

Project 3

KIMBERLY ROETEN
CIS 445-01: DATA MINING

For this project, the nodes that were used were an HMEQ train data set from the SAS library SAMPSIO, a data partition, an impute, which then splits off into 2 paths where transform variables are used and where they are not, so in this case the first path undergoes transform variables and interactive binning nodes, and second path undergoes a variable selection node. These two nodes connect to 3 neural network nodes each, one with 1 neuron in the hidden layer, one with 3 neurons in the hidden layer which is the default, and the last with 5 neurons in the hidden layer. This goes for both the neuron networks in the transform variables path and the variable selection path. All 6 of these neuron network nodes connect to a model comparison node, then the path concludes with a reporter node. No score or SAS code node are used in this project.

Confusion matrices in this flow process diagram that was built for this project exist for all of six of the neural network model nodes. There was no SAS Code node required for this project, let alone existent in the instructions, so this particular node is not included, which means a confusion matrix cannot be made using this node to see the distribution of both the flow that uses a “cutoff node” to change the probability cutoff and the other flow that drops the cutoff node and keeps probability cutoff as 0.5. But each of the 6 model nodes has generated in their own ways the probability that a customer was a widget buyer or not. As a refresher, the confusion matrices in this case are classified as the event classification tables in the output results, where the operating cut-off points affect the correct and incorrect classification rates and the number of false positive and false negative errors. So in other words, the confusion matrices show the number of correct and incorrect predictions made by the classification model compared to the actual outcomes, in this case target values, in the data given. What all 6 confusion matrices share in common is that they display matrices for 2 main classes, positive and negative. The positive value is the proportion of positive cases that were correctly identified, in contrast to the negative value that includes the proportion of negative cases that were correctly identified.

The first confusion matrix is the neural network for 1 neuron in the hidden layer in the transform variables path. Both train and validate data roles are present in the snippet and all positive and negative proportions produced results. According to the data, there were more true cases that were correctly identified than there were false cases, which is what we want, however for the train data role the amount of negative results was greater than they were for validate, and the amount of positive results was less for data role than they were for validate.

Second comes the neural network confusion matrix for 3 neurons in the hidden layer in the transform variables path, the data roles are also classified as train and validate, and all positive and negative proportions produced results as well. According to the data, there were more true cases that were correctly identified than there were false cases, and the same thing happened in terms of the amounts of positive and negative results and where they occurred most often. Since there are 3 neurons in the hidden layer in this model, more results overall are produced.

Third comes the neural network confusion matrix for 5 neurons in the hidden layer in the transform variables path, the data roles are also classified as train and validate, and all positive and negative proportions produced results as well. According to the data, there were more true

cases that were correctly identified than there were false cases, and the same thing happened again in terms of the amounts of positive and negative results and where they occurred most often. However, there less false proportions that produced for the train role than there were for the neural network with 3 neurons, which is surprising to me, considering this neural network node has 5 neurons. But there are more true proportions that are produced for the train role, which was expected.

Fourth comes the neural network confusion matrix for 1 neuron in the hidden layer in the variable selection path, the data roles are also classified as train and validate, and with variable selection more results overall are producing, which makes sense considering variable selection performs variable and attribute reduction unlike transform variables which transform the variables and create new ones, so the same amounts of initial variables still exist. More true cases are correctly identified than false cases, but there are less positive cases produced in this neural network than there were in the neural network with transform variables.

Fifth comes the neural network confusion matrix for 3 neurons in the hidden layer in the variable selection path, the data roles are also classified as train and validate, and overall less results were produced in this neural network, which is surprising considering there are more neurons in the hidden layer this time. However, something I found intriguing about these results compared to the results of the fourth neural network was that the amount of true negative and false positive results in both the train data roles were the same.

Lastly is where the neural network confusion matrix comes in, where 5 neurons in the hidden layer in the variable selection path The data roles were train and validate in this case, just like the previous 5 models. Less negative results, and this goes for both train and validate roles, were produced than positive, which is again what we want to see.

In terms of the ROC and lift charts of all six models, they measure the effectiveness of the classification models calculated as the ratios between the results obtained with and without the models. So in contrast to confusion matrices which evaluate models on the whole population, these charts evaluate model performance in portions of the population. They show how much more likely we are to receive positive responses than if we were to contact a random sample of responses from the population as a whole.

So according to the lift charts, by contacting anywhere from about 0-20% of the customers based on the predictive model, we will reach close to 3-5 times as many respondents, as if we use no model. By contacting anywhere from about 80-100% of the customers on the predictive model, we will reach around 0-1 times as many respondents, as if we use no model.

