In this tutorial you will learn to use the K-NN, Neural Networks and Ensemble models. In addition, you will learn how to deploy the model (score it).

IMPORTANT! For this tutorial you will need two datasets:

- 1 part will be used for the model building
- Another part will be used for scoring the model

The datasets should be identical by the number of variables and their levels.

So, you will begin with SAS EG first to divide the dataset into two parts – one part for model building (the major part of the dataset – 90% or 95%, depending on the size of the original dataset) and one part – to score the data. The second part represents what you will have to do in real life, when you are given a completely new dataset and are asked to predict, say, who is going to buy your product or pay out a loan.

SAS EG: DIVIDING DATASETS

1) Open SAS EG, open a new program, run a libname to create a permanent library:

```
libname DATA678 'C:\MBA678\Course_data';
```

2) Import the Excel telco_churn dataset into SAS EG and save it permanently by changing the library from WORK to DATA678.

3) Create a new variable.

Telco_churn has a total of 7043 observations, and we will divide the dataset into two parts in the following way:

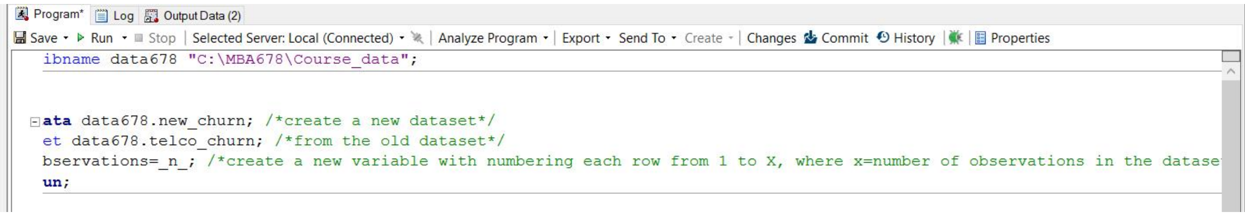
- **6043 rows** to run models on (this file will contain Observations <= 6043)
- **1000** to score the data (this dataset will contain Observations > 6043)

To do that, we need to create a new variable – observations that will simply assign a number to each row from 1 to X, where X = the number of total observations.

Write this program:

```
data data678.new_churn;      /*create a new permanent dataset*/
set data678.telco_churn;     /*from the old permanent dataset*/
observations=_n_;            /*create a new variable with numbering each
                             row from 1 to X, where x=number of
                             observations in the dataset*/
run;
```

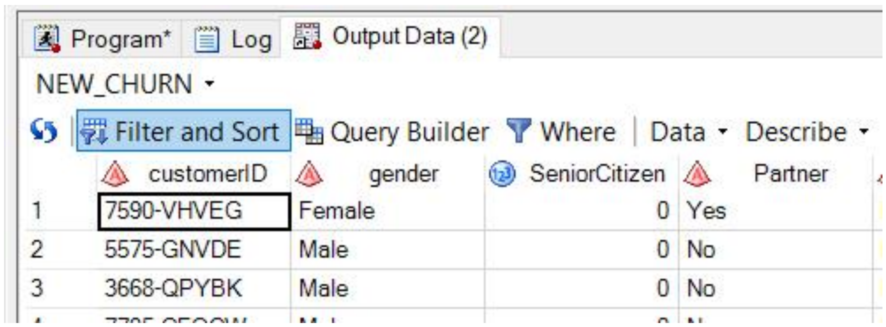
For the next step, we will be using the new file called **new_churn** and will just split it by using the **FILTER and SORT** task.

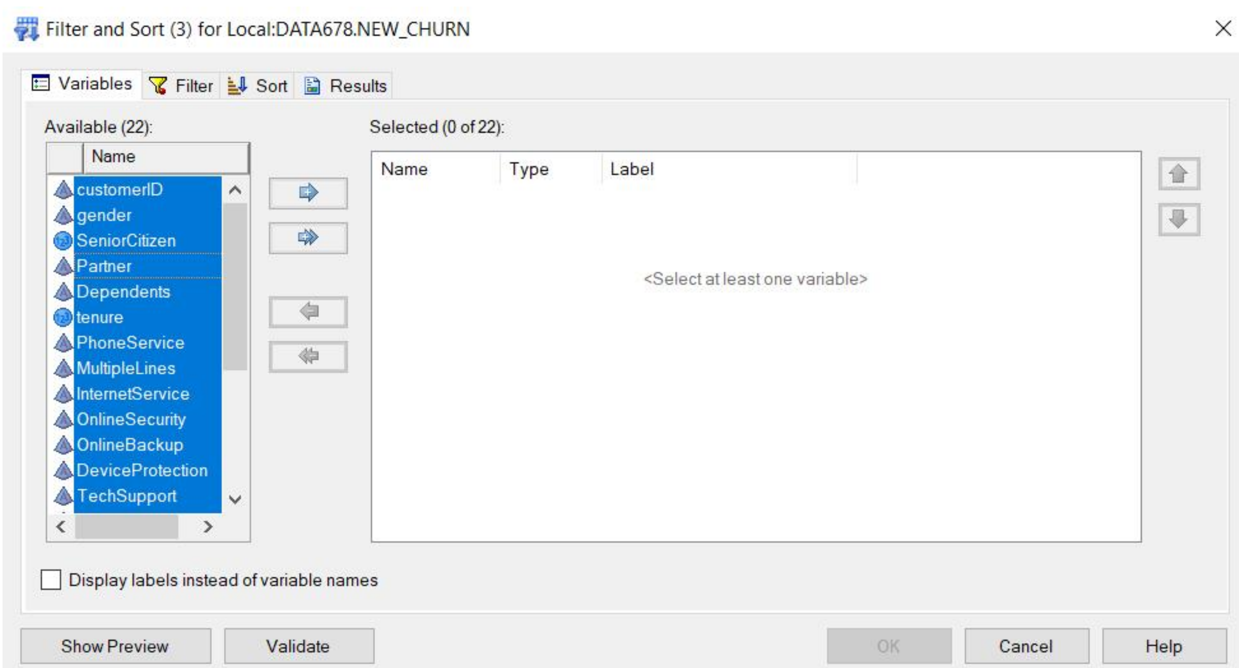


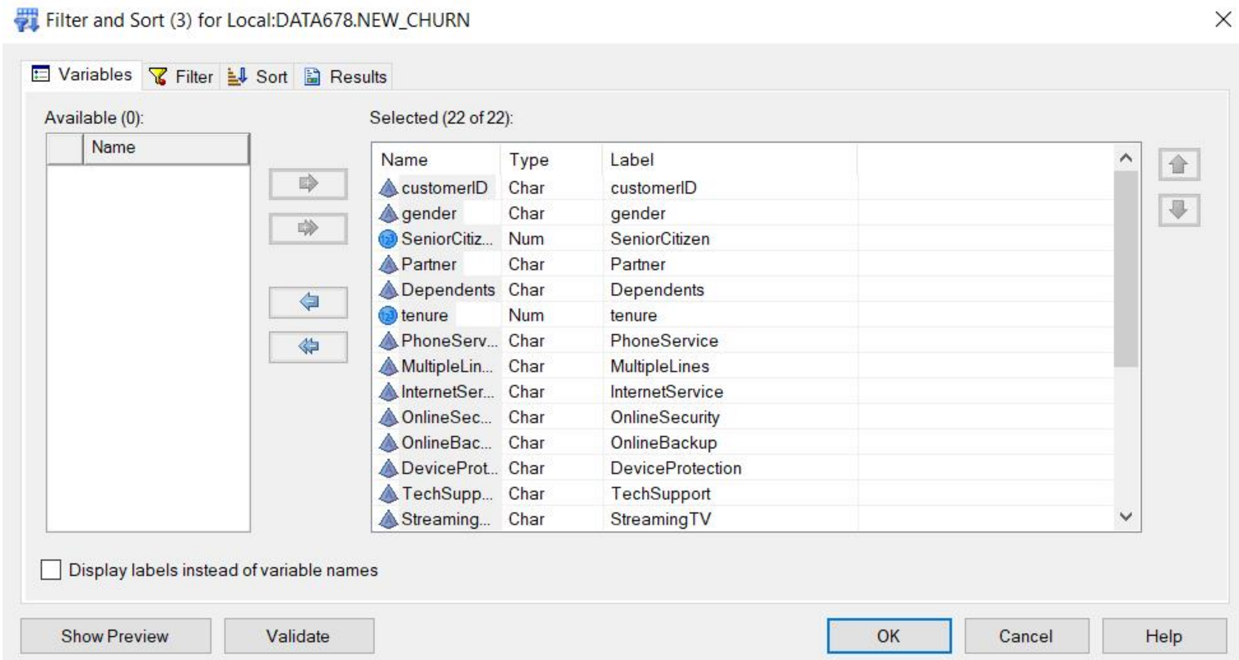
4) Splitting the dataset:

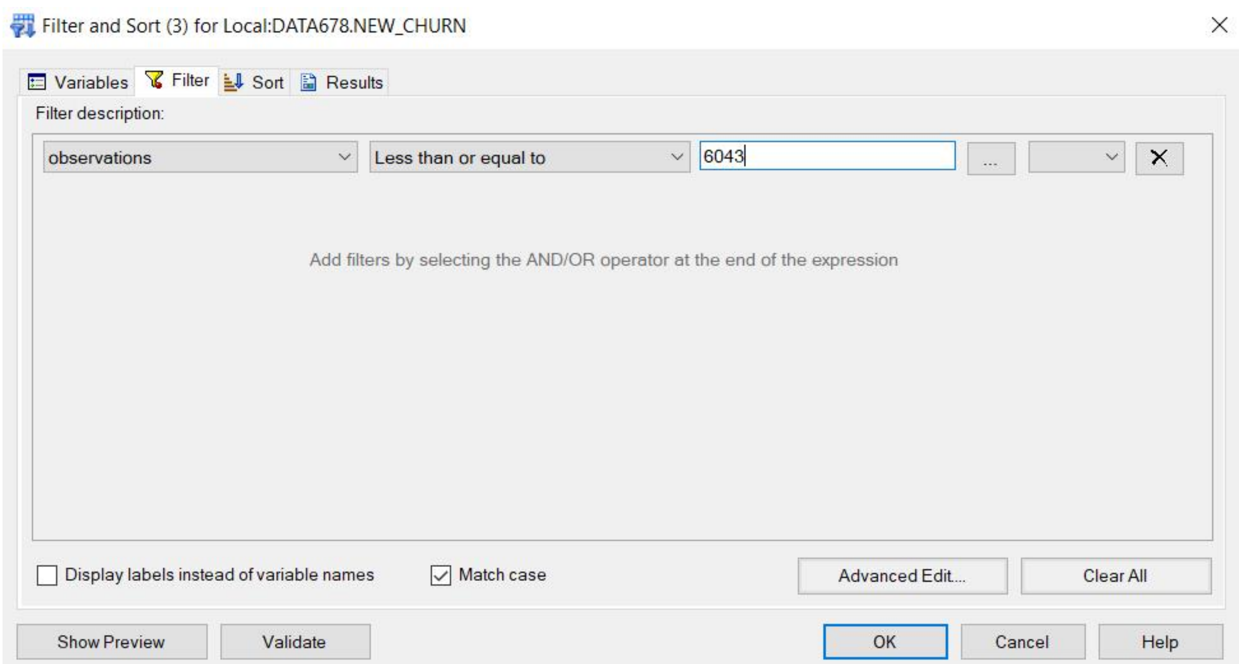
A) NEW_CHURN → FILTER AND SORT → SELECT AND DRAG TO THE RIGHT ALL THE VARIABLES → FILTER → (observations > Less than or equal to > 6043) → RESULTS (to permanently save the first dataset) → CHANGE → NAME THE FILE (or keep its name at default, we'll keep it at default) → SAVE

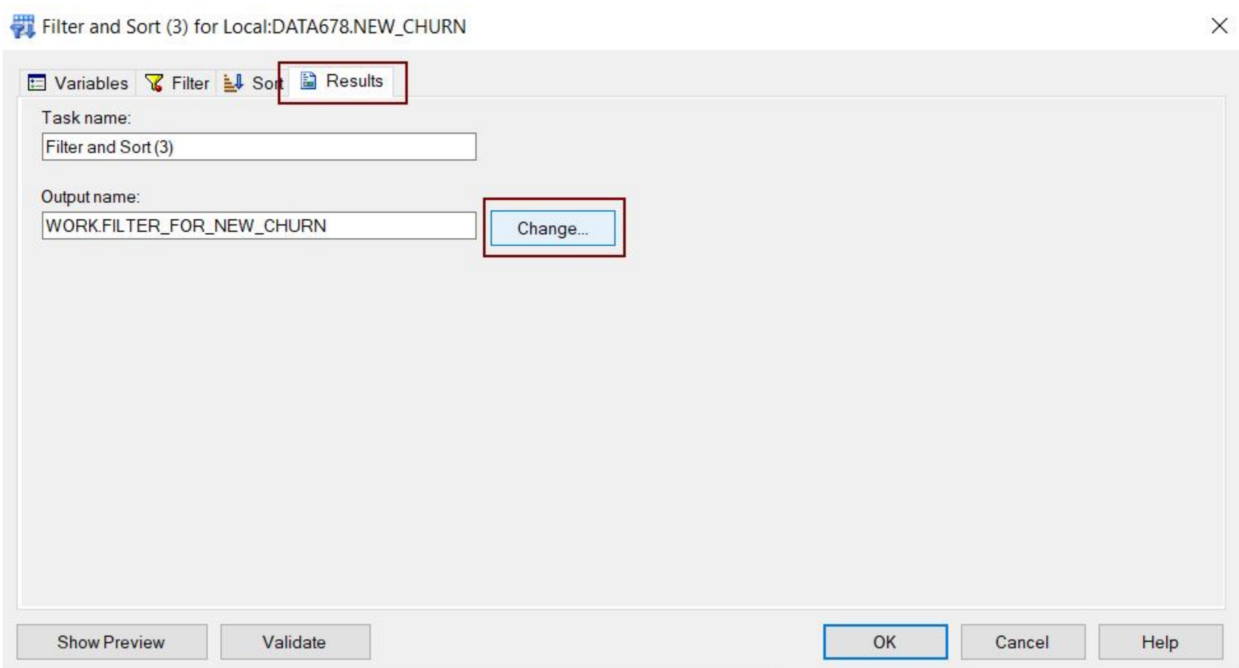
Doing this you will create the file called FILTER_FOR_NEW_CHURN with 6043 observations, which you will use to build models on.

