Train: Misclassification Rate	Train: Sum of Frequencie
Y	Neural	Neural	Neural Net...	WidgBuy	WidgBuy	0	
	Reg	Reg	Regression	WidgBuy	WidgBuy	0	
	Tree	Tree	Decision Tr...	WidgBuy	WidgBuy	0.15	

Output

```
*------------------------------------------------------------*
User:                 DTQUIR01
Date:                 October 30, 2017
Time:                 20:09:25
*------------------------------------------------------------*
* Training Output
*------------------------------------------------------------*




Variable Summary

          Measurement    Frequency
 Role        Level         Count
```







Property	Value
General	
Node ID	Score
Imported Data	
Exported Data	
Notes	
Train	
Variables	
Type of Scored Data	View
Use Fixed Output Names	Yes
Hide Variables	No
Hide Selection	
Score Data	
Validation	No
Test	No
Score Code Generation	
Optimized Code	Yes
C Score	No
Java Score	No
Java Package Name	Default
User Package Name	
Report	
Graphical Reports	Yes
Status	
Create Time	10/30/17 8:31 PM
Run ID	
Last Error	
Last Status	
Last Run Time	
Run Duration	

Run completed

DTQUIR01 as DTQUIR01 Connected to COB-IT-MBA007