JMP® ACADEMIC CASE STUDY

# JMP030: Customer Churn

## Neural Networks with JMP® Pro

From Building Better Models With JMP® Pro, Chapter 7, SAS Press (2015). Grayson, Gardner and Stephens.

Used with permission. For additional information see https://www.jmp.com/en_us/academic/building-better-models.html

jmp® STATISTICAL DISCOVERY

# Customer Churn
Neural Networks with JMP® Pro

## Key ideas
Neural networks, activation functions, model validation, confusion matrix, lift, prediction profiler, variable importance

## Background

Customer retention is a challenge in the ultracompetitive mobile phone industry. A mobile phone company is studying factors related to customer *churn*, a term used for customers who have moved to another service provider.

## The Task

The company would like to build a model to predict which customers are most likely to move their service to a competitor. This knowledge will be used to identify customers for targeted interventions, with the ultimate goal of reducing churn.

## The Data    Churn 2 BBM.jmp

The sample data set consists of 3,332 customer records. The response variable of interest is the column called Churn, which takes two values:

| | |
|---|---|
| **True** | The customer has moved to another service provider. |
| **False** | The customer still uses “our” service. |

The potential predictors are primarily related to service use and account. A high-level summary of all of the variables, produced with the Columns Viewer, is shown in Exhibit 1.

(To generate this output, use Cols > Columns Viewer. Select all the variables, check the Show Quartiles box, and click Show Summary. To deselect (un-highlight) the variables, click Clear Select.)

## Analysis

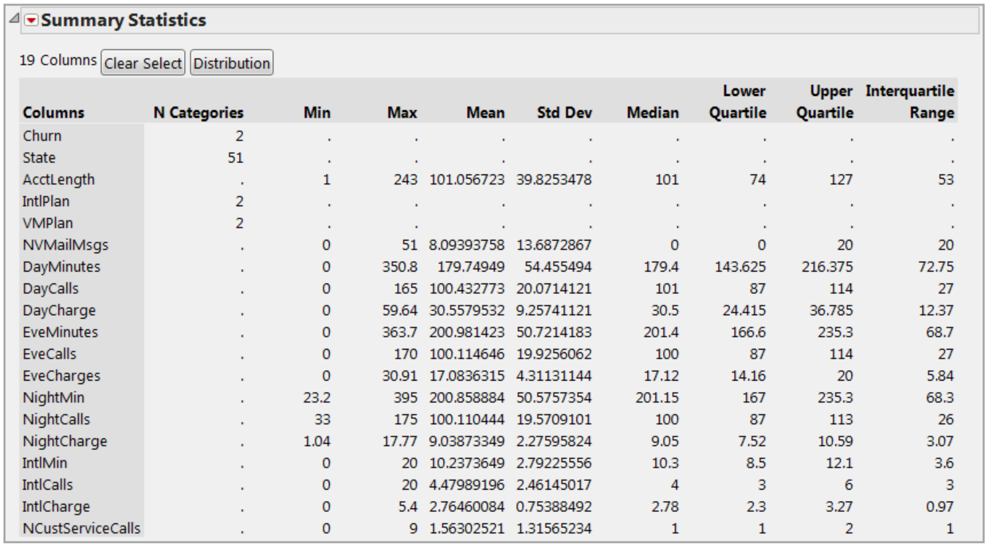
We start as we always do, by getting to know our data, and take the appropriate steps for preparing our data for modeling.[1] In Exhibit 1, we see that there are three categorical (State, IntlPlan and VMPlan) and 15 continuous factors that can be used as predictors. There are no missing values (if any variable is missing values, we’d see a column N Missing under Summary Statistics).

The statistics for the continuous variables provide insights into the centering, shape and spread of the distributions. For example, the mean of NVMailMsgs (the number of voice mail messages) is around eight, but the median is zero. This is an indication that the distribution is right-skewed.

Our response variable, **Churn**, is a two-level categorical variable. In Exhibit 2, we see that 14.5 percent of the data is for those who have “churned.”

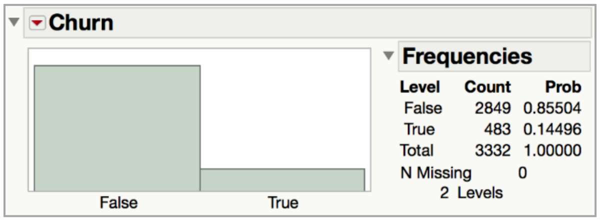
[1] We don’t address data quality or data preparation in this case study, and only briefly discuss tools for data exploration. For more information, see *Building Better Models With JMP Pro, Chapter 3 – Working With Data*.