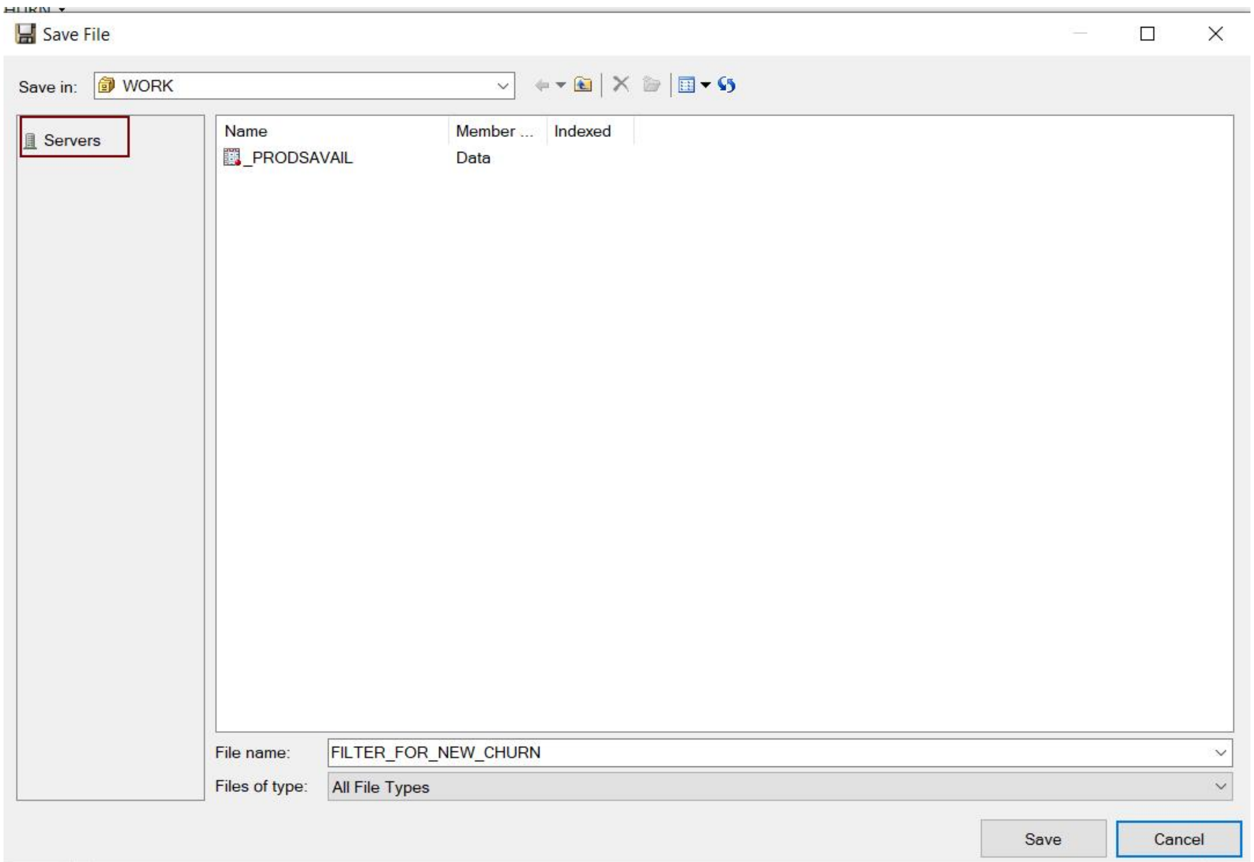


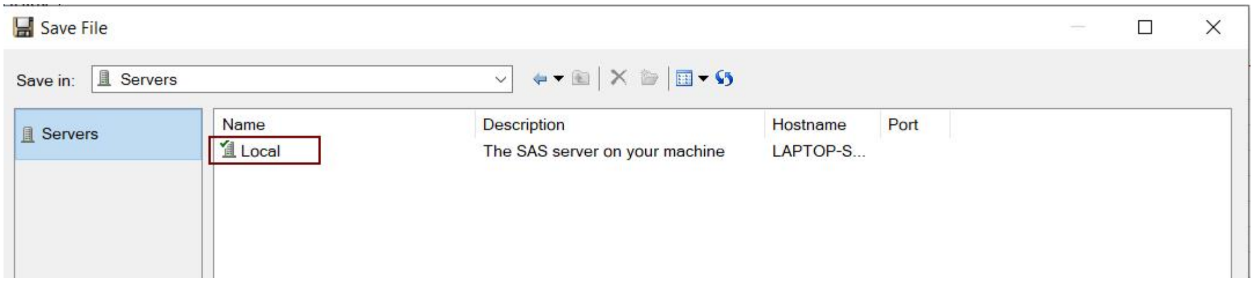


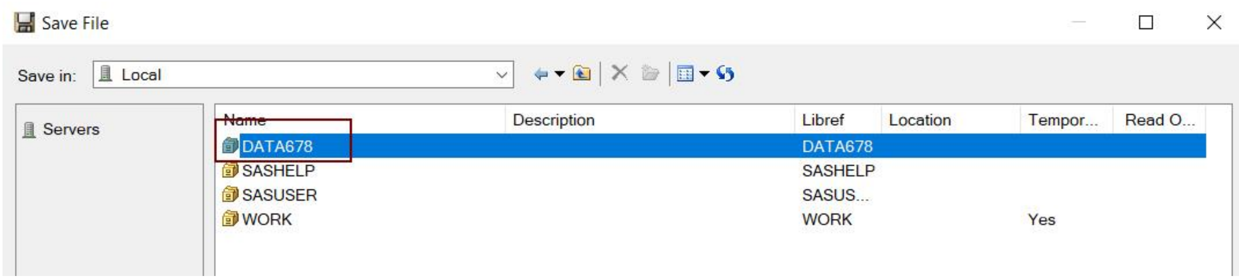

Filter and Sort (3) for Local:DATA678.NEW_CHURN
Variables
Filter
Sort
Results
Available (0):
Name
Selected (22 of 22):
Name
Type
Label
customerID Char customerID
gender Char gender
SeniorCitiz... Num SeniorCitizen
Partner Char Partner
Dependents Char Dependents
tenure Num tenure
PhoneServ... Char PhoneService
MultipleLin... Char MultipleLines
InternetSer... Char InternetService
OnlineSec... Char OnlineSecurity
OnlineBac... Char OnlineBackup
DeviceProt... Char DeviceProtection
TechSupp... Char TechSupport
Streaming... Char StreamingTV
Display labels instead of variable names
Show Preview
Validate
OK
Cancel
Help

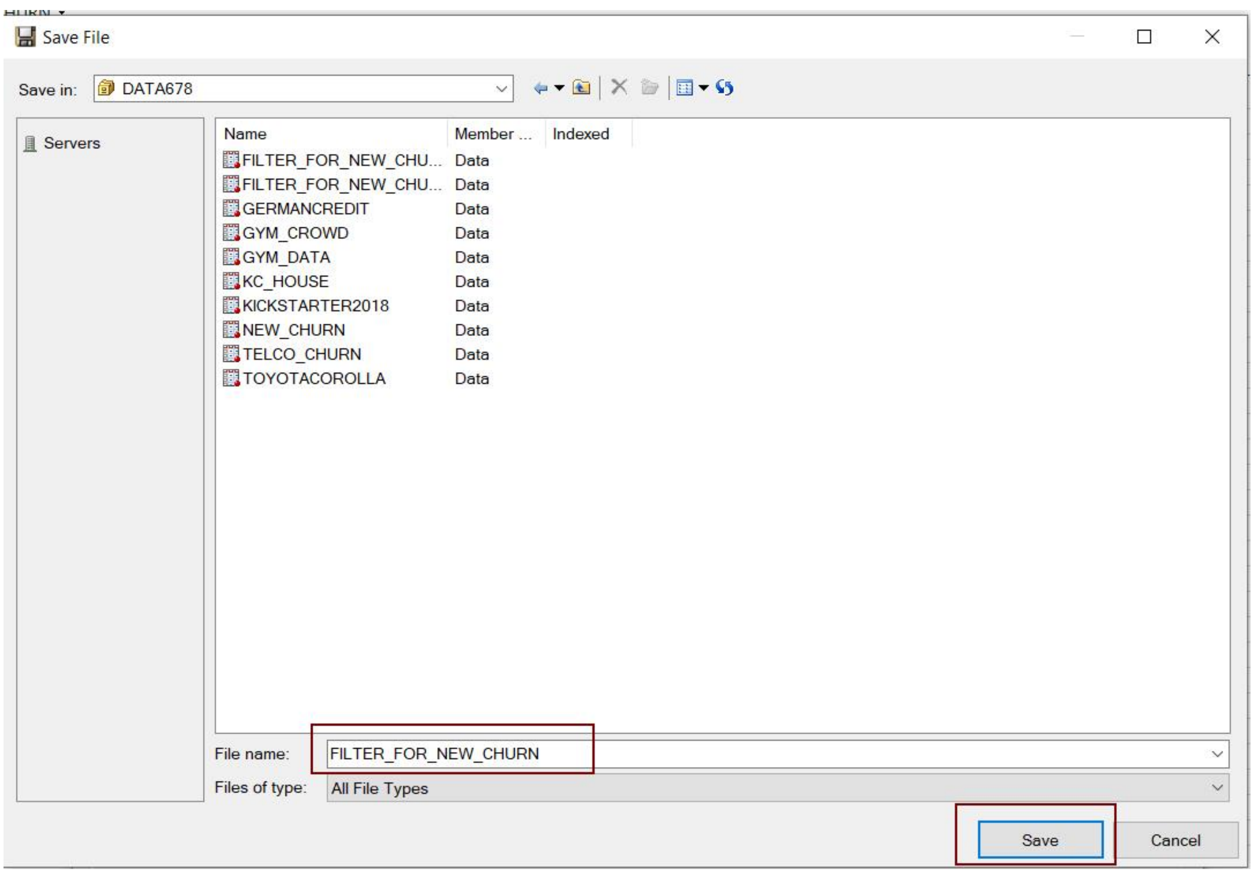

Filter and Sort (3) for Local:DATA678.NEW_CHURN
Variables
Filter
Sort
Results
Filter description:
observations
Less than or equal to
6043
Add filters by selecting the AND/OR operator at the end of the expression
Display labels instead of variable names
Match case
Advanced Edit...
Clear All
Show Preview
Validate
OK
Cancel
Help


Filter and Sort (3) for Local:DATA678.NEW_CHURN
Variables
Filter
Sort
Results
Task name:
Filter and Sort (3)
Output name:
WORK.FILTER_FOR_NEW_CHURN
Change...
Show Preview
Validate
OK
Cancel
Help


Save File
Save in:
WORK
Servers
Name
Member ...
Indexed
_PRODSAVAIL
Data
File name:
FILTER_FOR_NEW_CHURN
Files of type:
All File Types
Save
Cancel

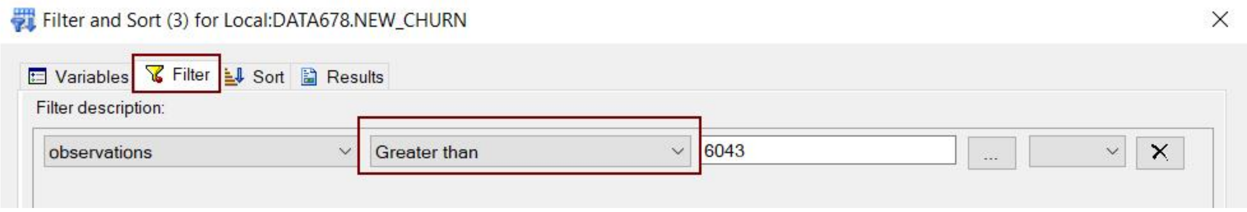






B) REPEAT THE PROCEDURE to create a file for scoring with 1000 observations (open NEW_CHURN again, the original file):

NEW_CHURN → FILTER AND SORT → SELECT AND DRAG TO THE RIGHT ALL THE VARIABLES → FILTER → (observations > Greater than > 6043) → RESULTS (to permanently save the first dataset) → CHANGE → NAME THE FILE (FILTER_FOR_NEW_CHURN_SCORE) → SAVE



You're ready to start working in SAS EM!

SAS EM: K-NN, NEURAL NETWORKS, ENSEMBLE, SCORE

1) Open an existing project or create a new one.

2) If it's a new project, run a libname statement (make sure that your libname is referring to the actual physical location of the newly created files):

- filter_for_new_churn
- filter_for_new_churn_score

For example:

```
libname DATA678 'C:\MBA678\Course_data';
```

3) Create a datasource:

A) from the filter_for_new_churn and set the variables as follows:

Name	Role	Level
Churn	Target	Binary
Contract	Input	Nominal
Dependents	Input	Binary
DeviceProtectic	Input	Nominal
InternetService	Input	Nominal
MonthlyCharge	Input	Interval
MultipleLines	Input	Nominal
OnlineBackup	Input	Nominal
OnlineSecurity	Input	Nominal
PaperlessBillin	Input	Binary
Partner	Input	Binary
PaymentMetho	Input	Nominal
PhoneService	Input	Binary
SeniorCitizen	Input	Binary
StreamingMovi	Input	Nominal
StreamingTV	Input	Nominal
TechSupport	Input	Nominal
TotalCharges	Input	Interval
customerID	Rejected	Nominal
gender	Input	Binary
observations	Rejected	Interval
tenure	Input	Interval

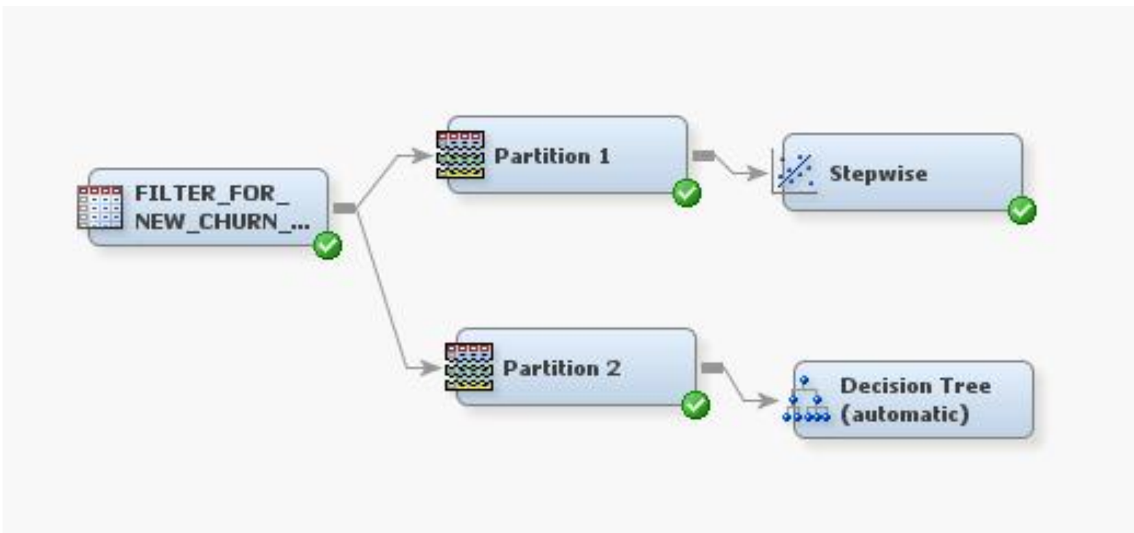
B) from the filter_for_new_churn_score and set the variables in the same way.

4) Create a new diagram (if you still don't have it): call it – Churn.

5) Drag the filter_for_new_churn into the diagram. Follow these steps:

a) Add two data partition nodes:
   a. Partition 1: Train 70%, Validate 30%
   b. Partition 2: Train 50%, Validate 25%, Test 25%
b) Attach a regression model to the Partition 1 node (Model selection: Stepwise)
c) Attach a decision tree node to the Partition 2 node (automatic)

Later, we will use these models for comparison purposes.

