Nga Nguyen

Project 2

Due: 10/31/17

For this project we were given two data sets to analyze, a set to train and build the three models and a set used to determine whether or not someone was a widget buyer. The 3 models that were run and compared were the neural network, decision tree and logistical regression. The training set depicted the population that we are trying to depict the behavior of. In this case we are trying to model if someone belonging to a specific population is or is not a widget buyer. While analyzing the data set, I saw that there are a total of 20 records in the test set. The test set contained 9 people who did not purchase widgets and 11 people who did purchase widgets. This means that they have near perfect entropy because this is an almost perfect split of 50% for widget buyers and non-widget buyers. In order to classify customers and non-customers for widgets we have to find the cut-off point of 0.5. The 0.5 cut-off point is the point in which a person is likely to purchase a widget. The cutoff point will be used to determine the sensitivity and the accuracy of the models. The lower the cutoff point, the lower the models will classify things as a non-widget buyer and the higher the cutoff point, the better the model will classify things as a non-widget buyer.

As see in the image below, this is a confusion matrix pulled from the output window of the model comparison mode. The confusion matrix allows us to compare the training data to each of the models. True Positive and True Negative are the variables that have been classified correctly by the training set. A False Negative is when the test set misidentifies something as negative. A false positive is when the test set misidentifies something as positive when it is actually negative. According to the model comparison that I ran, the only model to misclassify any data was the decision tree model, which had 3 false positives. The rules given for the decision tree are not accurate. It classified 3 people as widget buyers when in fact they were non-widget buyers.

Event Classification Table
Model Selection based on Train: Misclassification Rate (_MISC_)

Model Node	Model Description	Data Role	Target	Target Label	False Negative	True Negative	False Positive	True Positive
Tree	Decision Tree	TRAIN	WidgBuy	WidgBuy	0	6	3	11
Neural	Neural Network	TRAIN	WidgBuy	WidgBuy	0	9	0	11
Reg	Regression	TRAIN	WidgBuy	WidgBuy	0	9	0	11

The Lift charts and the ROC charts allow us to compare how well each of the models work. The ROC charts are used to see how accurate the graphs are at specific cutoff points. Ideally the results you want to see will be in the far top left-hand corner of the graph. In part d, is my ROC chart for each model. As you can see, 2 lines in the graph intercept one another. My Lift chart is also pictured below. Lift charts show how likely it is for someone to be a widget buyer. The cutoff point can be entered here so that you can analyze each of your models. The regression model and the neural network contained the same results for all of the thresholds, and the neural network is below the line for the regression model. The decision tree performed the worse out of all three models and this is seen in the ROC chart because the line for it is to the right of the other two models.

The decision tree is pictured in part b. It shows each of the clearly defined rules and the probabilities that are associated with each of those rules. The rules are as follows:

- If Income is Low, then person is Not a widget buyer
- If Income is High and Age is less than 30.5, the person is a widget buyer
- If Income is High and Age is greater than or equal to 30.5, then person is Not a Widget buyer

The most important variables here are Income followed by a person's age. The rest of the variables listed are considered to be insignificant. This was calculated by using calculating entropy to try to get purity for each of the rules, so that we can get the most ideal results. Ideally we want the number to be close to 0.