**Exhibit 1** Churn Data Summary



| Columns | N Categories | Min | Max | Mean | Std Dev | Median | Lower Quartile | Upper Quartile | Interquartile Range |
|---|---|---|---|---|---|---|---|---|---|
| Churn | 2 | . | . | . | . | . | . | . | . |
| State | 51 | . | . | . | . | . | . | . | . |
| AcctLength | . | 1 | 243 | 101.056723 | 39.8253478 | 101 | 74 | 127 | 53 |
| IntlPlan | 2 | . | . | . | . | . | . | . | . |
| VMPlan | 2 | . | . | . | . | . | . | . | . |
| NVMailMsgs | . | 0 | 51 | 8.09393758 | 13.6872867 | 0 | 0 | 20 | 20 |
| DayMinutes | . | 0 | 350.8 | 179.74949 | 54.455494 | 179.4 | 143.625 | 216.375 | 72.75 |
| DayCalls | . | 0 | 165 | 100.432773 | 20.0714121 | 101 | 87 | 114 | 27 |
| DayCharge | . | 0 | 59.64 | 30.5579532 | 9.25741121 | 30.5 | 24.415 | 36.785 | 12.37 |
| EveMinutes | . | 0 | 363.7 | 200.981423 | 50.7214183 | 201.4 | 166.6 | 235.3 | 68.7 |
| EveCalls | . | 0 | 170 | 100.114646 | 19.9256062 | 100 | 87 | 114 | 27 |
| EveCharges | . | 0 | 30.91 | 17.0836315 | 4.31131144 | 17.12 | 14.16 | 20 | 5.84 |
| NightMin | . | 23.2 | 395 | 200.858884 | 50.5757354 | 201.15 | 167 | 235.3 | 68.3 |
| NightCalls | . | 33 | 175 | 100.110444 | 19.5709101 | 100 | 87 | 113 | 26 |
| NightCharge | . | 1.04 | 17.77 | 9.03873349 | 2.27595824 | 9.05 | 7.52 | 10.59 | 3.07 |
| IntlMin | . | 0 | 20 | 10.2373649 | 2.79225556 | 10.3 | 8.5 | 12.1 | 3.6 |
| IntlCalls | . | 0 | 20 | 4.47989196 | 2.46145017 | 4 | 3 | 6 | 3 |
| IntlCharge | . | 0 | 5.4 | 2.76460084 | 0.75388492 | 2.78 | 2.3 | 3.27 | 0.97 |
| NCustServiceCalls | . | 0 | 9 | 1.56302521 | 1.31565234 | 1 | 1 | 2 | 1 |

**Exhibit 2** The Distribution of Churn



| Level | Count | Prob |
|---|---|---|
| False | 2849 | 0.85504 |
| True | 483 | 0.14496 |
| Total | 3332 | 1.00000 |

