Dillon Wallace

Dr. Zurada

CIS 445-S1

October 30, 2017

Project 2

A was dataset given to us for analysis which is from a company's client demographics and if that client was a buyer of the company's product, which are widgets. We ran three types of classifying models in order to give the company an appropriate analysis and range of opportunities to determine what types of clients will buy the widget and have the ability to use direct marketing algorithms accordingly. The three types of models we ran was a Neural Network (also called "Deep Learning Algorithms"), Logistical Regression, and decision trees (which are made from the C4.5 Algorithm). The training dataset is one of the most important parts, and as such creates many of the assumptions throughout the models. For example, the training set needs to, not only depict appropriate population of the sales/not sales, but proportionally to determine which variables are important and have a good model that classifies both sales and non-sales. The dataset we were given had 20 records total, which contained nine non-buyers and eleven buyers. Which is a pretty good proportion from the small dataset we were given, almost 50/50. Given that we want to classify both sales and customers that do not buy the widgets we have determine the cut-off point of 0.5, which is the point in which the probability of a customer buying a widget or not. This is a very important part of modeling because it determines the accuracy/sensitivity of the models, if the cut-off point was higher than the models may classify more things as non-buyer than otherwise, the reverse is also true if we lower the cutoff point.

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

This image is the results of a model comparison that compares the models with many different ways, one of which is a confusion matrix, this matrix compares how the training data compares to the actual results of the models classifying algorithms. True Negative or True Positive mean the model classified correctly. A false positive is when the model classified incorrectly, because it misidentified a positive when it is actually negative (non-buyer). The only model that misclassified any data was the Decision Tree model, which means the rules are not fully accurate, while the rest perfectly classified the training data. The Decision Tree classified three records as widget buyers, but according the to company's records, they were not. Two other ways to compare how the models work are called ROC (receiver operating characteristic chart) and Lift charts. ROC charts are used to determine the accuracy of graphs at all cut-off points, you want this chart to be at the top left corner the farthest. In the picture below (d) it

shows the ROC charts for all three models. An interesting thing you can see is that two of the lines intercept. These two models are the Neural Network and Logistical regression. The Lift charts are also shown below. Lift charts are used to show likely-ness of an event (which would be buying a widget in this case). You can use this to cut-off at a certain point in order to get the most of your analysis.

Using the decision tree model (which remember was the only one of the models we used that had error, which were false positives) node rules (seen below picture b), we can see the rules and probabilities. This can give the company a very clear set of guidelines to follow and demographics to cater their marketing to. For example, the most important variable according to this model is the income. If the customers income is low there is an 88% chance of buying a widget, conversely if the income is high (or not collected) then there is a lower probability (27.27%). However, if we branch off further and the clients' income is high (or missing) and age is greater than 30 ½ then the likely hood Is 60% to purchase a widget, but if they are older than this the likelihood is 0%. This tells us that lower income has a higher chance of buying a widget, but a higher income individual who is under thirty has an 60% chance of purchasing. Since there is a greater number of non-rich individuals who have a higher change of purchasing, this model suggests that they are much more likely (also keep in mind that there were three misclassifications, so this model is not perfect). While also looking at variable importance the two that the decision tree considers the most important are income and age. This is calculated by trying to purify the branches (or rules) in order to get perfect results. As we can see this method, for this dataset was not fully accurate, however the model gives you rules that can help the company make decisions.