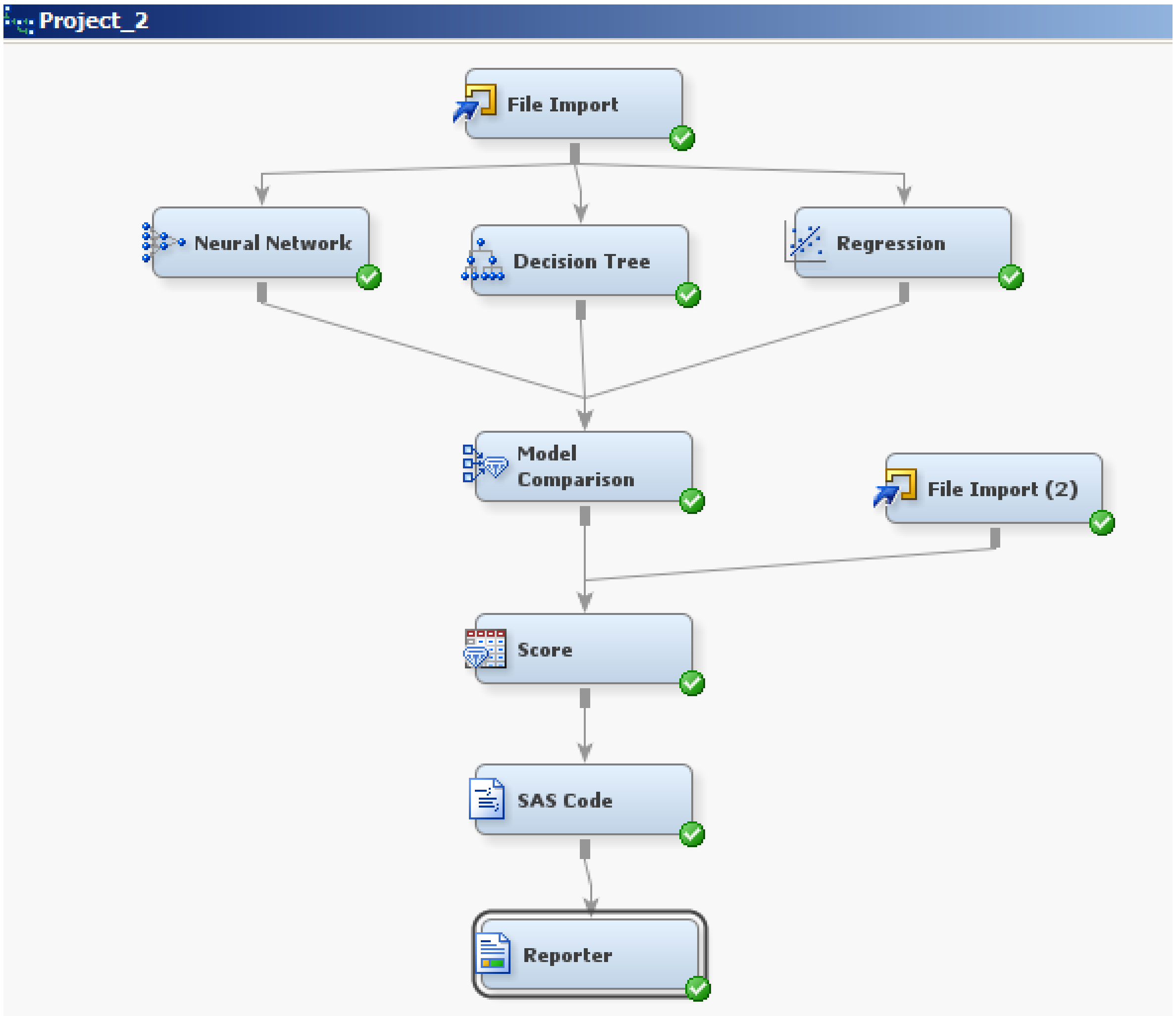
The logistical regression model is pictured in part f. The logistical regression model was one of the models that were found to be most accurate. The greater the absolute value was the more important that a variable was found to be. The more important the variable was, the higher the bar would be found. According to the Logistic regression coefficients, the most important variable were the Residence CHI and the Income High variables.

The variables with the most predictive power are different for each model. In the decision tree, Income and Age of the people were found to have the most predictive power. IN the logistic regression model, Residence and Income were determined to have the most predictive power and these sub-divides into Residence Chi and Income High. For the neural network Residence CHI and Income High are found to have the most predictive power.