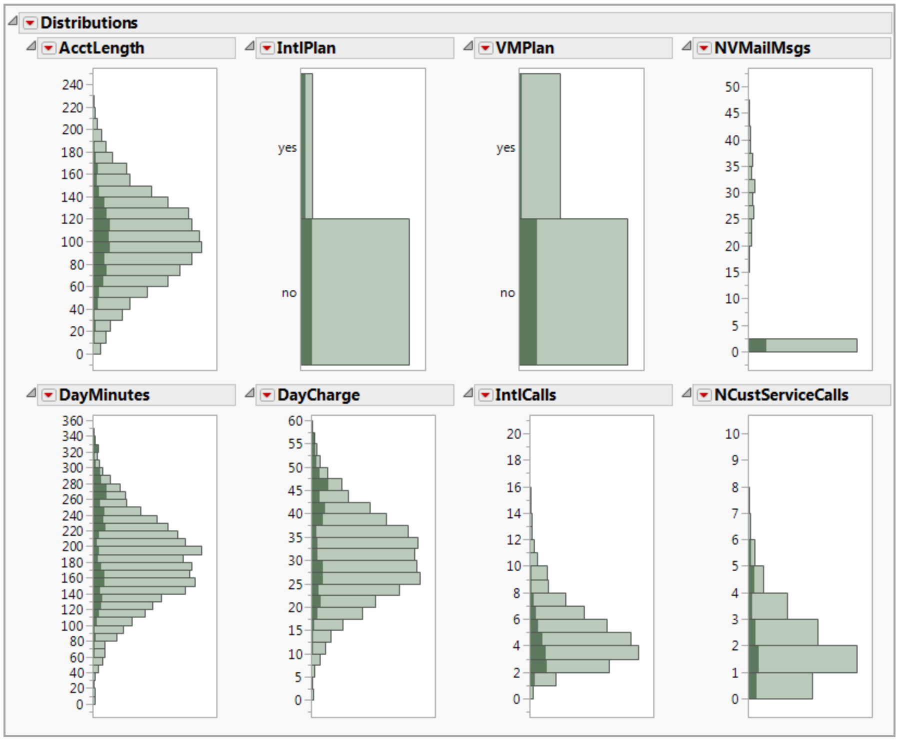
N Missing 0
2 Levels

The distributions of some of the potential predictors are shown in Exhibit 3. All of the rows where Churn=True are selected in the data table (click on the bar for True in the distribution of Churn to select these rows). A potentially good predictor variable is one in which the shaded regions in the histogram cluster in one or more regions of the graph.

The length of time someone has been a customer (AcctLength) does not seem to be related to churn (the distribution for the shaded region, in terms of centering, shape and spread, is similar to the overall distribution). Those customers who have churned tend to have higher-than-average daytime minute usage and daytime charges, very few voicemail messages, and lower-than-average international calls, and some have more customer service calls than average.

The type of plans that are associated with the account (International - IntlPlan or Voicemail - VMPlan) look like they may have some relationship with churn, but this is difficult to see from the distributions view.

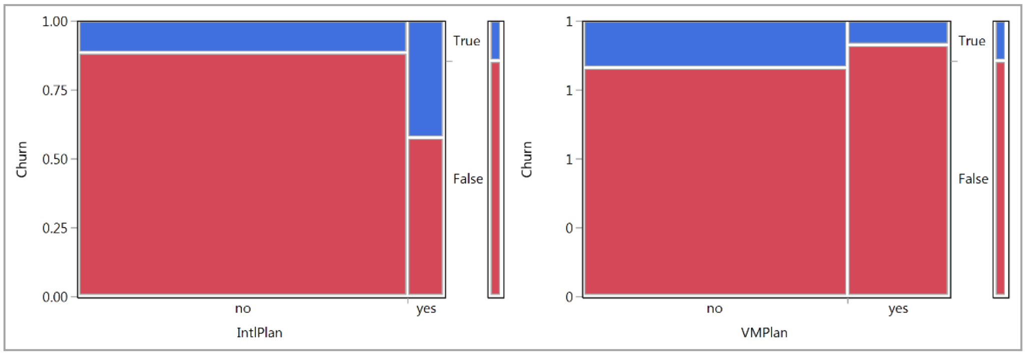
**Exhibit 3** Distributions of the Potential Predictors



We can examine the relationship between categorical predictors and churn with a Mosaic Plot using Analyze > Fit Y by X (or Graph > Graph Builder)**.** In Fit Y by X, select Churn as Y, Response and the categorical predictors as X, Factor, and click OK.

The plots of Churn versus IntlPlan and VMPlan are shown in Exhibit 4. Examining these graphs shows that customers with an international plan or without a voicemail plan appear to be more likely to churn. A contingency table and other statistics (not shown) are also provided.