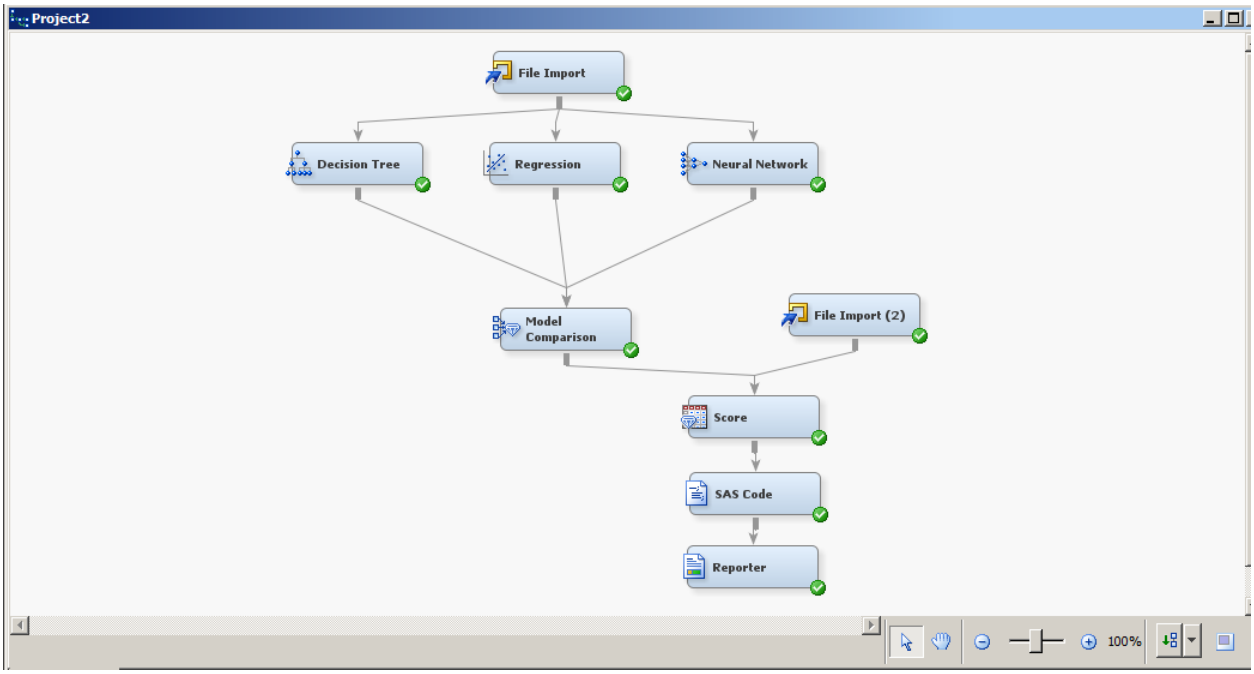
Going on to the logistical regression which encodes its logic into coeffiencts or effects, which are both represented by the graph and the output in photos (f). This regression was one of the models that scored perfectly. According to the rules if you live in residenceCHI the likelihood of purchasing a widget goes down by 14.79% (which is the most predictive in the group), and being high income decreases the possibility even further by 6.67%, this in conjunction, with age (for every year of a persons age the probability goes down by 0.972%; to summarize the older a person is the less likely they are to buy a widget. This model has brought up a new variable the other model did not consider significant, residenceCHI. This model says it is more significant than the decision tree (which did not consider it to have any significance in comparison to the others), which could possibly be why it classifies better than the decision tree for people who live there.

Neural Networks act kind of like how the brains works, it uses mathematical functions in order to fire a neuron (mathematical model that gets inputs and weights them). Looking at the final weights (e) seem to say that residenceCHI is weighted the highest out all of the variables. This neural network only contained one neuron, so all of the variables went into the one neuron. The second highest seems to be incomeHigh which would correspond to the other models. Neurons are hard to interpret especially if there are many nodes, in which case the weights could mean almost anything depending on how the network is setup. Since there is only one neuron and all inputs connect, it is much easier to give meaning to the weights listed above.