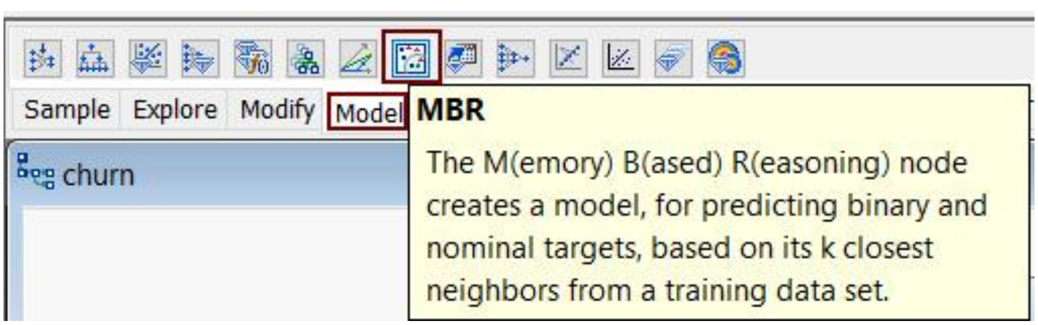


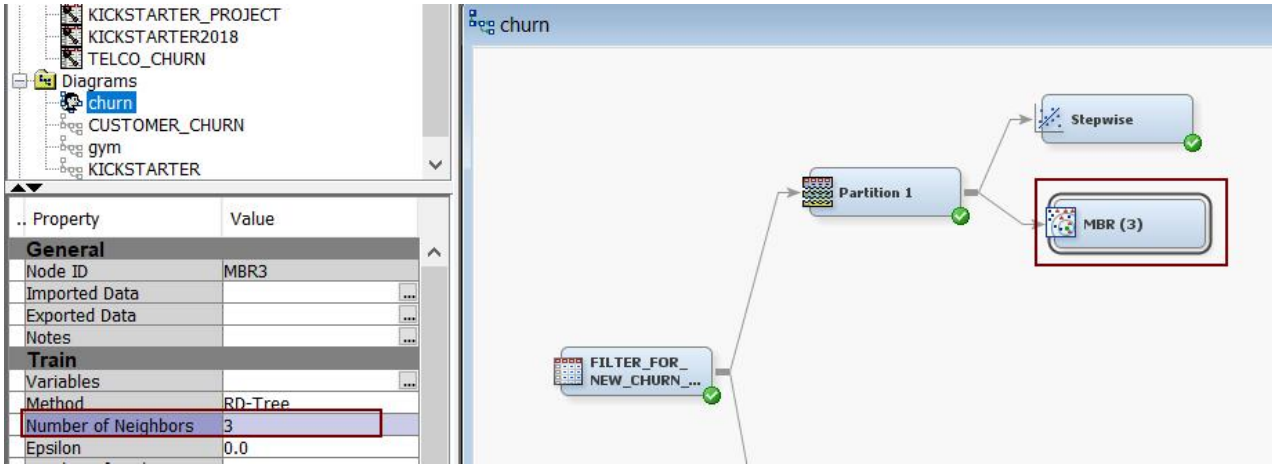
KNN

6) K-NN (the number of neighbours should always be odd):

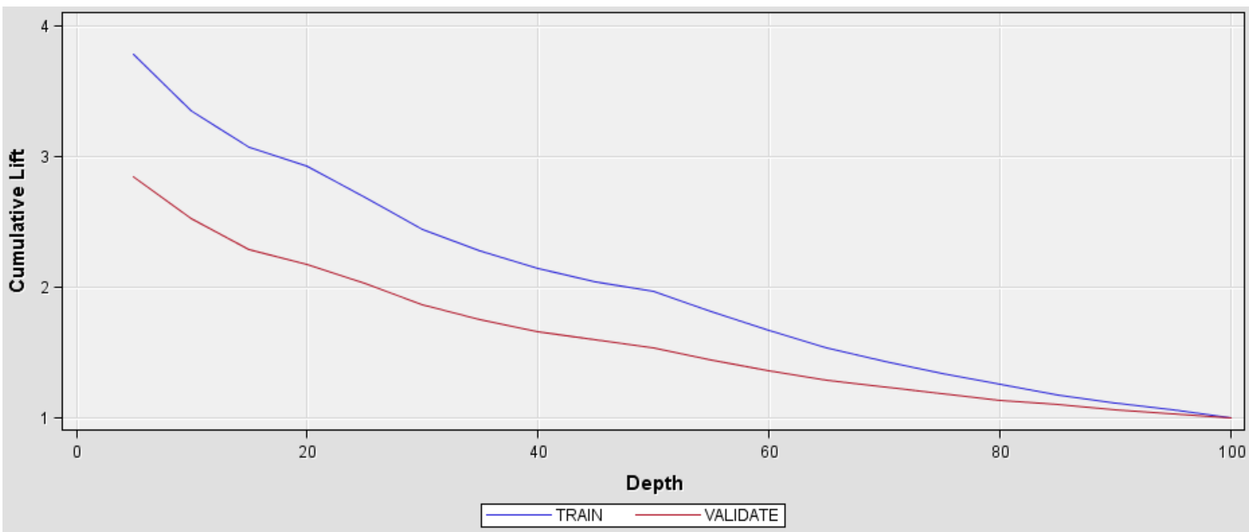
A) At first we will run the KNN model with 3 neighbors.

MODEL → MBR (add it to Partition 1) → SET NUMBER OF NEIGHBOURS TO **3** →RUN → RESULTS





If you look at the LIFT and errors, you will see that the model overfits the data and thus, it is not going to be a good predictive model.

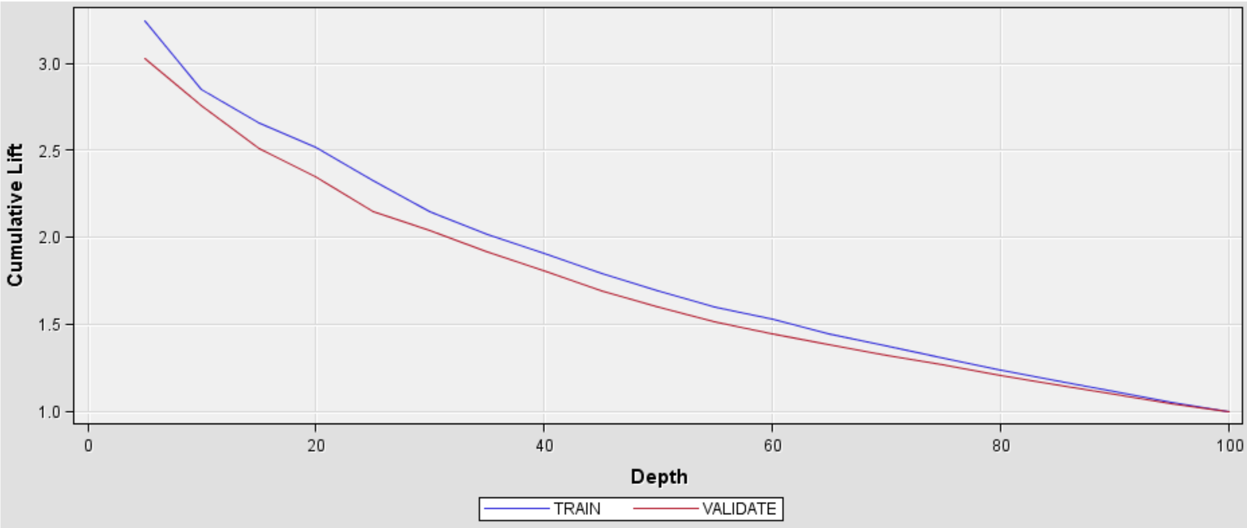


Target	Target Label	Fit Statistics	Statistics Label	Train	Validation	Test
Churn		NW	Number of Estima...	3	.	
Churn		NOBS	Sum of Frequencies	4221	1815	
Churn		SUMW	Sum of Case Wei...	8442	3630	
Churn		DFT	Total Degrees of ...	4221	.	
Churn		DFM	Model Degrees of ...	3	.	
Churn		DFE	Degrees of Freed...	4218	.	
Churn		ASE	Average Squared ...	0.099924	0.179063	
Churn		RASE	Root Average Squ...	0.316107	0.423159	
Churn		DIV	Divisor for ASE	8442	3630	
Churn		SSE	Sum of Squared E...	843.5556	650	
Churn		MSE	Mean Squared Error	0.099995	0.179063	
Churn		RMSE	Root Mean Squar...	0.316219	0.423159	
Churn		AVERR	Average Error Fun...	0.289355	0.82772	
Churn		ERR	Error Function	2442.736	3004.624	
Churn		MAX	Maximum Absolut...	1	1	
Churn		FPE	Final Prediction Er...	0.100066	.	
Churn		RFPE	Root Final Predicti...	0.316332	.	
Churn		AIC	Akaike's Informati...	2448.736	.	
Churn		SBC	Schwarz's Bayesi...	2467.779	.	

B) At first we will run the KNN model with 17 neighbors. Add a new mode to the Partition 1 node.

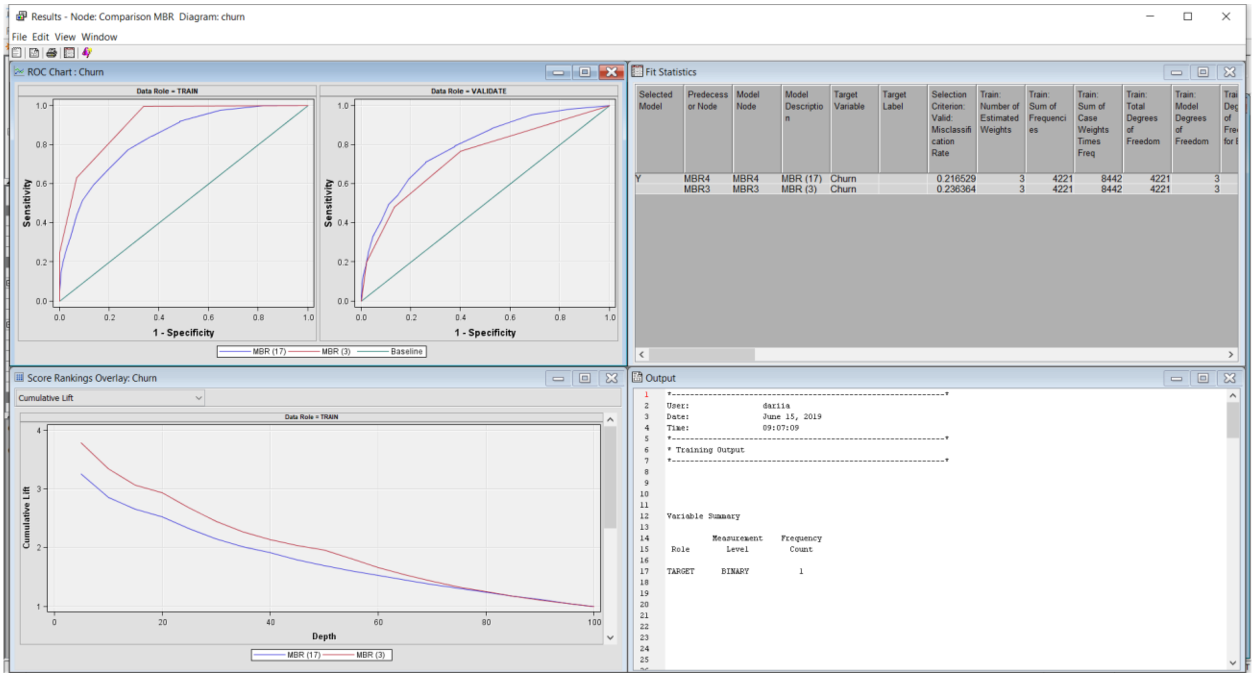
MODEL → MBR (add it to Partition 1) → SET NUMBER OF NEIGHBOURS TO **17** →RUN → RESULTS

As you can see from the lift and the errors, the results are much better.



Target	Target Label	Fit Statistics	Statistics Label	Train	Validation	Test
Churn		NW	Number of Estima...	3	.	
Churn		NOBS	Sum of Frequencies	4221	1815	
Churn		SUMW	Sum of Case Wei...	8442	3630	
Churn		DFT	Total Degrees of ...	4221	.	
Churn		DFM	Model Degrees of ...	3	.	
Churn		DFE	Degrees of Freed...	4218	.	
Churn		ASE	Average Squared ...	0.1389	0.150253	
Churn		RASE	Root Average Squ...	0.372692	0.387625	
Churn		DIV	Divisor for ASE	8442	3630	
Churn		SSE	Sum of Squared E...	1172.591	545.4187	
Churn		MSE	Mean Squared Error	0.138998	0.150253	
Churn		RMSE	Root Mean Squar...	0.372825	0.387625	
Churn		AVERR	Average Error Fun...	0.423383	0.474705	
Churn		ERR	Error Function	3574.202	1723.18	
Churn		MAX	Maximum Absolut...	0.941176	1	
Churn		FPE	Final Prediction Er...	0.139097	.	
Churn		RFPE	Root Final Predicti...	0.372957	.	
Churn		AIC	Akaike's Informati...	3580.202	.	
Churn		SBC	Schwarz's Bayesi...	3599.245	.	

Now, let's compare the models and choose the best one by attaching the Model Comparison node to the two MBRs. According to SAS and we can also tell, MBR17 is much better than the MBR3 model.



NEURAL NETWORKS

7) Neural networks:

In SAS you can find three nodes responsible for Neural networks:

- MODEL → Neural Network (here you can assign a number of neurons and a number of iterations)
- MODEL → Autoneural (the node will choose the best architecture for you)
- HPDM → HP Neural (here you have all the control over the number of hidden layers, neurons and also iterations as well as how many sets of iterations you want to run)

The Neural networks need partitioning into TRAIN, VALIDATE and TEST datasets. So, we will be attaching them to the Partition 2 node.

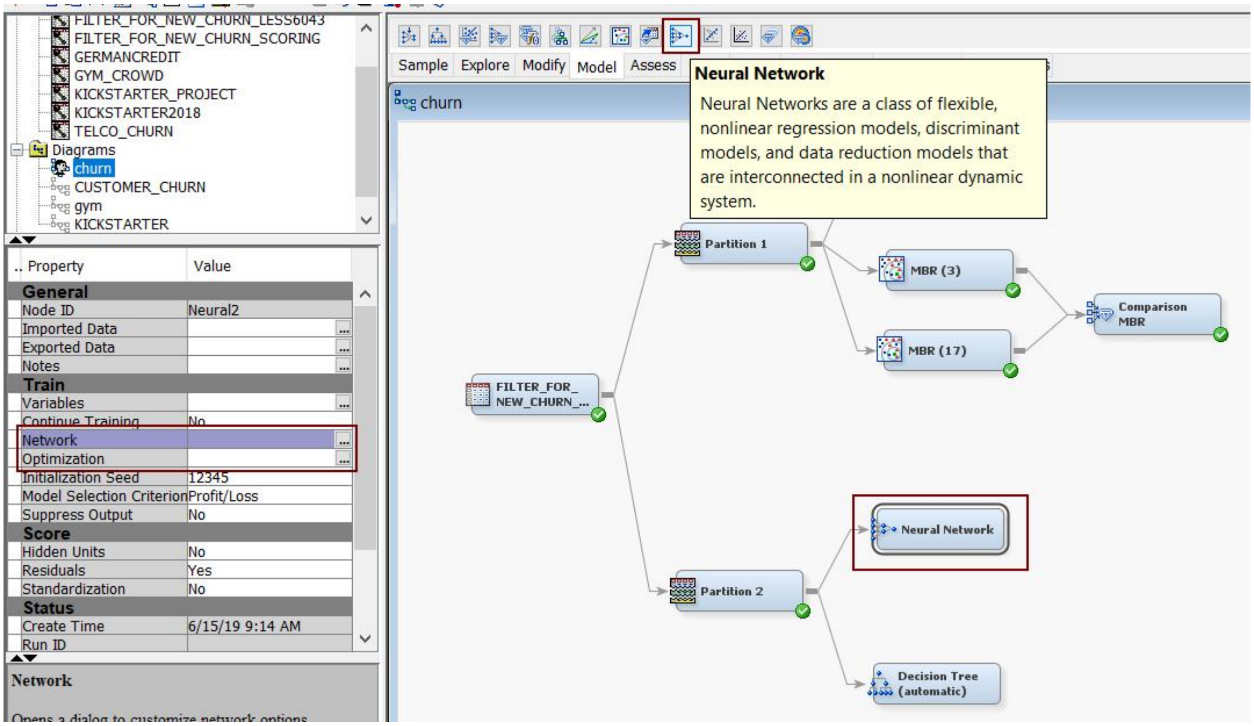
Since neural networks are bad at selecting variables for analysis, we will use a logistic regression model, which will serve as a variable filter. Its output will not be used when the Neural model is trained; however, the variables rejected by the regression won't serve as inputs in the Neural network. It will help the Neural network model avoid overfitting.