**Exhibit 4** Churn versus Intlplan and VMPlan

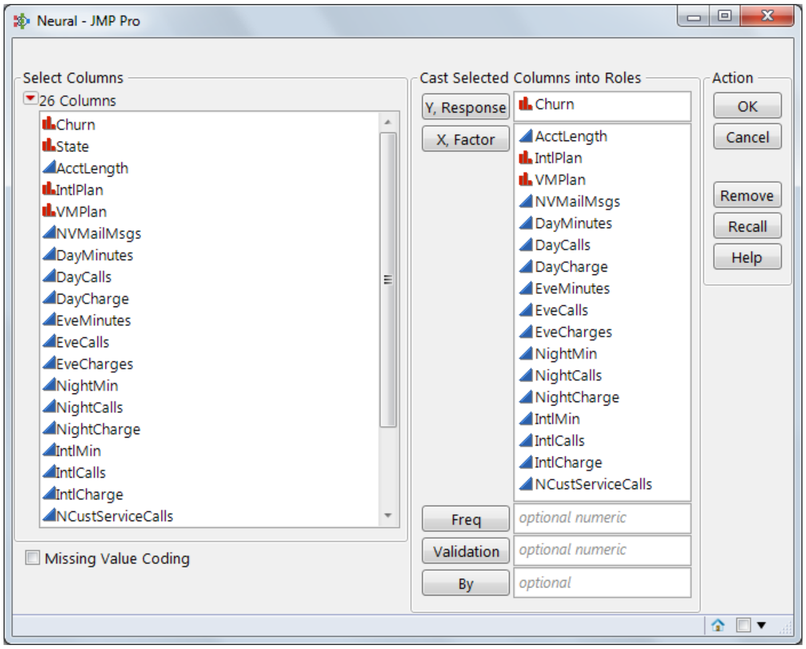


## Modeling

After exploring our variables and gaining an understanding of potential relationships (note that this was only partially done in the previous section), we fit a neural network in JMP Pro using Analyze > Modeling > Neural (see Exhibit 5). In this example, we omit the variable State and focus on predictors related to call and plan information.

Since we are not missing values from any of the variables, we leave the Missing Value Coding box unchecked. When this option is selected, *informative missing coding* of missing values is applied (see JMP Help for details on this option).

**Exhibit 5** Neural Dialog Window



Neural - JMP Pro

Y, Response: Churn

X, Factor: AcctLength, IntlPlan, VMPlan, NVMailMsgs, DayMinutes, DayCalls, DayCharge, EveMinutes, EveCalls, EveCharges, NightMin, NightCalls, NightCharge, IntlMin, IntlCalls, IntlCharge, NCustServiceCalls

Freq: optional numeric

Validation: optional numeric

By: optional

□ Missing Value Coding

The resulting Model Launch dialog (shown in Exhibit 6) lets us specify the structure of the JMP Pro neural network model and other fitting options (note that fewer fitting options are available in the standard version of JMP).

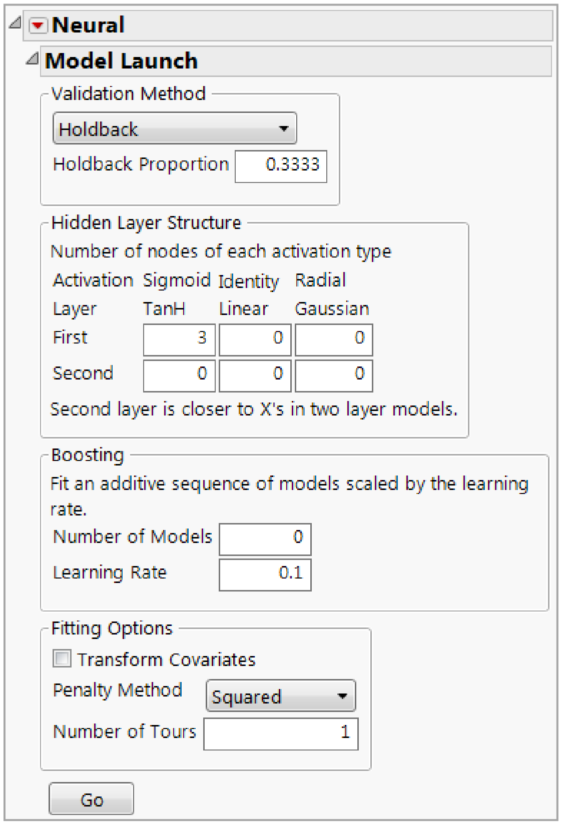
**About Neural Networks**

A neural network is a very flexible algorithm that can model complex relationships between inputs and outputs. Each neural network has an input layer, one or more hidden layers, and an output layer. Each hidden layer has one or more *nodes*. In each node in the hidden layer, a linear combination of the input variables is transformed. The transformation that is applied is denoted with a symbol that indicates the type of transformation.

In JMP Pro, there are three types of transformation functions that can be used in a neural network model: *TanH*, *Linear*, and *Gaussian*. These are also referred to as *activation functions*. In the standard version of JMP, only the *TanH* function is available.

- *TanH* is the hyperbolic tangent function, which is similar in shape to the logistic function used for logistic regression models.
- *Linear* is similar to constructing a linear regression model. The linear combination of predictor variables is not transformed.
- *Gaussian* is a bell-shaped function, which is similar to the normal distribution density function.

**Exhibit 6** Specifying the neural network model



Neural

Model Launch

Validation Method

Holdback

Holdback Proportion 0.3333

Hidden Layer Structure

Number of nodes of each activation type

| Activation<br>Layer | Sigmoid<br>TanH | Identity<br>Linear | Radial<br>Gaussian |
|---|---|---|---|
| First | 3 | 0 | 0 |
| Second | 0 | 0 | 0 |

Second layer is closer to X's in two layer models.