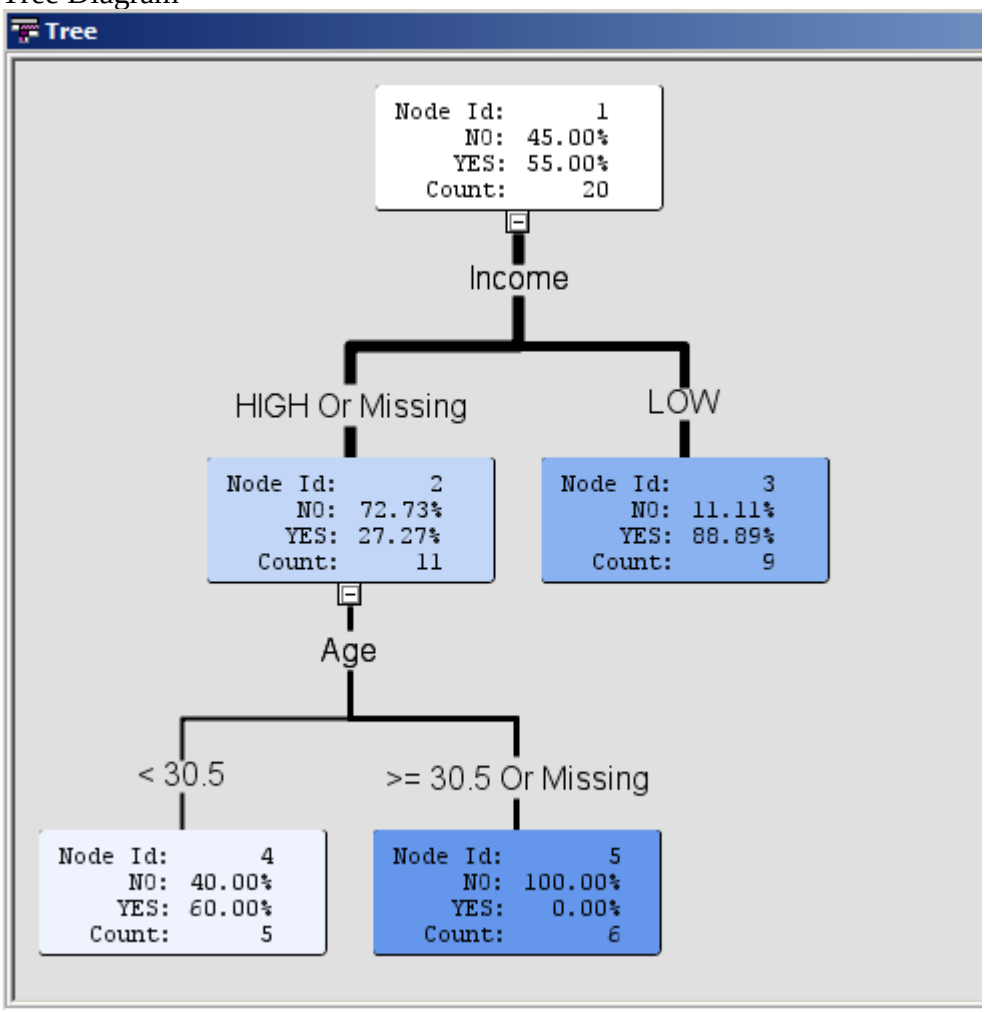
Using the data given to use to classify if the customers would be considered widget buyers or not (which is called the test dataset) there was nine records total. The best model was considered to be the neural network (e), which scored the data and gave these results: three were considered

to be widget buyers, the other six, however were considered to be non-widget buyers. Using SAS code we were able to create the probability given to us from the neural network, it gave us the predication (if the person was a widget buyer or not, and the chance).

Workflow/Diagram



a.) Tree Diagram



Node Rules

```
4   if Income IS ONE OF: LOW
5   then
6    Tree Node Identifier   = 3
7    Number of Observations = 9
8    Predicted: WidgBuy=Yes = 0.89
9    Predicted: WidgBuy=No = 0.11
10
11  *------------------------------------------------------------*
12   Node = 4
13  *------------------------------------------------------------*
14  if Income IS ONE OF: HIGH or MISSING
15  AND Age < 30.5
16  then
17   Tree Node Identifier   = 4
18   Number of Observations = 5
19   Predicted: WidgBuy=Yes = 0.60
20   Predicted: WidgBuy=No = 0.40
21
22  *------------------------------------------------------------*
23   Node = 5
24  *------------------------------------------------------------*
25  if Income IS ONE OF: HIGH or MISSING
26  AND Age >= 30.5 or MISSING
27  then
28   Tree Node Identifier   = 5
29   Number of Observations = 6
30   Predicted: WidgBuy=Yes = 0.00
31   Predicted: WidgBuy=No = 1.00
```