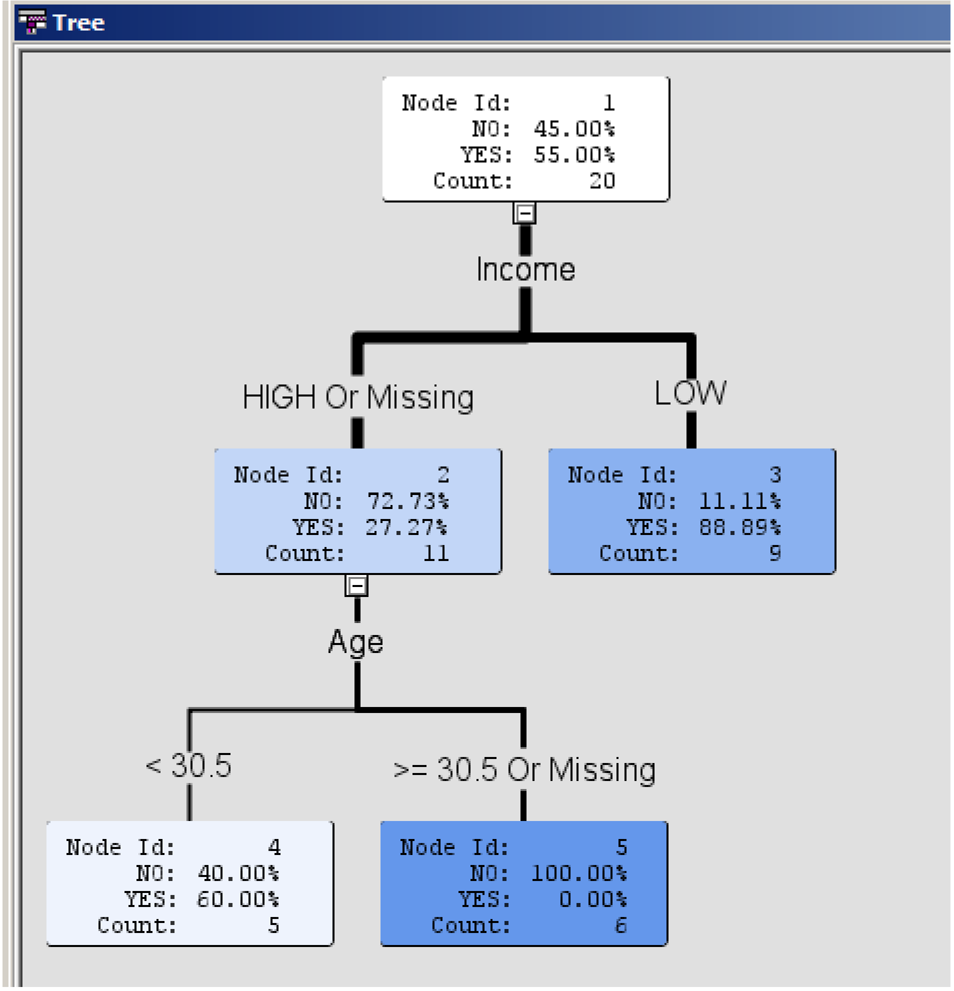
The neural network works using a mathematical function to execute a neuron when the number reaches a certain threshold. In the neural network, Residence CHI was found to be weighted the most out of all the variables and Income High was the second most. There is only one neuron in the neural network, so this is where all the variables have been put. If there were more than one neuron, then this neural network would have been a lot more difficult to interpret.

In the end, the test set contained 9 records that were used to classify people as widget buyers and as non-widget buyers. Out of all three models that we tested, the neural network seemed to be better than the other two models. Doing analysis with SAS code gave us the ability to find the probability from the neural network. It showed how likely someone was to be a widget buyer or a non-widget buyer.