Boosting

Fit an additive sequence of models scaled by the learning rate.

Number of Models 0

Learning Rate 0.1

Fitting Options

□ Transform Covariates

Penalty Method Squared

Number of Tours 1

We use the default model that has a single hidden layer with three nodes, each with *TanH* activation functions.

We also use the default Holdback Validation Method**,** using a third of the data as the holdback portion. Rows used for the holdback sample can be saved as a column in the data table (select Save Validation from the red triangle for the fitted model).

Note: The Neural platform in JMP, unlike other modeling platforms, requires some form of model validation to aid in the model-building process and to help prevent overfitting. The basic idea behind validation (or cross-validation) is to hold a subset of the data out of the model-building process. This process forms two partitions of the data, a *training set* and a *validation set* (note that a third set, or *test set*, can also be used.) The model is built using the *training* set, while the holdout *validation* set is then used to see how well the model performs and to aid in model selection.

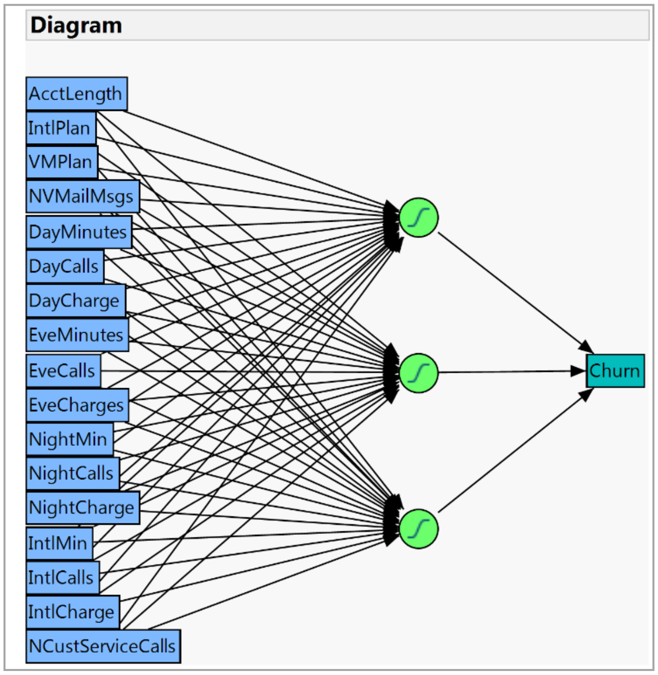
JMP provides a number of holdback methods for validation in the Neural platform. The default is a one-third *holdback proportion*. In JMP Pro, a *Validation* column, with rows assigned to training and validation sets, is recommended. To use this method, the validation column is entered into the Validation field in the Neural launch dialog (see Exhibit 5.)

For more information on model validation in neural networks and creating a validation column, see *Building Better Models With JMP Pro* Chapters 7 and 8, or search for “validation” in JMP Help.

## The Neural Model and Results

After running the model (click Go at the bottom – see Exhibit 6), we can display model structure (select **Diagram** from the red triangle for the model). We see input variables mapping to each of the activation functions in the hidden layer, and nodes in the hidden layer mapping to the output layer (Exhibit 7). The s-shaped curve in each of the nodes in the hidden layer indicates that the *TanH* activation function was used.

**Exhibit 7** Diagram for the Fitted Neural Network Model



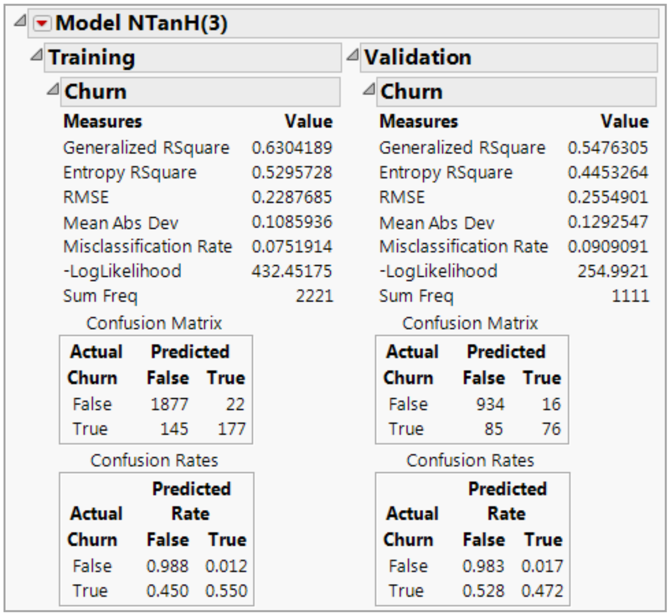
Model results for both the training and validation sets are shown in Exhibit 8[2].