For this application, the neural network model that would be the best model to classify future customers is the one with 5 neurons in the hidden layer and on the variable selection path. The reason for this is the less errors in the confusion matrices produced, the better.

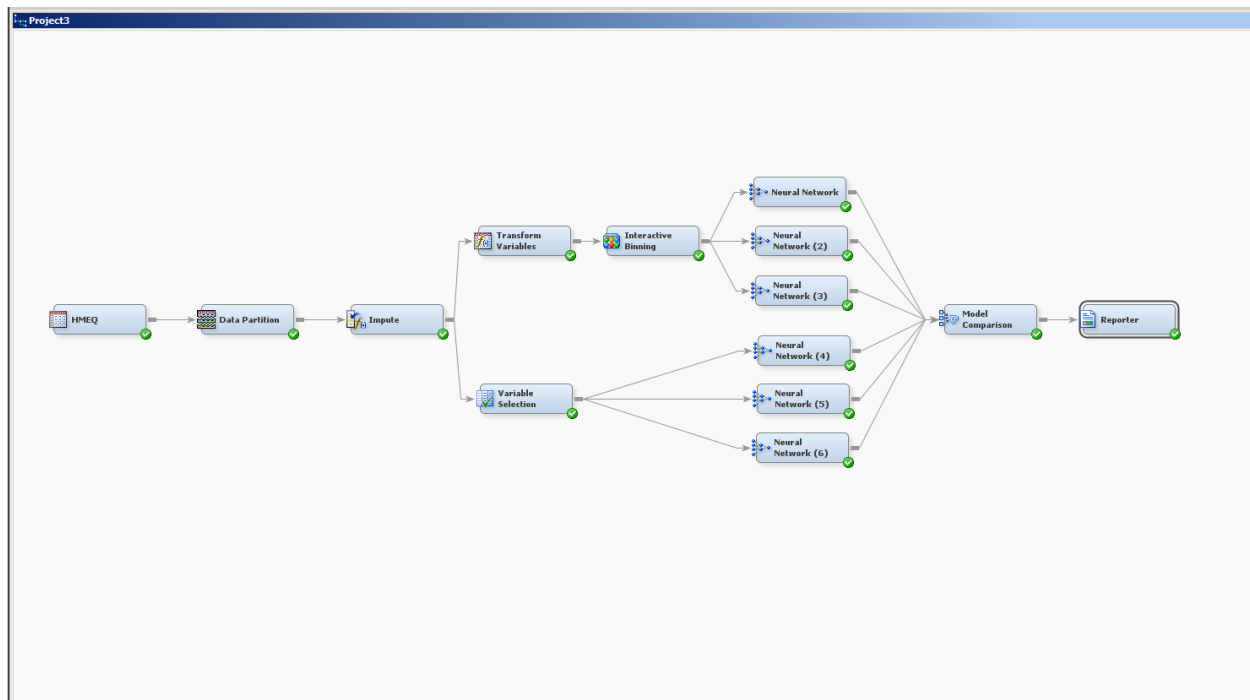
According to this workflow diagram, using variable selection is the better solution, even when the variables that are transformed undergo interactive binning because the amounts of variables are reduced, so the less errors we have to keep track of, the node helps eliminate the

information that is not needed, so in this case variable selection does help in achieving better classification accuracy rates, because the less there are, the more accurate the results of the model.

After thoroughly examining and interpreting the final weights of the best neural network model, it is safe to conclude that according to the neural network residence in terms of H11 in the BAD target variable was ranked the variable with the most weight.

After analyzing the best neural network model with respect to the overall correct classification accuracy rates, correct classification accuracy of bad and good loans, as well as the false positive and false negative classification errors, this model tends to classify more bad loans than good loans, which is understandable in a way because in this case, it has an option to expand its reasoning in terms of a decision tree, the more if then statements we know the better.

1A. Workflow/diagram with all nodes used



1Bi. Neural Network Confusion Matrix (1 neuron in hidden layer, transform variables path)

Results - Node: Neural Network Diagram: Project3				
File Edit View Window				
Output				
425	Event Classification Table			
426	Data Role=TRAIN Target=BAD Target Label=' '			
427	False	True	False	True
428	Negative	Negative	Positive	Positive
429	439	2294	91	155
430	Data Role=VALIDATE Target=BAD Target Label=' '			
431	False	True	False	True
432	Negative	Negative	Positive	Positive
433	423	2275	111	172
434				
435				
436				
437				
438				
439				
440				
441				
442				