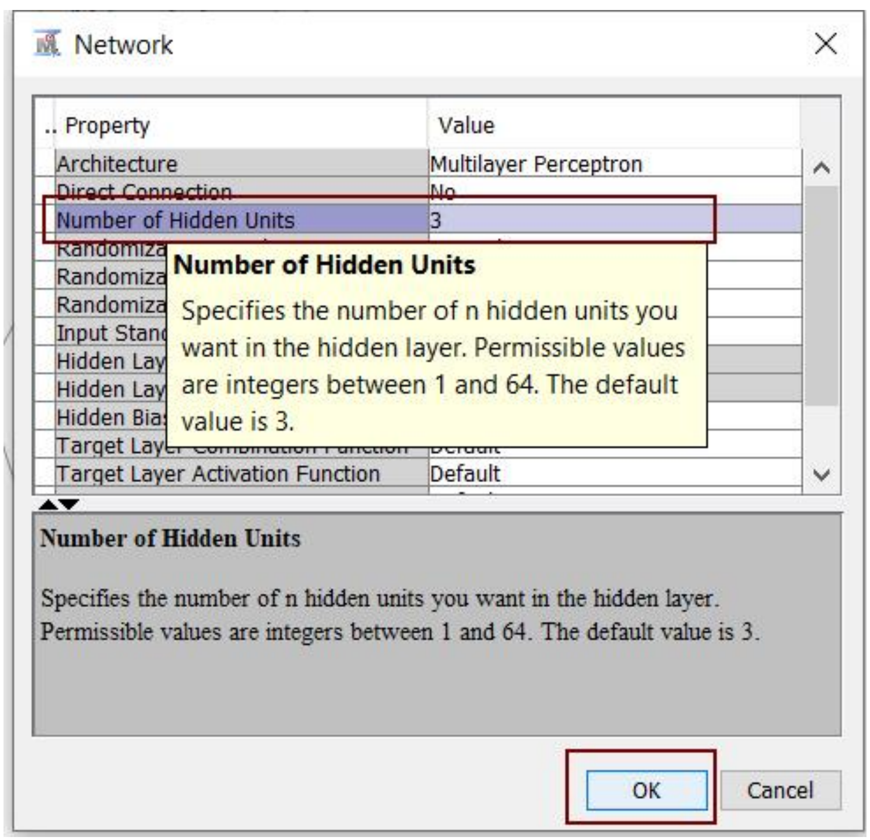
MODEL → REGRESSION (attach it to Partition 2; stepwise)

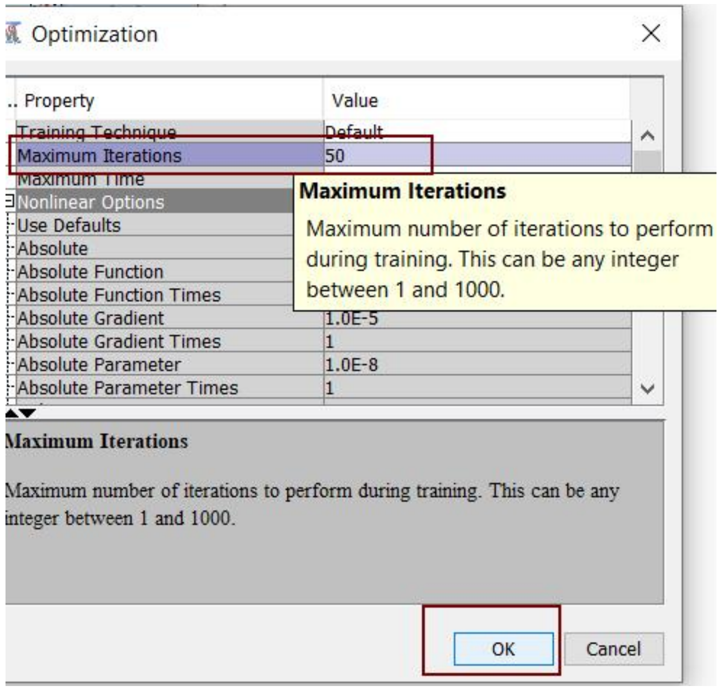
After the stepwise regression attached to Partition 2 follow these steps:

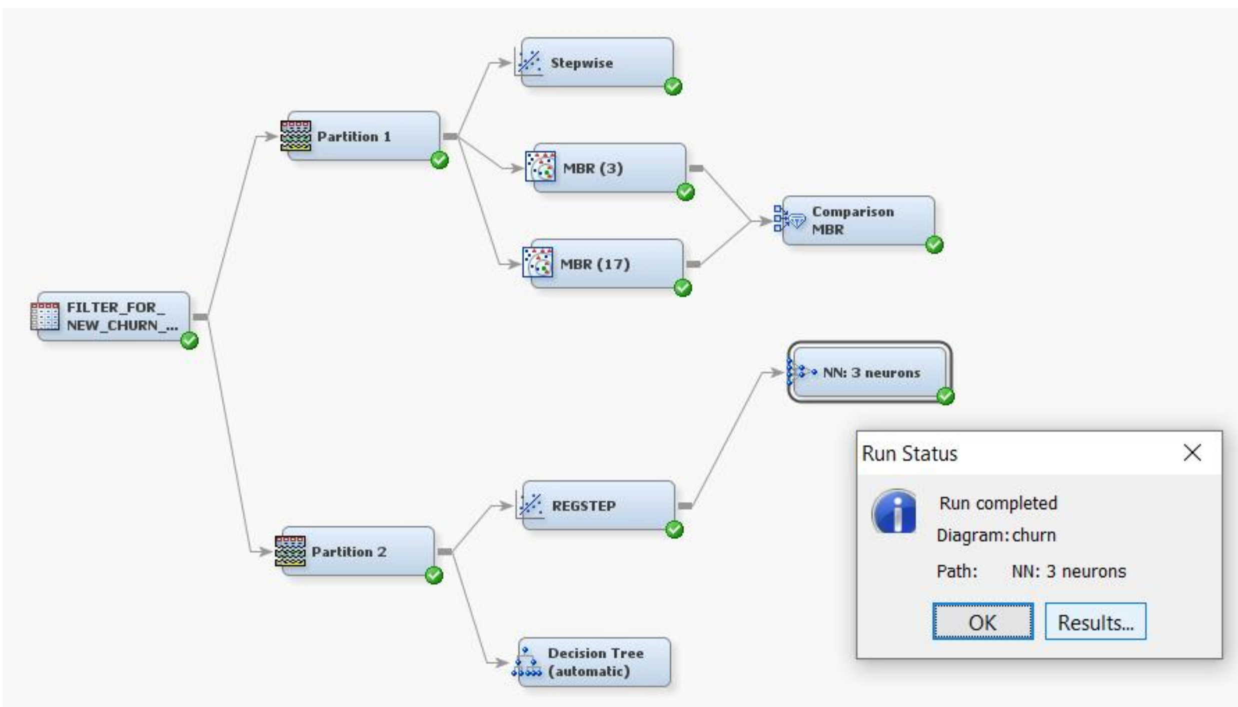
A) NEURAL NETWORK:

A.1. (MODEL → STEPWISE REGRESSION) → MODEL → NEURAL NETWORK → (LEFT) NETWORK: click on the three dots → NUMBER OF HIDDEN UNITS (here you can change the number of neurons) – leave at default → OK → (left) OPTIMIZATION: click on the three dots → MAXIMUM ITERATIONS (here you can change the number of iterations) – leave at default → OK → RUN the node → RESULTS

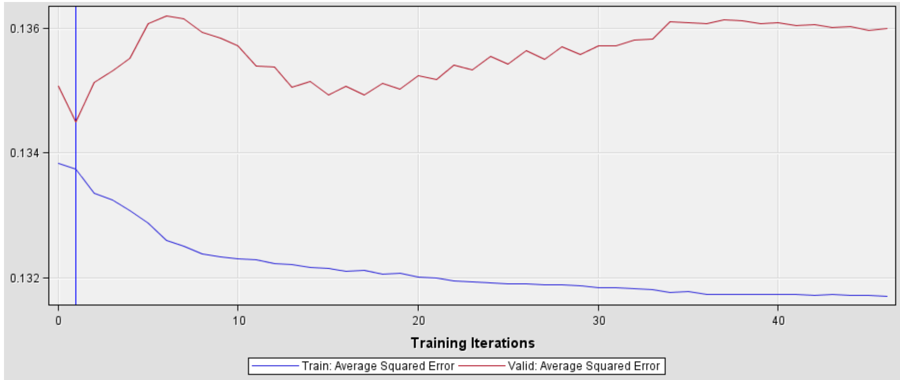








The iterations plot shows us that the convergence happened at iteration 1:



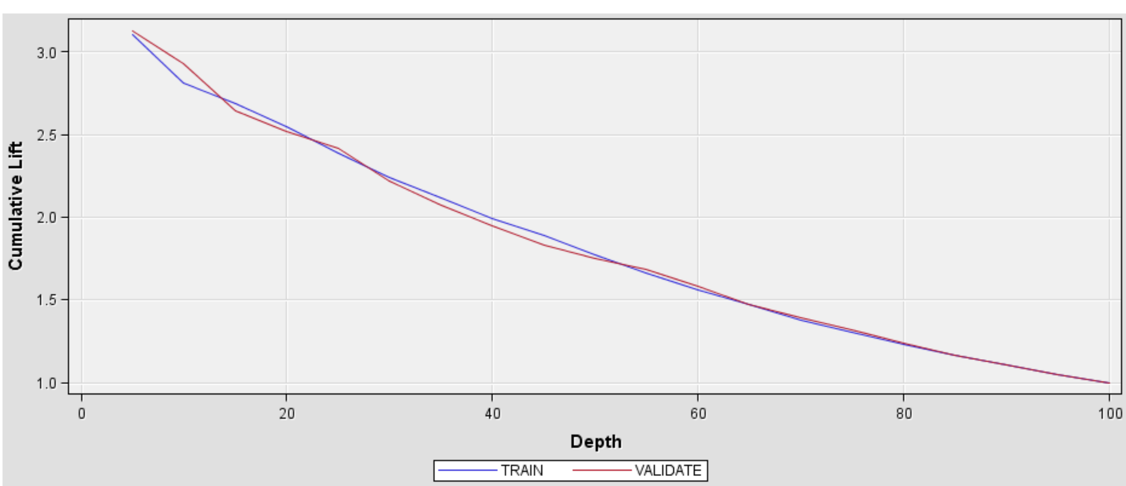
Check the output window to see if the convergence criterion has been satisfied, also check it out for how many iterations your model required.

```
46*      0      54      0      0.40698   0.000012   0.000433   0.00258      0.181

                              Optimization Results

Iterations                                 46  Function Calls                       56
Jacobian Calls                             48  Active Constraints                    0
Objective Function               0.4069819266  Max Abs Gradient Element   0.0004334047
Lambda                           0.0025806015  Actual Over Pred Change    0.1809418373
Radius                           0.1155184026

Convergence criterion (FCONV=0.0001) satisfied.
```

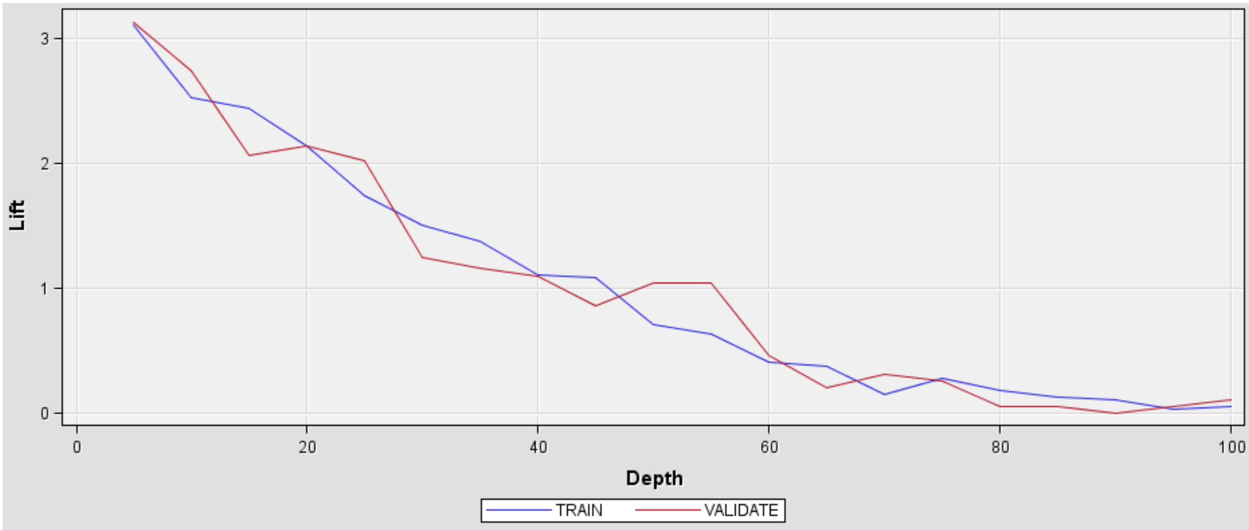
NOTE! If instead of the Convergence criterion satisfied you see another message, then you have to go to the OPTIMIZATIOn and increase a number of iterations, say to 100 or more and run the model again.

Now, you are ready to analyze the model. As usually, check the lift, cumulative lift and errors. Look for signs of the data overfitting as well.