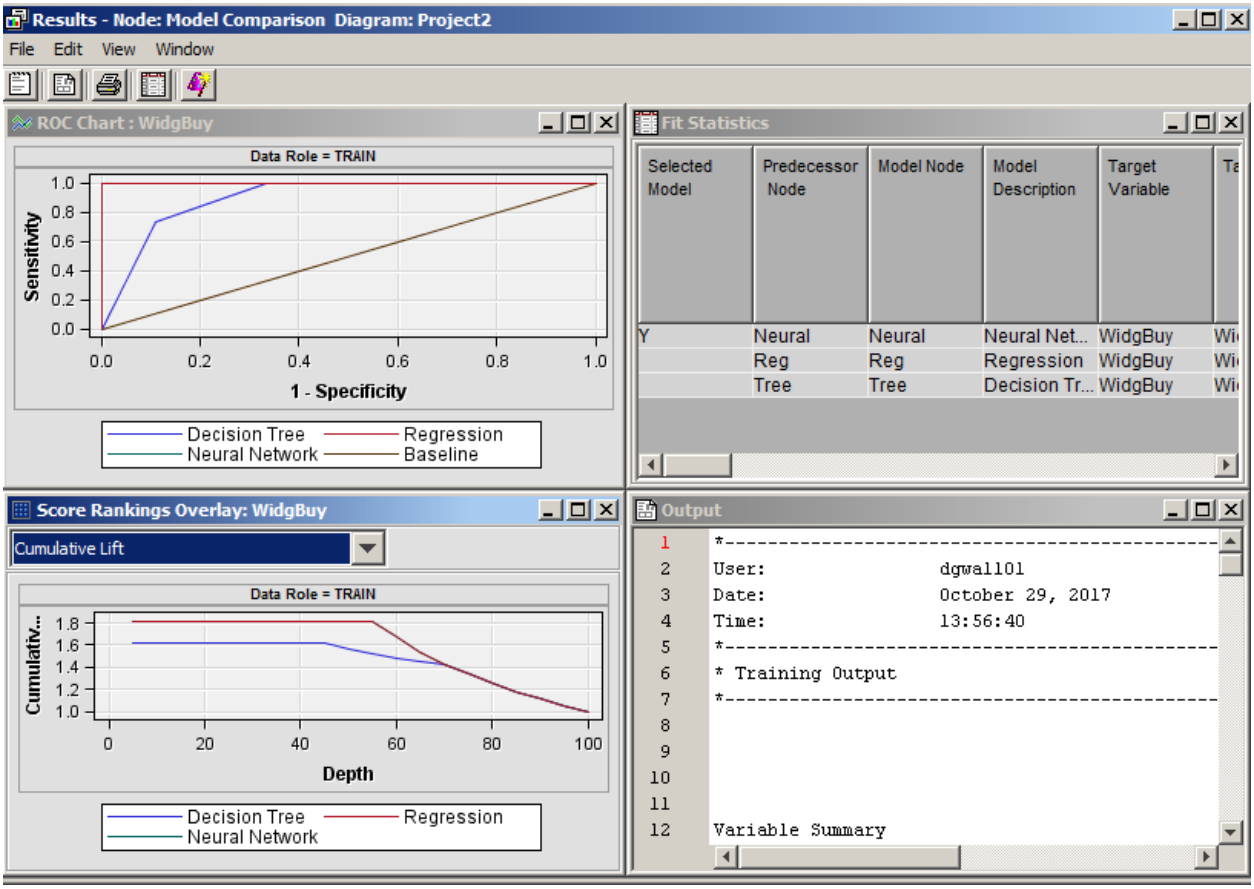
b.) Tree Rules

c.) Table w/ relative importance/entropy of 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