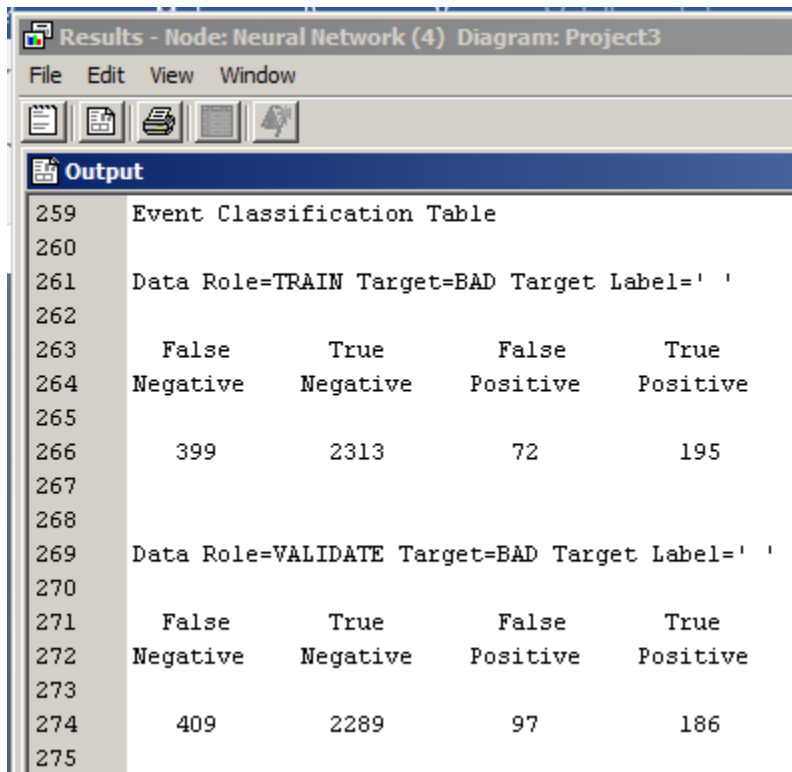
1Bii. Neural Network Confusion Matrix (3 neurons in hidden layer, transform variables path)

Results - Node: Neural Network (2) Diagram: Project3				
File Edit View Window				
Output				
735	Event Classification Table			
736				
737	Data Role=TRAIN Target=BAD Target Label=' '			
738				
739	False	True	False	True
740	Negative	Negative	Positive	Positive
741				
742	352	2283	102	242
743				
744				
745	Data Role=VALIDATE Target=BAD Target Label=' '			
746				
747	False	True	False	True
748	Negative	Negative	Positive	Positive
749				
750	310	2268	118	285
751				

1Biii. Neural Network Confusion Matrix (5 neurons in hidden layer, transform variables path)

Results - Node: Neural Network (3) Diagram: Project3				
File Edit View Window				
Output				
1015	Event Classification Table			
1016				
1017	Data Role=TRAIN Target=BAD Target Label=' '			
1018				
1019	False	True	False	True
1020	Negative	Negative	Positive	Positive
1021				
1022	342	2286	99	252
1023				
1024				
1025	Data Role=VALIDATE Target=BAD Target Label=' '			
1026				
1027	False	True	False	True
1028	Negative	Negative	Positive	Positive
1029				
1030	310	2269	117	285
1031				

1Biv. Neural Network Confusion Matrix (1 neuron in hidden layer, variable selection path)