Cumulative lift



Lift



Average Squared Error

Fit Statistics

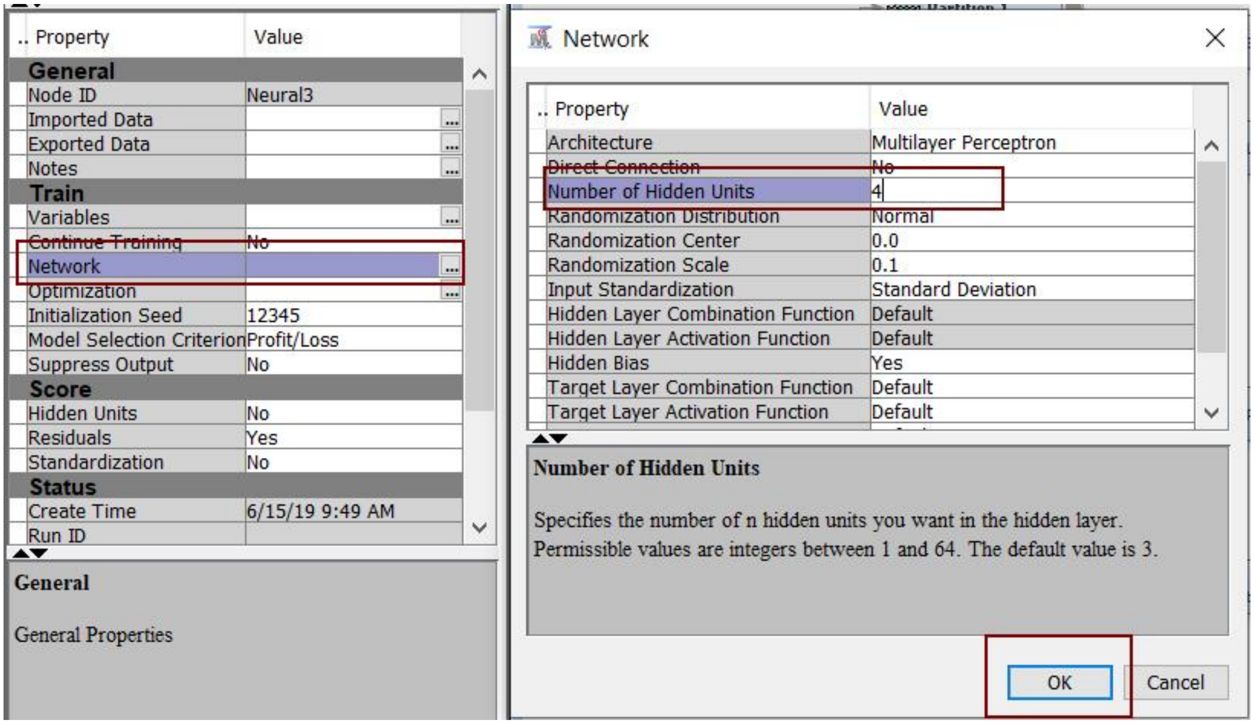
Target	Target Label	Fit Statistics	Statistics Label	Train	Validation	Test
Churn		DFT	Total Degrees of ...	3020	.	.
Churn		DFE	Degrees of Freed...	2959	.	.
Churn		DFM	Model Degrees of ...	61	.	.
Churn		NW	Number of Estima...	61	.	.
Churn		AIC	Akaike's Informati...	2615.426	.	.
Churn		SBC	Schwarz's Bayesi...	2982.22	.	.
Churn		ASE	Average Squared ...	0.133742	0.134499	0.137682
Churn		MAX	Maximum Absolut...	0.986933	0.98734	0.986915
Churn		DIV	Divisor for ASE	6040	3022	3024
Churn		NOBS	Sum of Frequencies	3020	1511	1512
Churn		RASE	Root Average Squ...	0.365707	0.366741	0.371056
Churn		SSE	Sum of Squared E...	807.8005	406.4554	416.3512
Churn		SUMW	Sum of Case Wei...	6040	3022	3024
Churn		FPE	Final Prediction Er...	0.139256	.	.
Churn		MSE	Mean Squared Error	0.136499	0.134499	0.137682
Churn		RFPE	Root Final Predicti...	0.37317	.	.
Churn		RMSE	Root Mean Squar...	0.369458	0.366741	0.371056
Churn		AVERR	Average Error Fun...	0.412819	0.4121	0.419501
Churn		ERR	Error Function	2493.426	1245.366	1268.571
Churn		MISC	Misclassification ...	0.195033	0.19722	0.194444
Churn		WRONG	Number of Wrong ...	589	298	294

As we can see, the model is pretty decent. It doesn’t overfit (thanks to the regression, perhaps).

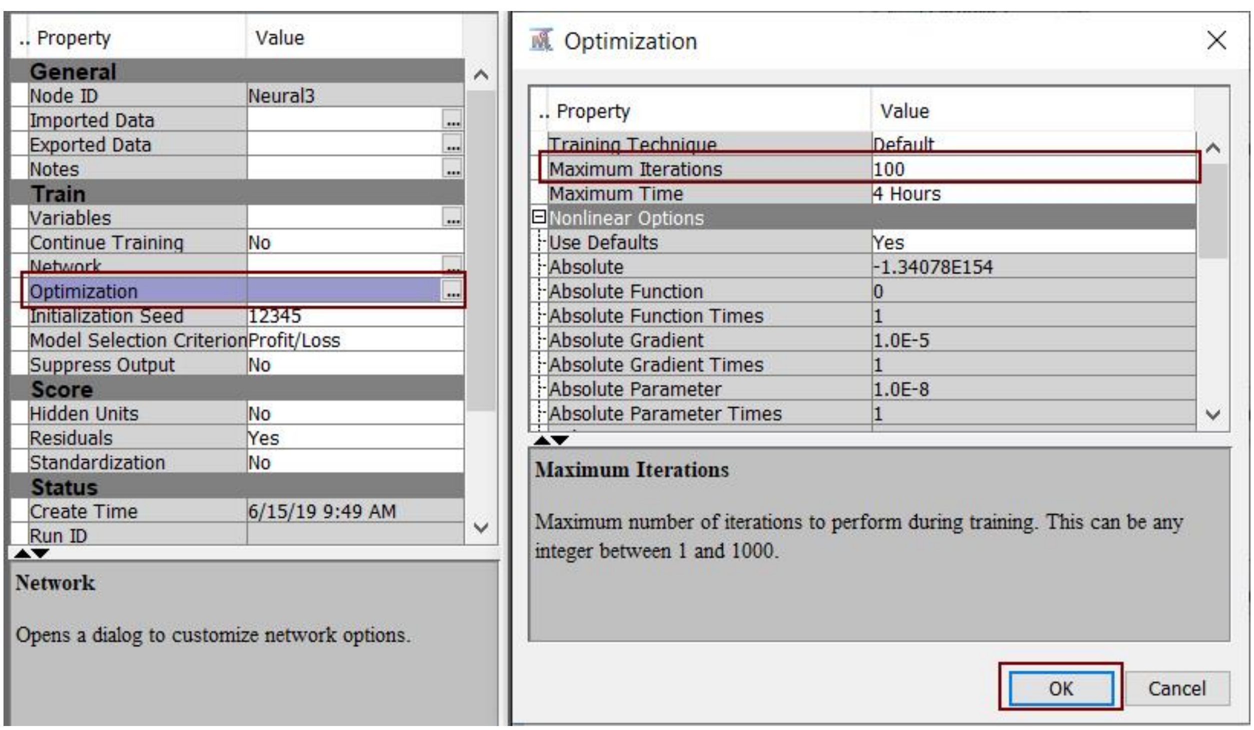
A.2. We will add another Neural Network node to the regression, but we will change the default number of hidden units and the number of iterations:

(MODEL → STEPWISE REGRESSION) → MODEL → NEURAL NETWORK → (LEFT) NETWORK: click on the three dots → NUMBER OF HIDDEN UNITS: 4 → OK → (left) OPTIMIZATION: click on the three dots → MAXIMUM ITERATIONS: 100 → OK → RUN the node → RESULTS

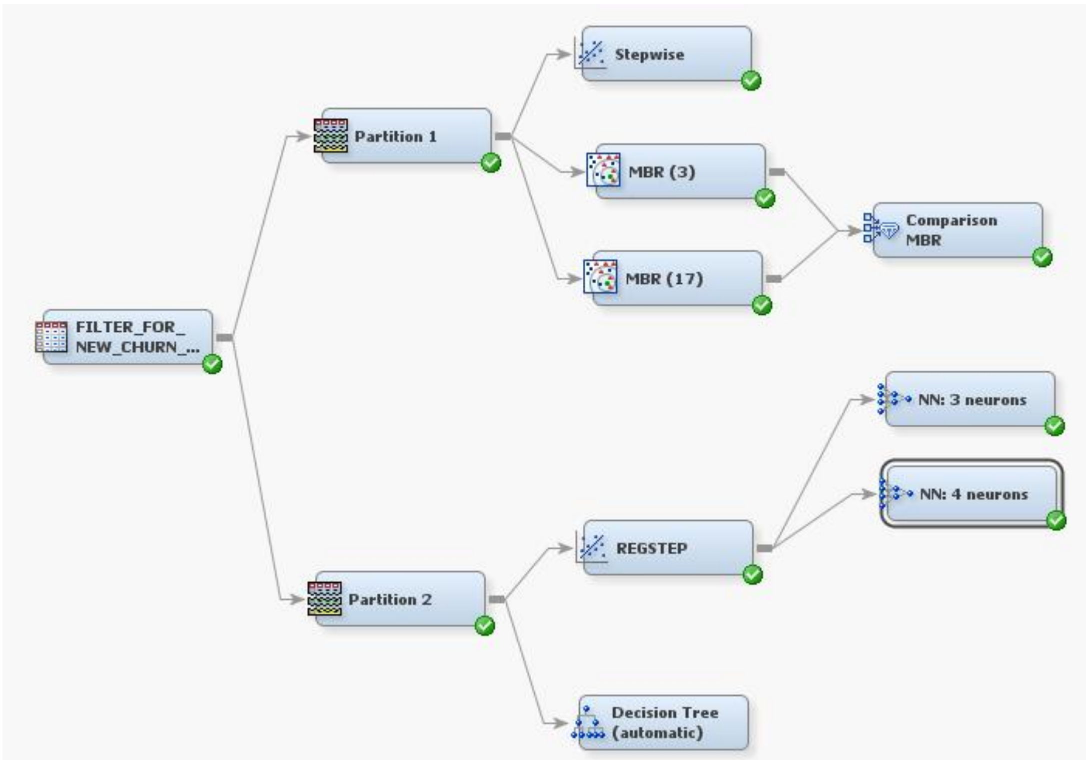
Number of hidden units:



Maximum iterations:



Again, check the iteration plot. Go to the output to see whether the convergence has occurred. If everything is fine, then take a note of how many iterations have passed and check out the model's performance by assessing its cumulative lift, lift and average squared error. You can also look into the misclassification rate. As you can see from the results, the model is pretty good (the graphs are not provided here).

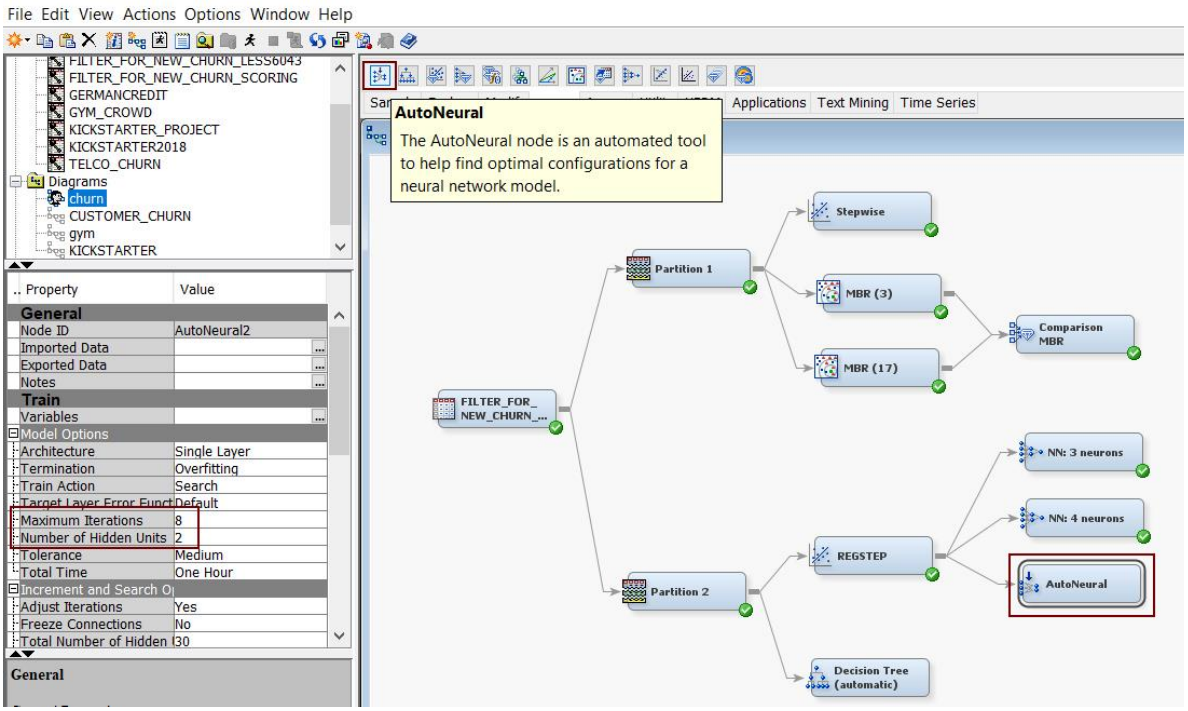


B) AUTONEURAL:

(MODEL → STEPWISE REGRESSION) → MODEL → AUTONEURAL→ RUN the node → RESULTS

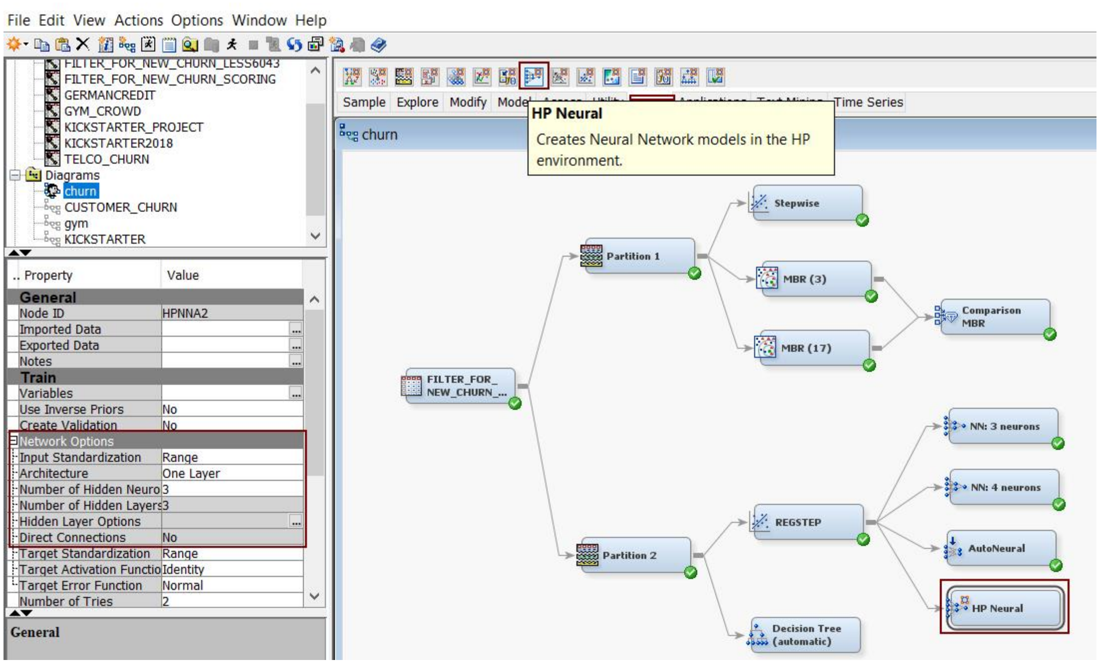
If you want to, you can change the number of iterations from 8 to a greater number as well as a number of neurons (Number of Hidden Units) to 3 or a greater number. We will keep the values at default.