The response variable (**Churn)** for this model is categorical. As we have seen with logistic regression and classification trees, the confusion matrix and overall misclassification rate provide indications of the predictive ability of our model. Since the validation set was not directly used in estimating the model parameters, it provides a less biased assessment of model performance.

The misclassification rate for the validation data is 9.09 percent. Examining the confusion matrix, we see that for customers who actually churned, 47.2 percent of the time the model correctly predicted that they would churn.

[2] Since a random holdout sample is used, your model results may be slightly different. To obtain the same results shown in Exhibit 8, set the random seed to 1,000 prior to launching the Neural platform. The random seed can be set with the Random Seed Reset add-in, which can be installed from the JMP File Exchange at community.jmp.com/docs/DOC-6601. Note that Windows and Mac may produce different results using the same random seed.

**Exhibit 8** Churn, Neural Network Model Results



Model NTanH(3)

Training

Churn

| Measures | Value |
|---|---|
| Generalized RSquare | 0.6304189 |
| Entropy RSquare | 0.5295728 |
| RMSE | 0.2287685 |
| Mean Abs Dev | 0.1085936 |
| Misclassification Rate | 0.0751914 |
| -LogLikelihood | 432.45175 |
| Sum Freq | 2221 |

Confusion Matrix

| Actual | Predicted | |
|---|---|---|
| Churn | False | True |
| False | 1877 | 22 |
| True | 145 | 177 |

Confusion Rates

| | Predicted | |
|---|---|---|
| Actual | Rate | |
| Churn | False | True |
| False | 0.988 | 0.012 |
| True | 0.450 | 0.550 |

Validation

Churn

| Measures | Value |
|---|---|
| Generalized RSquare | 0.5476305 |
| Entropy RSquare | 0.4453264 |
| RMSE | 0.2554901 |
| Mean Abs Dev | 0.1292547 |
| Misclassification Rate | 0.0909091 |
| -LogLikelihood | 254.9921 |
| Sum Freq | 1111 |

Confusion Matrix

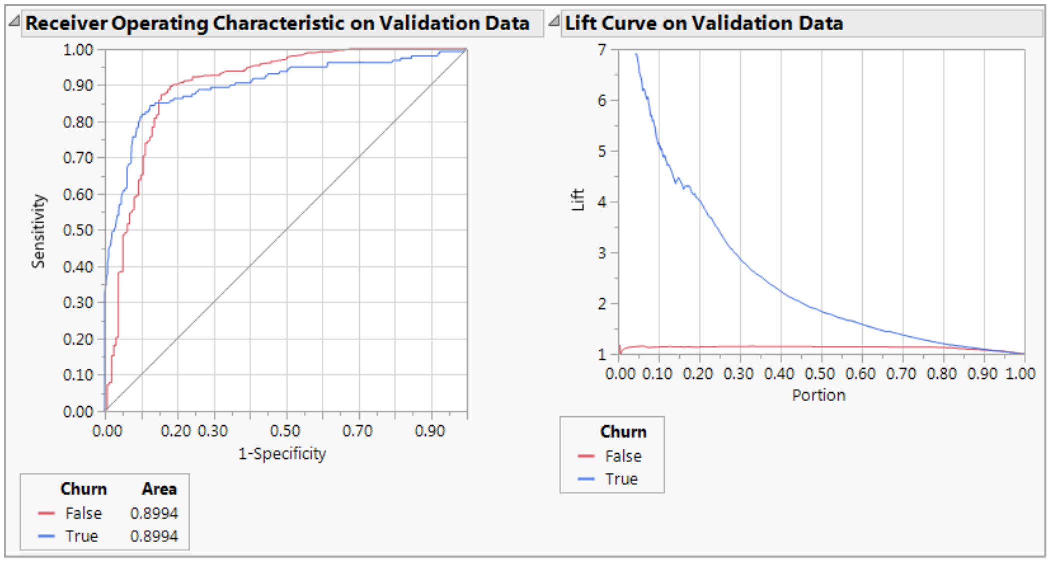
| Actual | Predicted | |
|---|---|---|
| Churn | False | True |
| False | 934 | 16 |
| True | 85 | 76 |

Confusion Rates

| | Predicted | |
|---|---|---|
| Actual | Rate | |
| Churn | False | True |
| False | 0.983 | 0.017 |
| True | 0.528 | 0.472 |