a. Workflow/diagram with all nodes



b. tree diagram generated by decision tree



c. window with rules generated by decision tree

Node Rules

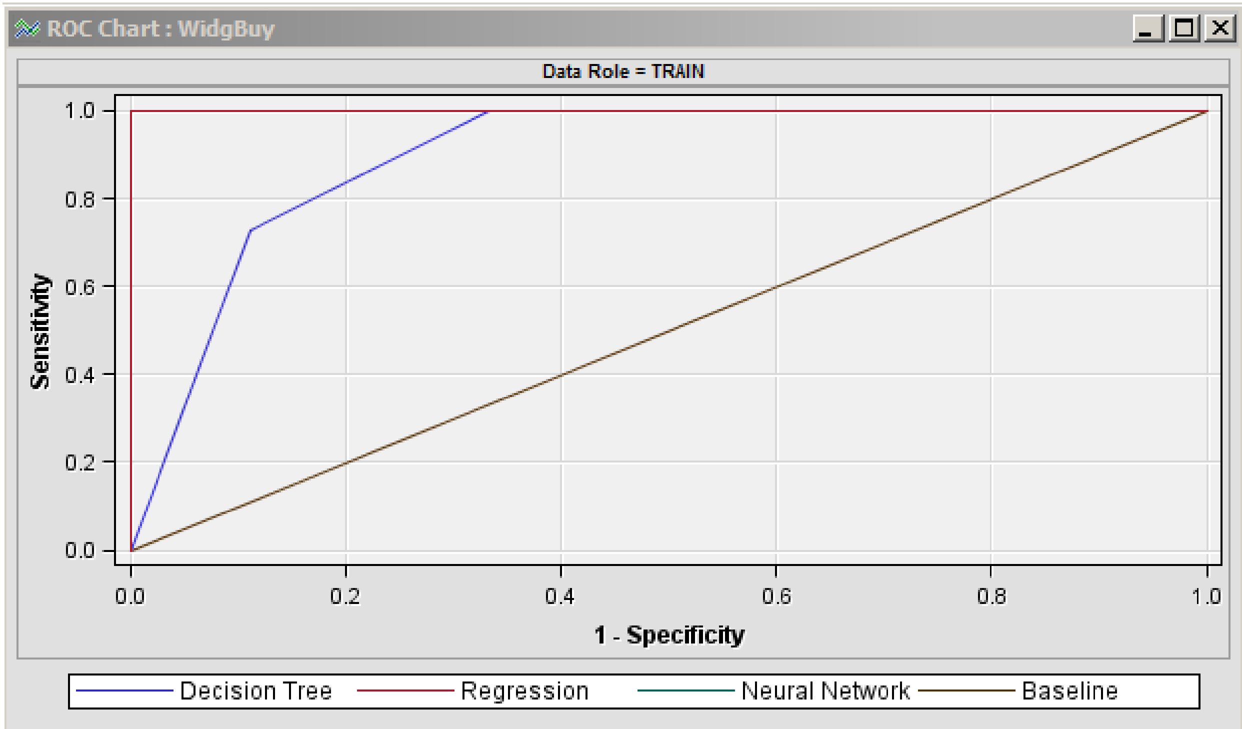
```
*------------------------------------------------------------*
 Node = 3
*------------------------------------------------------------*
if Income IS ONE OF: LOW
then
 Tree Node Identifier   = 3
 Number of Observations = 9
 Predicted: WidgBuy=Yes = 0.89
 Predicted: WidgBuy=No = 0.11

*------------------------------------------------------------*
 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
```

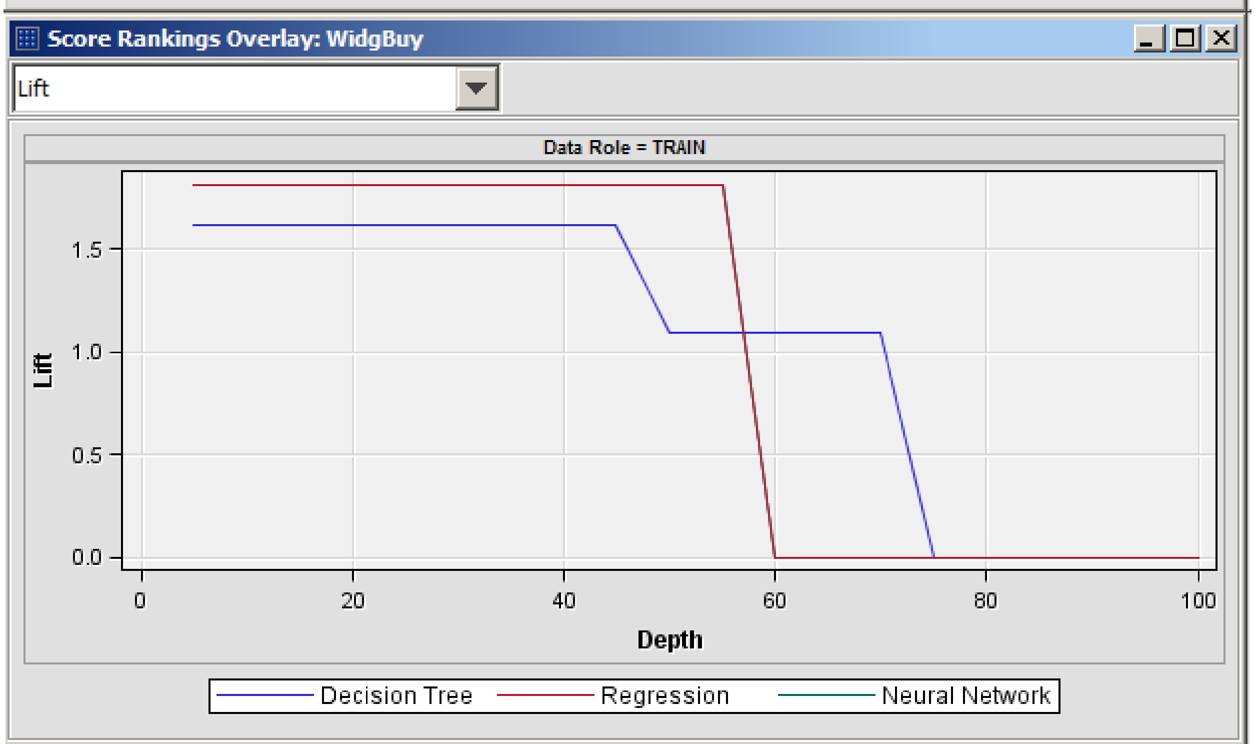
d. table with relative importance of variables used in the decision tree