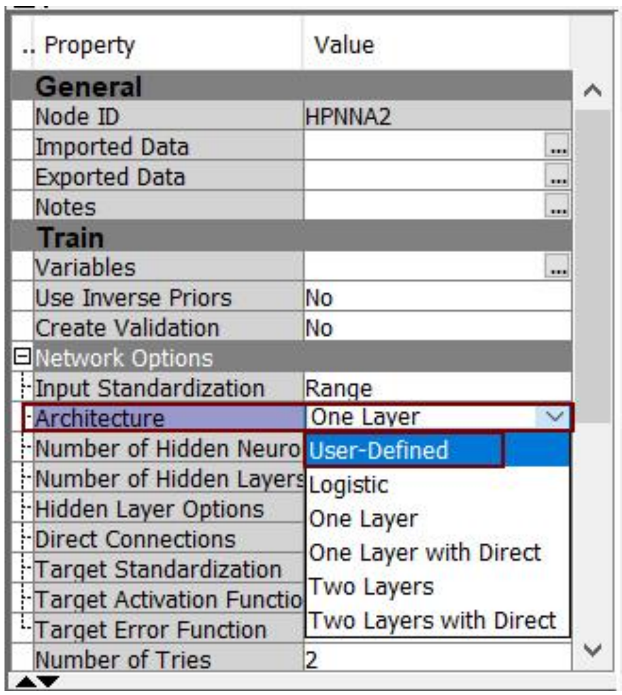
To check the model's performance, at first you should assess it by looking at whether the convergence has been satisfied as well as how many iterations have been performed. Then you can assess the model by looking at the lift, cumulative lift, Average squared error as well as the misclassification rate, for instance.

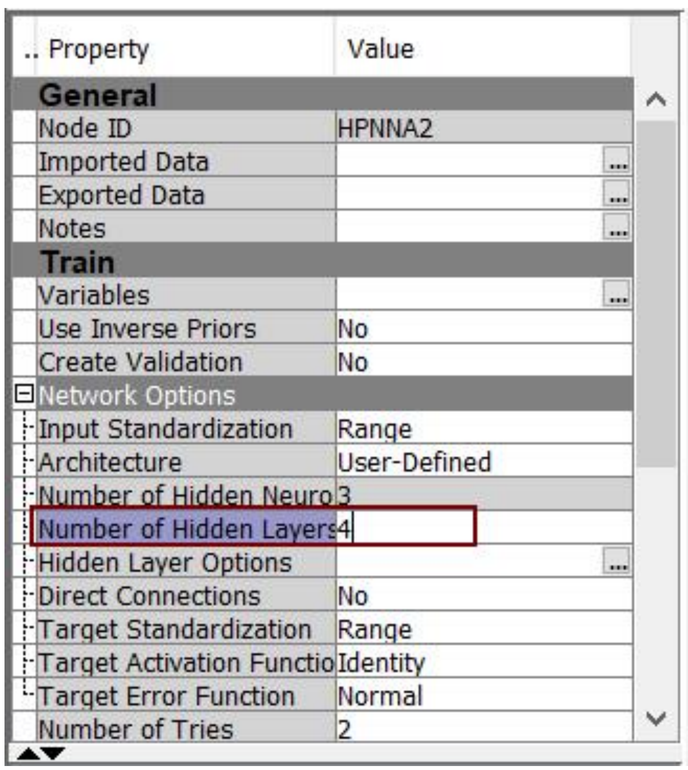


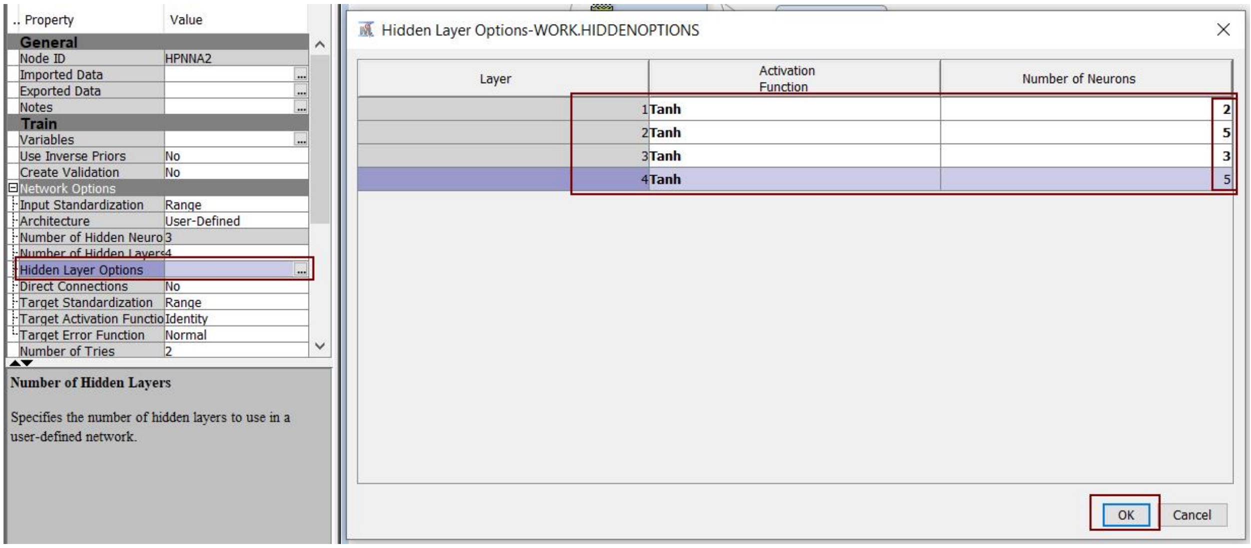
C) HP Neural:

(MODEL → STEPWISE REGRESSION) → HPDM → HP NEURAL → (LEFT) NETWORK OPTIONS → ARCHITECTURE → USER DEFINED → NUMBER OF HIDDEN LAYERS → 4 → HIDDEN LAYER OPTIONS → ESTIMATE DIFFERENT NUMBER OF NEURONS FOR EACH LAYER (if you want to) → LAYER1: 2, LAYER2: 5, LAYER3: 3, LAYER4: 5 → OK → RUN (THE NODE) → RESULTS

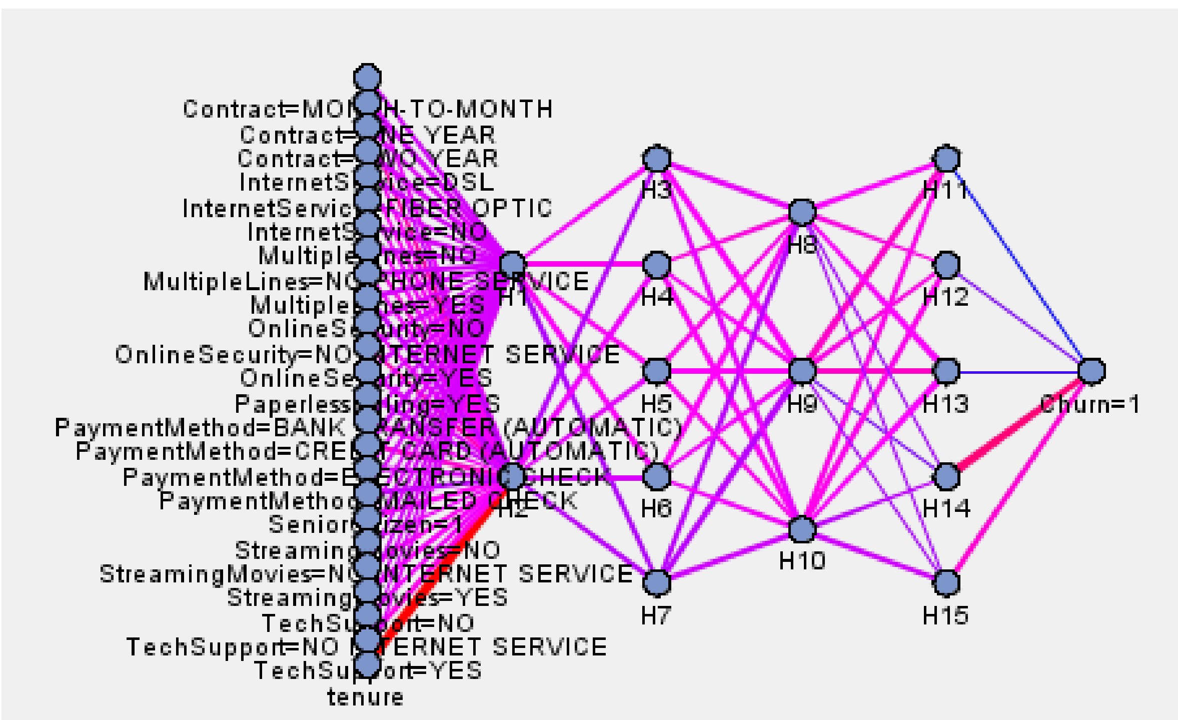



.. Property
Value
General
Node ID
HPNNA2
Imported Data
Exported Data
Notes
Train
Variables
Use Inverse Priors
No
Create Validation
No
Network Options
Input Standardization
Range
Architecture
One Layer
Number of Hidden Neuro
User-Defined
Number of Hidden Layers
Logistic
Hidden Layer Options
One Layer
Direct Connections
One Layer with Direct
Target Standardization
Two Layers
Target Activation Functio
Two Layers with Direct
Target Error Function
Number of Tries
2


.. Property
Value
General
Node ID
HPNNA2
Imported Data
Exported Data
Notes
Train
Variables
Use Inverse Priors
No
Create Validation
No
Network Options
Input Standardization
Range
Architecture
User-Defined
Number of Hidden Neuro
3
Number of Hidden Layers
4
Hidden Layer Options
Direct Connections
No
Target Standardization
Range
Target Activation Functio
Identity
Target Error Function
Normal
Number of Tries
2



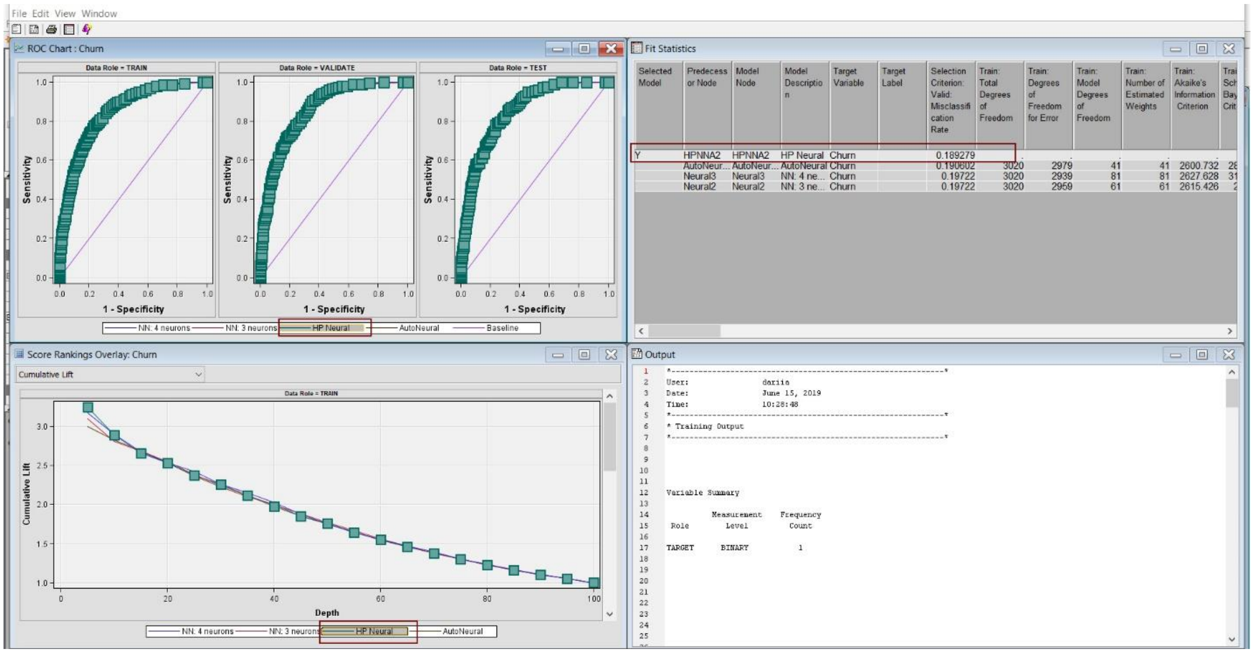
Check if the model converged. Then start analyzing the model. You can also see the model's architecture. Remember to assess the lift, cumulative lift and other goodness of fit and model's performance measures.



If you assess the measures, you will see that the model's performance is adequate.

Now, let's compare the models by using the Model Comparison node and select the best neural network model from the ones we ran.

According to SAS, the best model is the HP Neural.



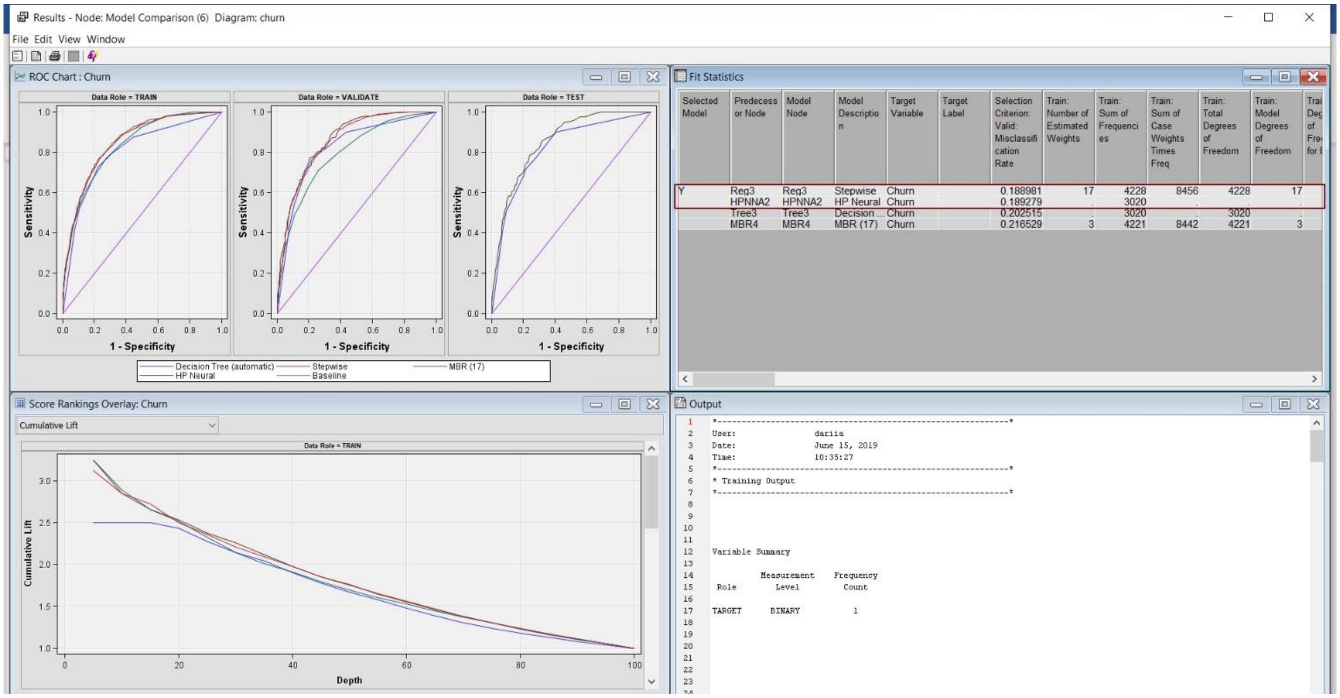
ENSEMBLE

8) Assess the models and choose the best.

ASSESS → MODEL COMPARISON: ATTACH TO Regression, MBR17, HP Neural and Decision tree.