Results - Node: Neural Network (4) Diagram: Project3

File Edit View Window

Output

259 Event Classification Table

260

261 Data Role=TRAIN Target=BAD Target Label=' '

262

263	False	True	False	True
264	Negative	Negative	Positive	Positive
265				
266	399	2313	72	195
267				

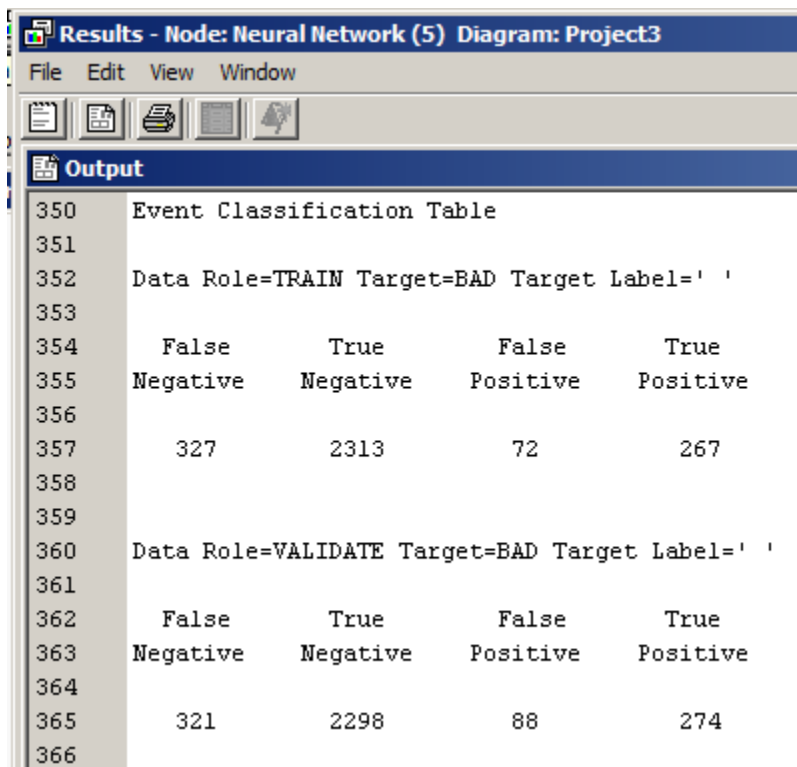
268

269 Data Role=VALIDATE Target=BAD Target Label=' '

270

271	False	True	False	True
272	Negative	Negative	Positive	Positive
273				
274	409	2289	97	186
275				

1Bv. Neural Network Confusion Matrix (3 neurons in hidden later, variable selection path)



Results - Node: Neural Network (5) Diagram: Project3

File Edit View Window

Output

350 Event Classification Table

351

352 Data Role=TRAIN Target=BAD Target Label=' '

353

354	False	True	False	True
355	Negative	Negative	Positive	Positive
356				
357	327	2313	72	267
358				

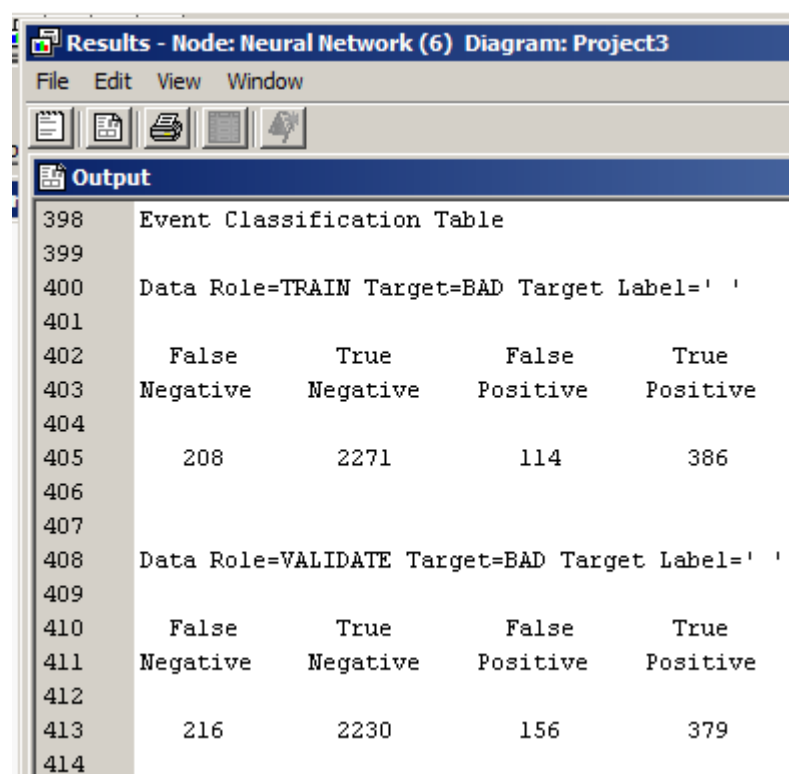
359

360 Data Role=VALIDATE Target=BAD Target Label=' '

361

362	False	True	False	True
363	Negative	Negative	Positive	Positive
364				
365	321	2298	88	274
366				

1Bvi. Neural Network Confusion Matrix (5 neurons in hidden later, variable selection path)



Results - Node: Neural Network (6) Diagram: Project3

File Edit View Window

Output

398 Event Classification Table

399

400 Data Role=TRAIN Target=BAD Target Label=' '

401

	False	True	False	True
	Negative	Negative	Positive	Positive
	208	2271	114	386

405

406

407

408 Data Role=VALIDATE Target=BAD Target Label=' '

409

	False	True	False	True
	Negative	Negative	Positive	Positive
	216	2230	156	379

410

411

412

413

414

1C. Model Comparison ROC and Lift Charts