Variable Importance

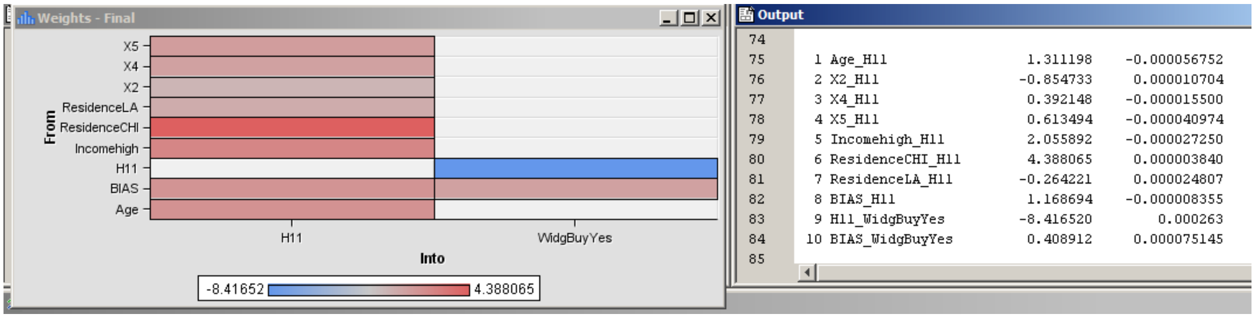
Variable Name	Label	Number of Splitting Rules	Importance
Income	Income	1	1.0000
Age	Age	1	0.7228
X5	X5	0	0.0000
X2	X2	0	0.0000
Residence	Residence	0	0.0000
X4	X4	0	0.0000

e. Lift and ROC charts for the 3 models

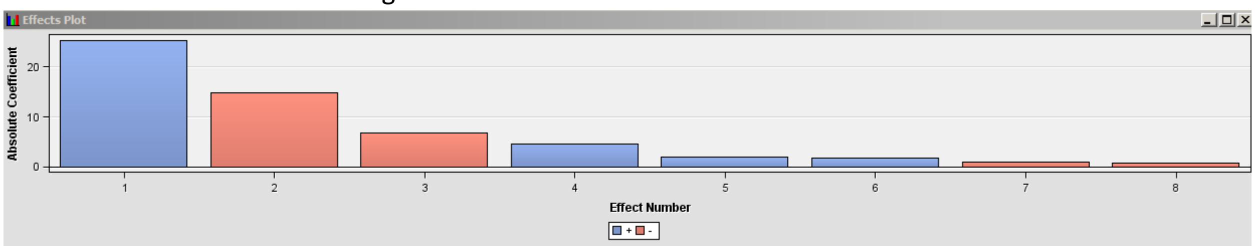




f. Window with the final weights for the neural network



g. Chart with the effects for the regression model



h. output with the probabilities from the SAS Code Node

Output

```
Time:                   14:25:34
*------------------------------------------------------------*
* Training Output
*------------------------------------------------------------*




Variable Summary

                   Measurement    Frequency
Role                  Level         Count

ASSESS              BINARY            1
ASSESS              NOMINAL           1
CLASSIFICATION      NOMINAL           3
INPUT               INTERVAL          4
INPUT               NOMINAL           2
PREDICT             INTERVAL          2
REJECTED            INTERVAL          2
REJECTED            NOMINAL           1
RESIDUAL            INTERVAL          2
SEGMENT             NOMINAL           2
TARGET              BINARY            1





Obs    EM_CLASSIFICATION    EM_EVENTPROBABILITY

 1           YES                  0.99985
 2           YES                  0.99985
 3           YES                  0.99980
 4           NO                   0.00044
 5           NO                   0.00036
 6           NO                   0.00035
 7           NO                   0.00034
 8           NO                   0.00033
 9           NO                   0.00033
```