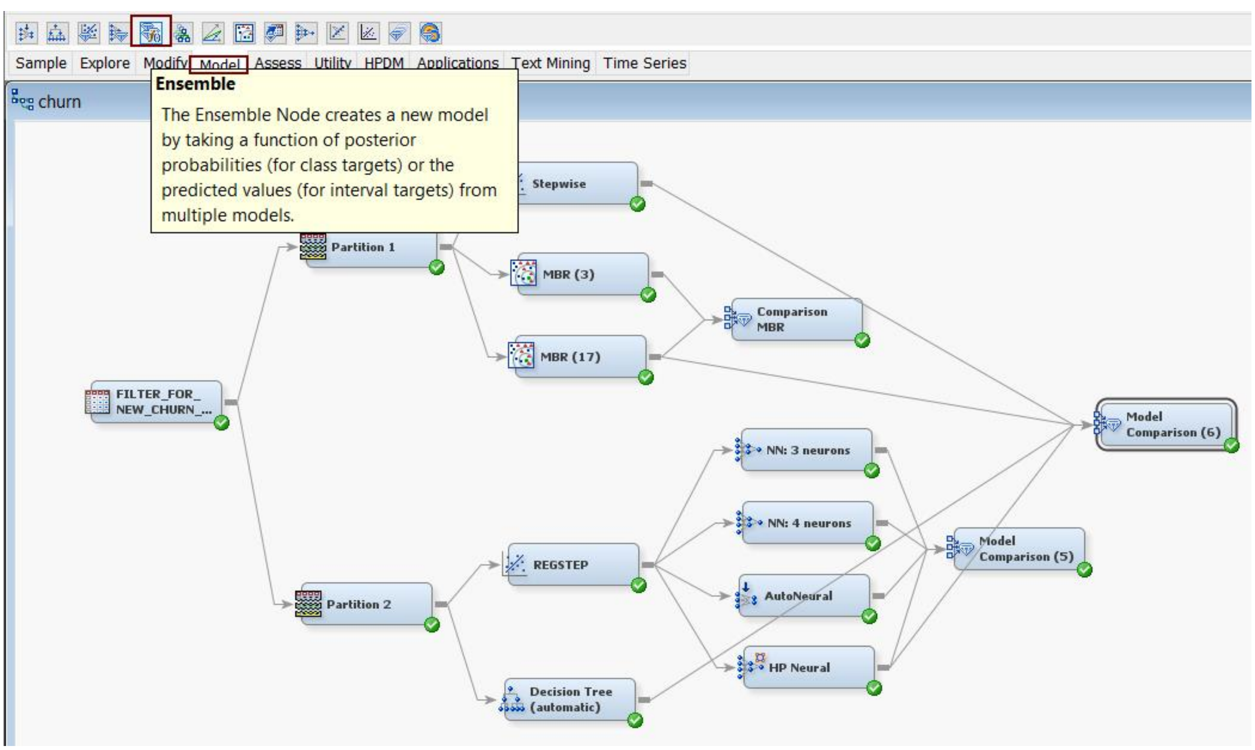
For Ensemble you will need two best models. Ensemble will combine these two models to create a super model (if you are lucky 😉). You should only attach it to the best models, otherwise, Ensemble will not perform well.

According to SAS, the best two models are Regression (attached to Partition1) and HP Neural. We will use them to build Ensemble.

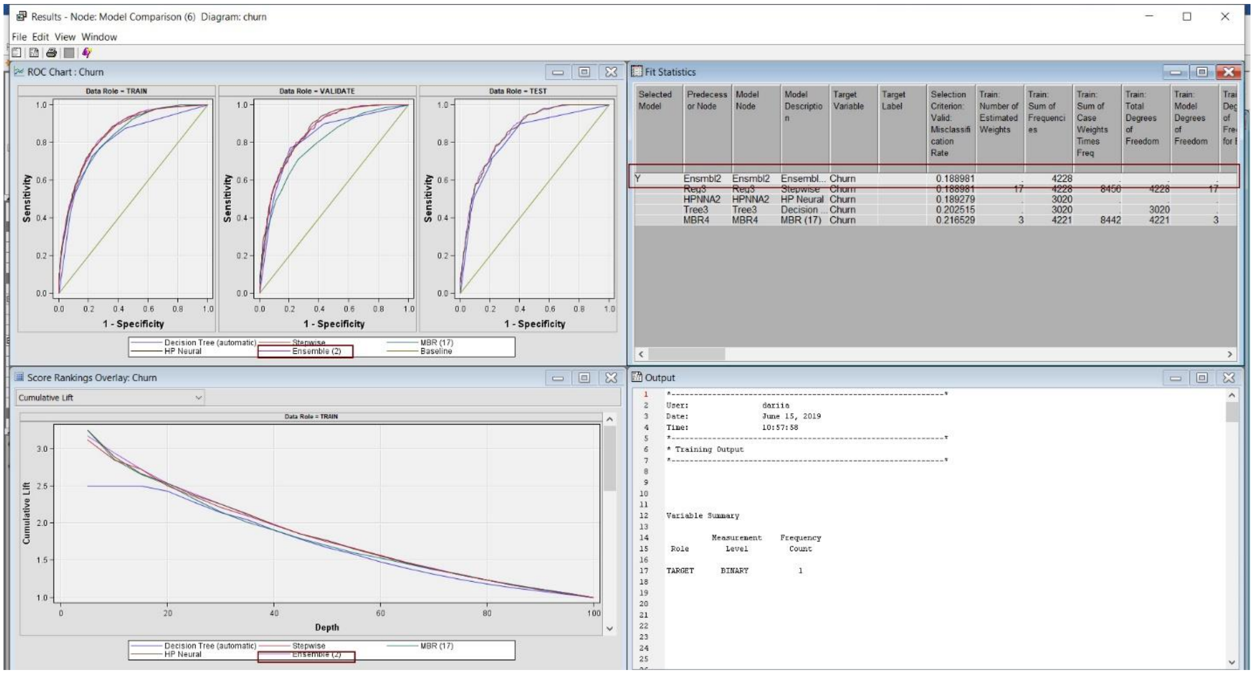


9) Create the Ensemble model:

MODEL → ENSEMBLE → DRAG AND ATTACH TO IT: STEPWISE (FROM PARTITION1) + HP NEURAL → ATTACH ENSEMBLE TO THE MASTER MODEL COMPARISON NODE → RUN



We can click on the results to see whether the ENSEMBLE model performed better than the others. According to SAS, Ensemble is the best model. Again, you can also go back to the Ensemble node, right-click on it to estimate the model's performance. The results are ok, the model doesn't seem to overfit. So, this model will be the one, we will use for deployment.

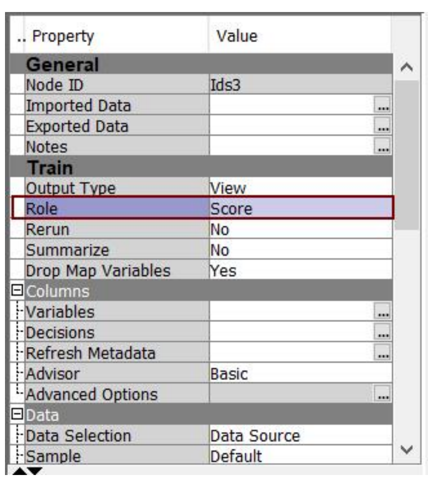


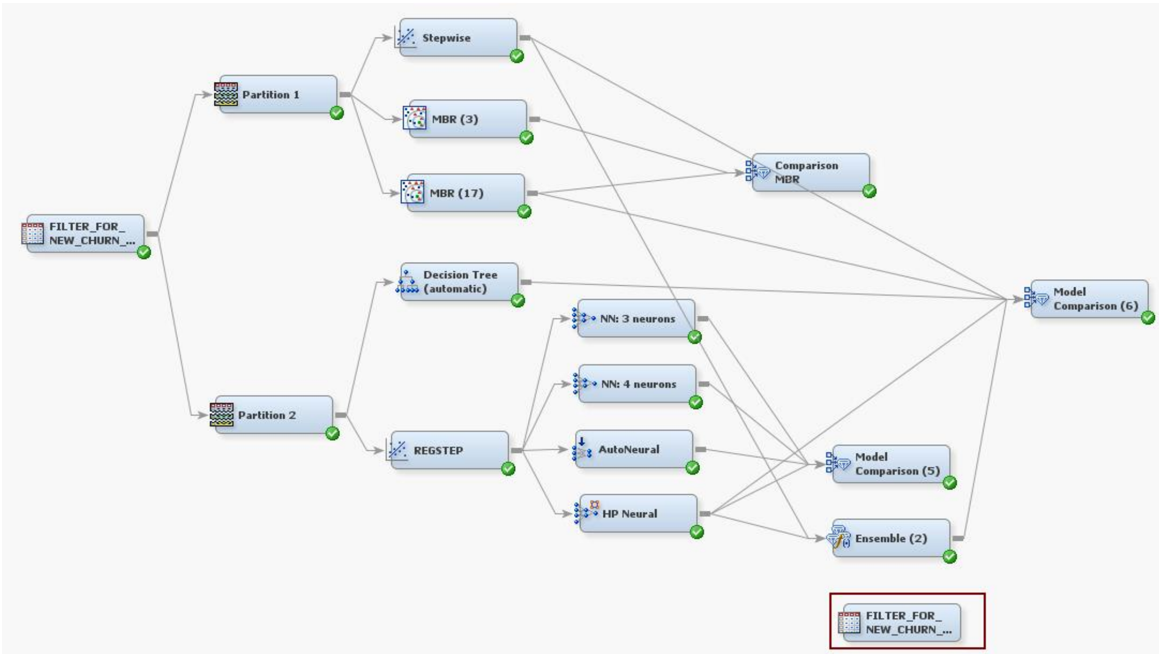
MODEL DEPLOYMENT: SCORE

10) Score the data (deploy the model).

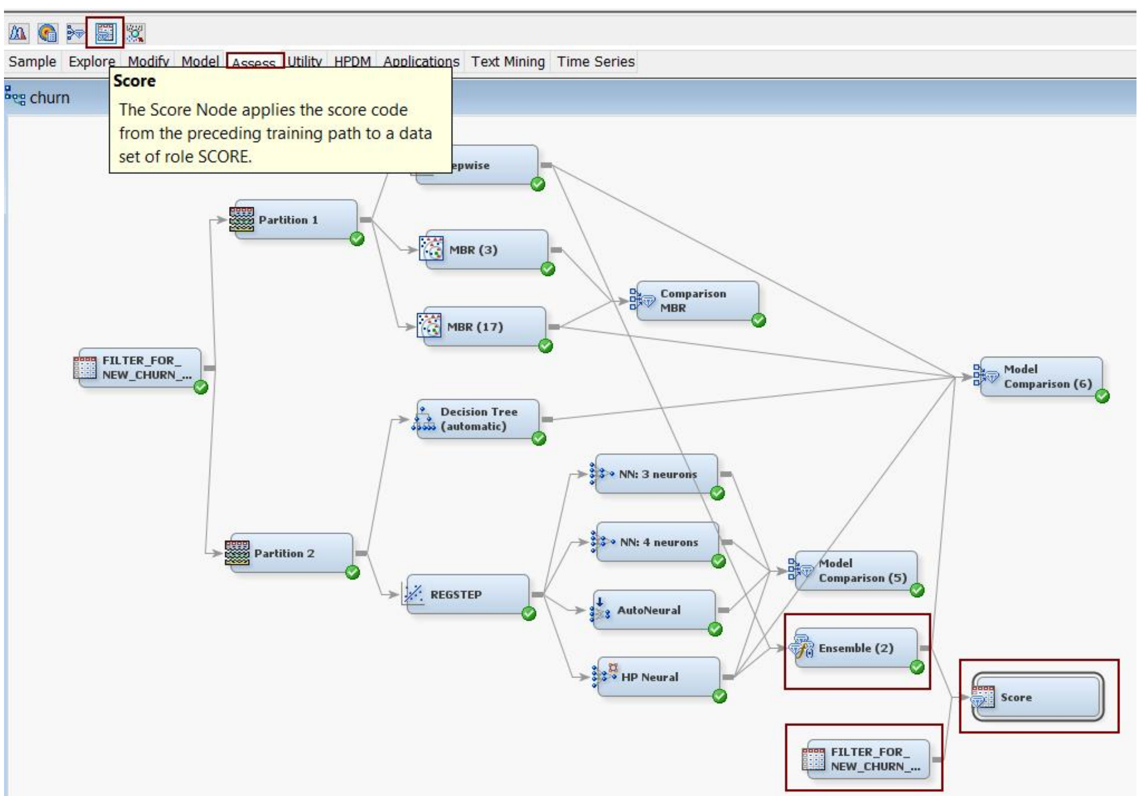
To deploy the model, you will need to add a few nodes to your diagram:

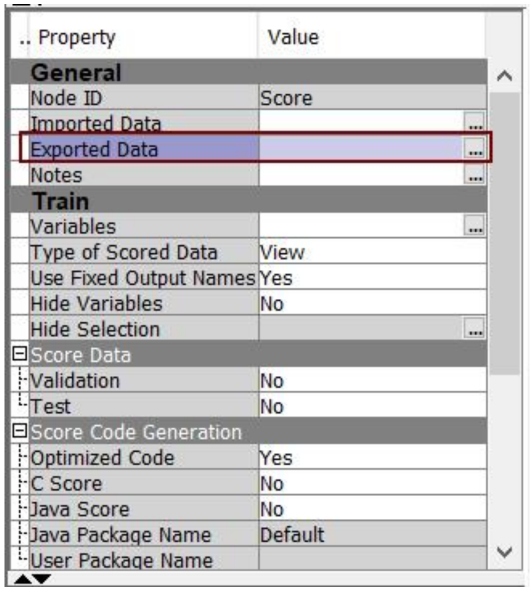
- DRAG AND DROP the dataset that we created with 1000 rows for the model deployment (filter_for_new_churn_score) into the diagram and set its role to SCORE.