d.) Life and ROC for all models



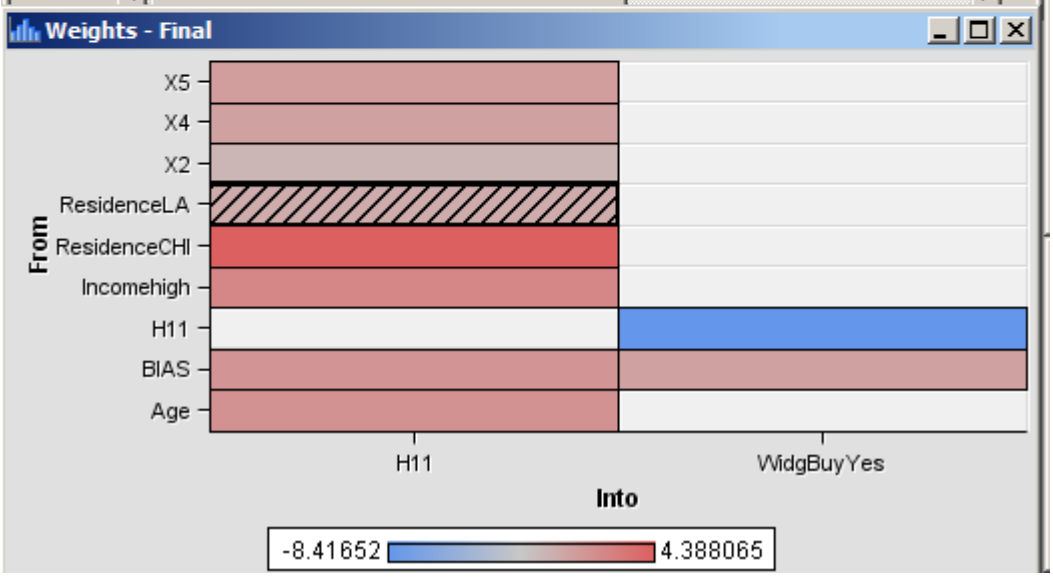
e.) Final weights of NN

Results - Node: Neural Network Diagram: Project2

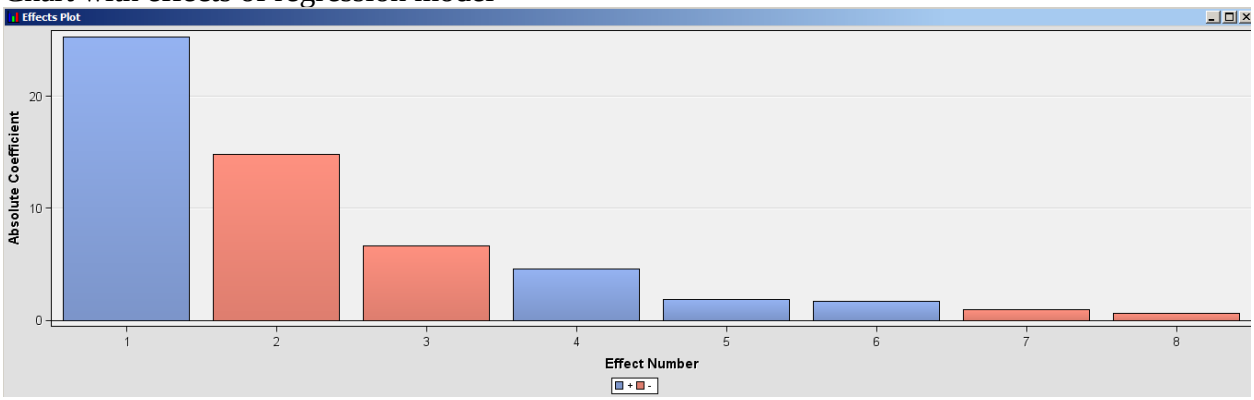
File Edit View Window

Output

75	1	Age_H11	1.311198	-0.000056752
76	2	X2_H11	-0.854733	0.000010704
77	3	X4_H11	0.392148	-0.000015500
78	4	X5_H11	0.613494	-0.000040974
79	5	Incomehigh_H11	2.055892	-0.000027250
80	6	ResidenceCHI_H11	4.388065	0.000003840
81	7	ResidenceLA_H11	-0.264221	0.000024807
82	8	BIAS_H11	1.168694	-0.000008355
83	9	H11_WidgBuyYes	-8.416520	0.000263
84	10	BIAS_WidgBuyYes	0.408912	0.000075145
85				



f.) Chart with effects of regression model



Analysis of Maximum Likelihood Estimates

Parameter		DF	Estimate	Standard Error	Wald Chi-Square	Pr > ChiSq	Standardized Estimate	Exp(Est)
Intercept		1	25.3156	155.0	0.03	0.8703		999.000
Age		1	-0.9725	5.8043	0.03	0.8669	-3.7018	0.378
Income	high	1	-6.6741	26.0980	0.07	0.7982		0.001
Residence	CHI	1	-14.7970	49.4991	0.09	0.7650		0.000
Residence	LA	1	1.8557	42.8673	0.00	0.9655		6.396
X2		1	4.5656	38.9956	0.01	0.9068	1.7317	96.125
X4		1	1.6689	101.9	0.00	0.9869	0.2293	5.306
X5		1	-0.6534	10.7500	0.00	0.9515	-0.8021	0.520

g.) Output w/ probabilities of SAS output

The First 9 Observations

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