ROC and lift curves (available from the model red triangle) provide additional information about the predictive ability of our model. For example, from the lift curve in Exhibit 9, we see that for rows that are in the top 20 percent of the sorted probability of Churn=True (Portion = 0.20), there are roughly four times more churners than we would expect if we drew 20 percent of customers at random.[3]

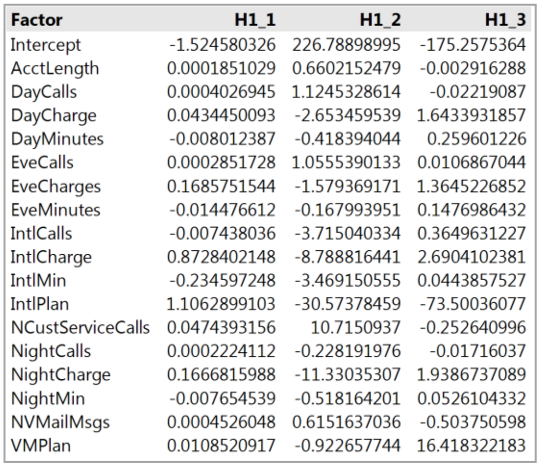
**Exhibit 9** ROC, and Lift Curves for the Neural Network Model



Neural networks have many parameters that must be estimated when building models. To view these parameter estimates, select **Show Estimates** from the red triangle for the model. For this particular model, there are 58 parameters (54 for hidden nodes and four for the output prediction equation). These estimates were saved to a JMP data table and rearranged into the format shown in Exhibit 10.

[3] Lift is a comparison of the churn rate for a given portion of the data when the data are sorted in order of the predicted probability, compared to the churn rate for the entire population (see Building Better Models With JMP Pro, Chapter 6 for details).

**Exhibit 10** Churn, Parameter Estimates for Neural Network Hidden Layer Nodes



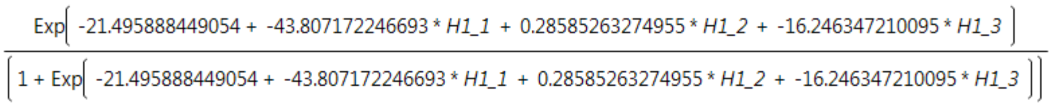
| Factor | H1_1 | H1_2 | H1_3 |
|---|---|---|---|
| Intercept | -1.524580326 | 226.78898995 | -175.2575364 |
| AcctLength | 0.0001851029 | 0.6602152479 | -0.002916288 |
| DayCalls | 0.0004026945 | 1.1245328614 | -0.02219087 |
| DayCharge | 0.0434450093 | -2.653459539 | 1.6433931857 |
| DayMinutes | -0.008012387 | -0.418394044 | 0.259601226 |
| EveCalls | 0.0002851728 | 1.0555390133 | 0.0106867044 |
| EveCharges | 0.1685751544 | -1.579369171 | 1.3645226852 |
| EveMinutes | -0.014476612 | -0.167993951 | 0.1476986432 |
| IntlCalls | -0.007438036 | -3.715040334 | 0.3649631227 |
| IntlCharge | 0.8728402148 | -8.788816441 | 2.6904102381 |
| IntlMin | -0.234597248 | -3.469150555 | 0.0443857527 |
| IntlPlan | 1.1062899103 | -30.57378459 | -73.50036077 |
| NCustServiceCalls | 0.0474393156 | 10.7150937 | -0.252640996 |
| NightCalls | 0.0002224112 | -0.228191976 | -0.01716037 |
| NightCharge | 0.1666815988 | -11.33035307 | 1.9386737089 |
| NightMin | -0.007654539 | -0.518164201 | 0.0526104332 |
| NVMailMsgs | 0.0004526048 | 0.6151637036 | -0.503750598 |
| VMPlan | 0.0108520917 | -0.922657744 | 16.418322183 |

Parameter estimates can also be seen when the formulas are saved to the data table (from the red triangle for the model, select Save Formulas).

For this example, six columns will be saved to the data table:

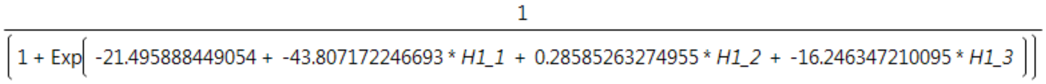
- Probability(Churn=False) and Probability(Churn=True): These formulas, which compute the predicted probabilities for Churn, are shown in Exhibits 11 and 12.
- Formulas for the three hidden layer nodes (H1_1, H1_2, and H1_3). These formulas contain the parameter estimates displayed in Exhibit 10.
- Most Likely Churn: This makes a classification based on which output category (Churn=True or Churn=False) has the largest predicted probability.