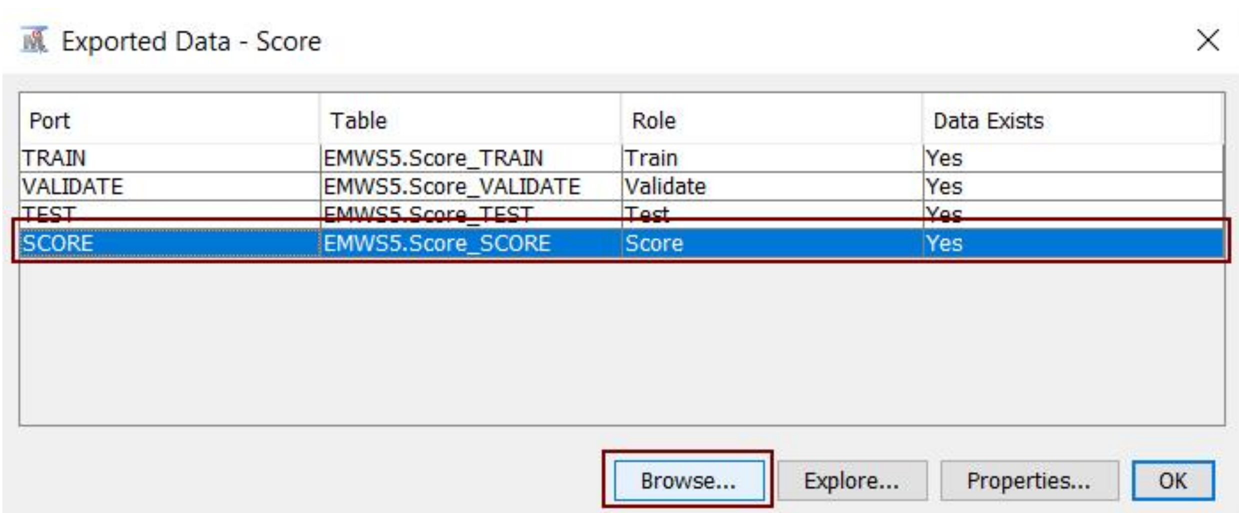


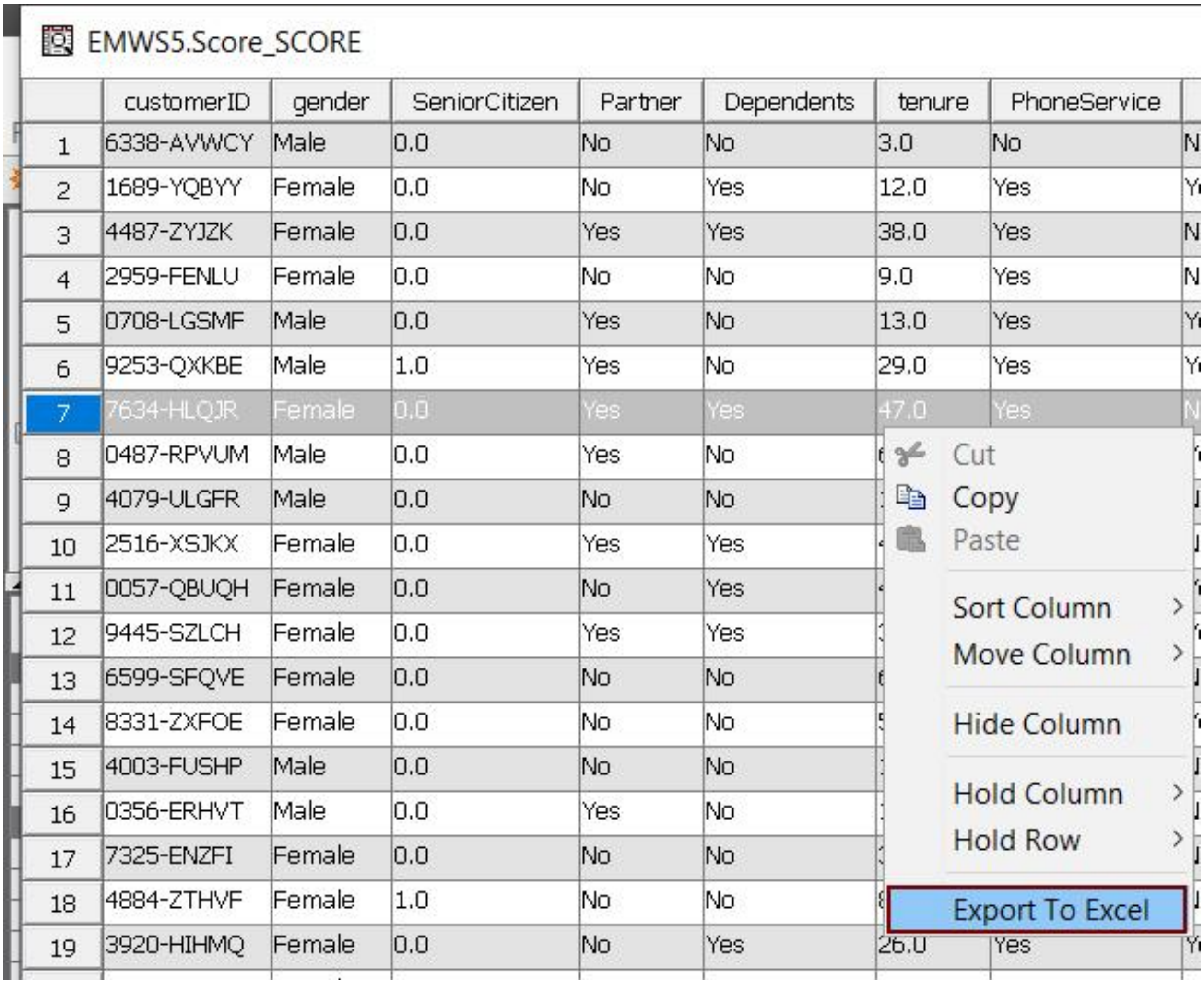


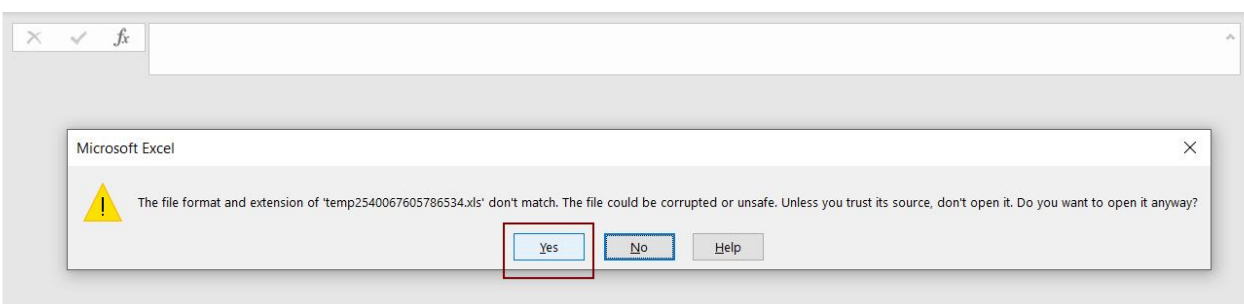
- ASSESS → SCORE → ATTACH TO THE NODE **THE BEST MODEL AND THE NEW DATASET** (you can also attach to the node the master model comparison node instead of the best model; SAS will use the model it considered best to score the data) → RUN → with the SCORE node action: GENERAL: EXPORTED DATA (to see predictions) → Click on the three dots → SCORE → BROWSE → EXPORT TO EXCEL → YES → SAVE AS → AFTER THAT: check out PREDICTION FOR CHURN

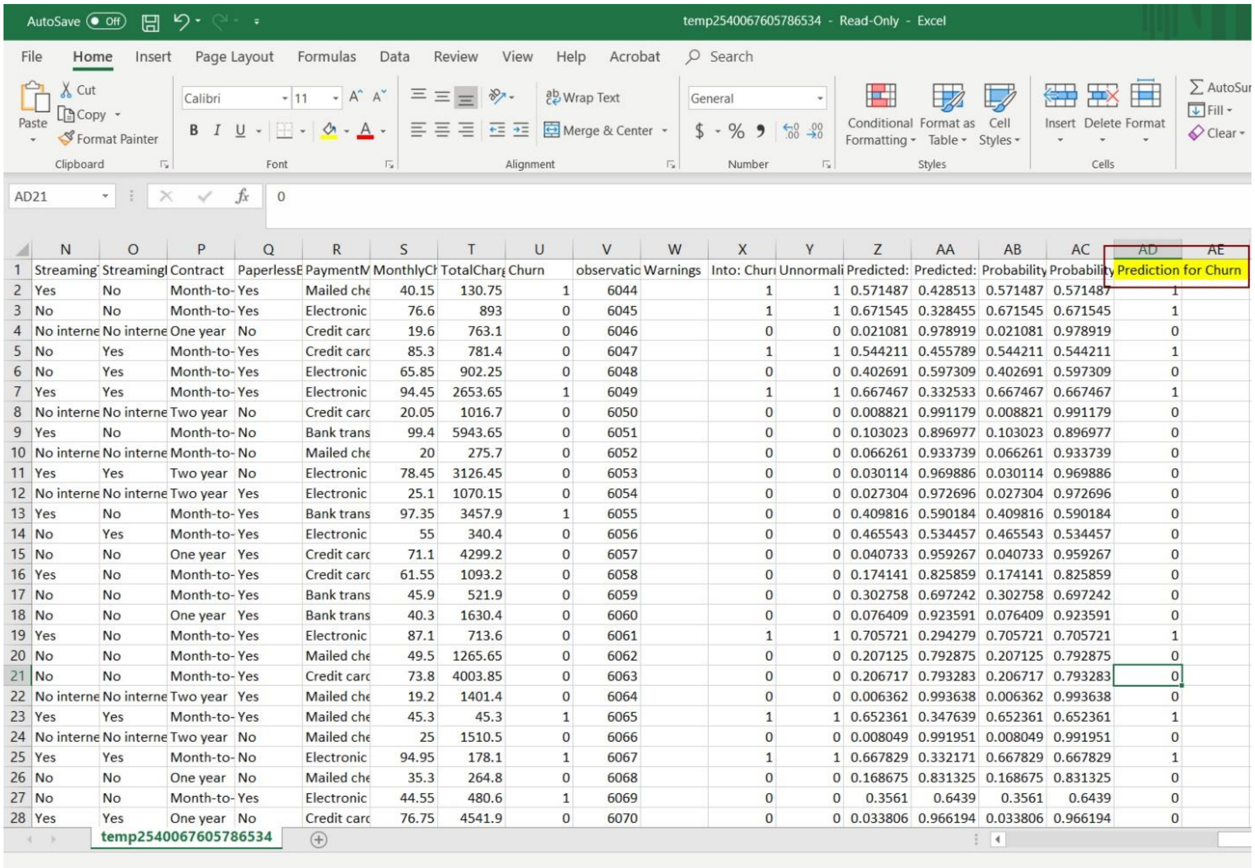



Property
Value
General
Node ID
Score
Imported Data
Exported Data
Notes
Train
Variables
Type of Scored Data
View
Use Fixed Output Names
Yes
Hide Variables
No
Hide Selection
Score Data
Validation
No
Test
No
Score Code Generation
Optimized Code
Yes
C Score
No
Java Score
No
Java Package Name
Default
User Package Name


Exported Data - Score
Port
Table
Role
Data Exists
TRAIN
EMWS5.Score_TRAIN
Train
Yes
VALIDATE
EMWS5.Score_VALIDATE
Validate
Yes
TEST
EMWS5.Score_TEST
Test
Yes
SCORE
EMWS5.Score_SCORE
Score
Yes
Browse...
Explore...
Properties...
OK


EMWS5.Score_SCORE
customerID
gender
SeniorCitizen
Partner
Dependents
tenure
PhoneService
Cut
Copy
Paste
Sort Column
Move Column
Hide Column
Hold Column
Hold Row
Export To Excel


Microsoft Excel
The file format and extension of 'temp2540067605786534.xls' don't match. The file could be corrupted or unsafe. Unless you trust its source, don't open it. Do you want to open it anyway?
Yes
No
Help



	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	AA	AB	AC	AD
1	Streaming	Streaming	Contract	PaperlessE	PaymentM	MonthlyCh	TotalCharg	Churn	observatic	Warnings	Into: Churn	Unnormali	Predicted:	Predicted:	Probability	Probability	Prediction for Churn
2	Yes	No	Month-to-	Yes	Mailed che	40.15	130.75	1	6044		1	1	0.571487	0.428513	0.571487	0.571487	1
3	No	No	Month-to-	Yes	Electronic	76.6	893	0	6045		1	1	0.671545	0.328455	0.671545	0.671545	1
4	No interne	No interne	One year	No	Credit carc	19.6	763.1	0	6046		0	0	0.021081	0.978919	0.021081	0.978919	0
5	No	Yes	Month-to-	Yes	Credit carc	85.3	781.4	0	6047		1	1	0.544211	0.455789	0.544211	0.544211	1
6	No	Yes	Month-to-	Yes	Electronic	65.85	902.25	0	6048		0	0	0.402691	0.597309	0.402691	0.597309	0
7	Yes	Yes	Month-to-	Yes	Electronic	94.45	2653.65	1	6049		1	1	0.667467	0.332533	0.667467	0.667467	1
8	No interne	No interne	Two year	No	Credit carc	20.05	1016.7	0	6050		0	0	0.008821	0.991179	0.008821	0.991179	0
9	Yes	No	Month-to-	No	Bank trans	99.4	5943.65	0	6051		0	0	0.103023	0.896977	0.103023	0.896977	0
10	No interne	No interne	Month-to-	No	Mailed che	20	275.7	0	6052		0	0	0.066261	0.933739	0.066261	0.933739	0
11	Yes	Yes	Two year	No	Electronic	78.45	3126.45	0	6053		0	0	0.030114	0.969886	0.030114	0.969886	0
12	No interne	No interne	Two year	Yes	Electronic	25.1	1070.15	0	6054		0	0	0.027304	0.972696	0.027304	0.972696	0
13	Yes	No	Month-to-	Yes	Bank trans	97.35	3457.9	1	6055		0	0	0.409816	0.590184	0.409816	0.590184	0
14	No	Yes	Month-to-	Yes	Electronic	55	340.4	0	6056		0	0	0.465543	0.534457	0.465543	0.534457	0
15	No	No	One year	Yes	Credit carc	71.1	4299.2	0	6057		0	0	0.040733	0.959267	0.040733	0.959267	0
16	Yes	No	Month-to-	Yes	Credit carc	61.55	1093.2	0	6058		0	0	0.174141	0.825859	0.174141	0.825859	0
17	No	No	Month-to-	Yes	Bank trans	45.9	521.9	0	6059		0	0	0.302758	0.697242	0.302758	0.697242	0
18	No	No	One year	Yes	Bank trans	40.3	1630.4	0	6060		0	0	0.076409	0.923591	0.076409	0.923591	0
19	Yes	No	Month-to-	Yes	Electronic	87.1	713.6	0	6061		1	1	0.705721	0.294279	0.705721	0.705721	1
20	No	No	Month-to-	Yes	Mailed che	49.5	1265.65	0	6062		0	0	0.207125	0.792875	0.207125	0.792875	0
21	No	No	Month-to-	Yes	Credit carc	73.8	4003.85	0	6063		0	0	0.206717	0.793283	0.206717	0.793283	0
22	No interne	No interne	Two year	Yes	Mailed che	19.2	1401.4	0	6064		0	0	0.006362	0.993638	0.006362	0.993638	0
23	Yes	Yes	Month-to-	Yes	Mailed che	45.3	45.3	1	6065		1	1	0.652361	0.347639	0.652361	0.652361	1
24	No interne	No interne	Two year	No	Mailed che	25	1510.5	0	6066		0	0	0.008049	0.991951	0.008049	0.991951	0
25	Yes	Yes	Month-to-	No	Electronic	94.95	178.1	1	6067		1	1	0.667829	0.332171	0.667829	0.667829	1
26	No	No	One year	No	Mailed che	35.3	264.8	0	6068		0	0	0.168675	0.831325	0.168675	0.831325	0
27	No	No	Month-to-	Yes	Electronic	44.55	480.6	1	6069		0	0	0.3561	0.6439	0.3561	0.6439	0
28	Yes	Yes	One year	No	Credit carc	76.75	4541.9	0	6070		0	0	0.033806	0.966194	0.033806	0.966194	0

You can take one step further and estimate whether your model has performed as predicted. Since this 1000-row dataset (the one you have just scored) contains the predicted churn score together with the actual churn, you can create a confusion matrix and calculate Specificity, Sensitivity, Lift, Cumulative Lift, Misclassification rate etc. To create the confusion matrix follow these steps (these are the steps for creation of a two-way table, or a crosstab):

SAS EG → IMPORTN THE EXCEL FILE YOU HAVE EXPORTED AFTER SCORING THE NEW DATASET → DESCRIBE → TABLE ANALYSIS → DATA: churn and prediction_for_churn → TABLES: drag prediction_for_churn (first) and churn (second) → CELL STATISTICS: take of the check against the column percentages (you should only have one check left: cell frequencies) → RUN

The FREQ Procedure

predicted

Table of Churn by Prediction_for_Churn

			Prediction_for_Churn(Prediction for Churn)		
			0	1	Total
actual	Churn				
	0	Frequency	TRUE NEGATIVE 669	FALSE POSITIVE 61	730
	1	Frequency	FALSE NEGATIVE 130	TRUE POSITIVE 140	270
	Total	Frequency	799	201	1000

Copy the received table into an Excel file to do various calculations from the decision matrix.