**Exhibit 11** Probability(Churn=False) Formula



$$\frac{\mathrm{Exp}\left(-21.495888449054 + -43.807172246693 * H1\_1 + 0.28585263274955 * H1\_2 + -16.246347210095 * H1\_3\right)}{\left(1 + \mathrm{Exp}\left(-21.495888449054 + -43.807172246693 * H1\_1 + 0.28585263274955 * H1\_2 + -16.246347210095 * H1\_3\right)\right)}$$

**Exhibit 12** Probability(Churn=True) Formula



$$\frac{1}{\left(1 + \mathrm{Exp}\left(-21.495888449054 + -43.807172246693 * H1\_1 + 0.28585263274955 * H1\_2 + -16.246347210095 * H1\_3\right)\right)}$$

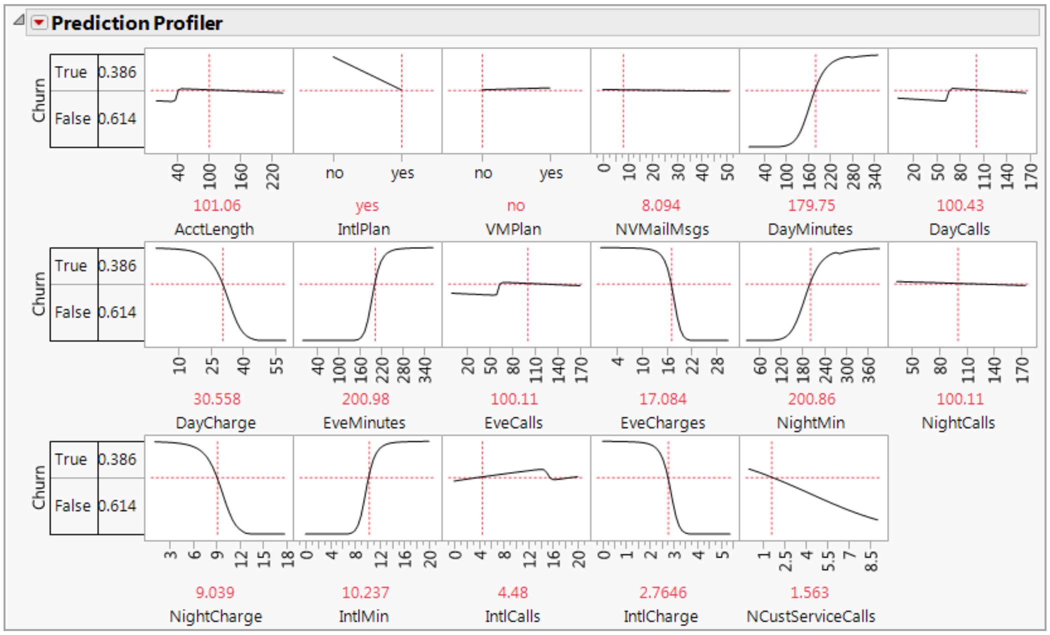
Examining these formulas provides an understanding of the complexity of neural network models and demonstrates why they can be difficult to interpret. However, predicted probabilities can be used to better understand the conditions leading to churn. One way to do this is to look at groups of customers with the highest and lowest probability of churn. This examination can then be used to establish *profiles* of churners and non-churners, and can help guide marketing campaigns and other customer retention initiatives (Linoff and Berry, 2011).

Another way to develop a better understanding of our model is to use the Categorical Profiler (from the red triangle for the model). The profiler can be used to make predictions as well as to explore how the predicted probability of churn changes as the values of various predictors change. In Exhibit 13, we use the Arrange in Rows **option** (from the red triangle next to Prediction Profiler**)** to display profiles for all of predictors on one screen.

A quick exploration of the model using the profiler leads to some important insights:

- For customers with international calling plans (drag the line for IntlPlan from no to yes), the probability of churning increases as the usage minutes for day, night, evening, and international calls decreases, and also as the charges for those types of calls increase.
- There appears to be a threshold for each of these factors where churning becomes very likely.
- The more service calls a customer has had seems to slowly increase the probability of churning.
- Factors such as the number of calls, account length, voicemail features, and voicemail usage do not seem to be strongly related to the probability of churning.

**Exhibit 13** Churn, Categorical Profiler



### Identifying Important Variables

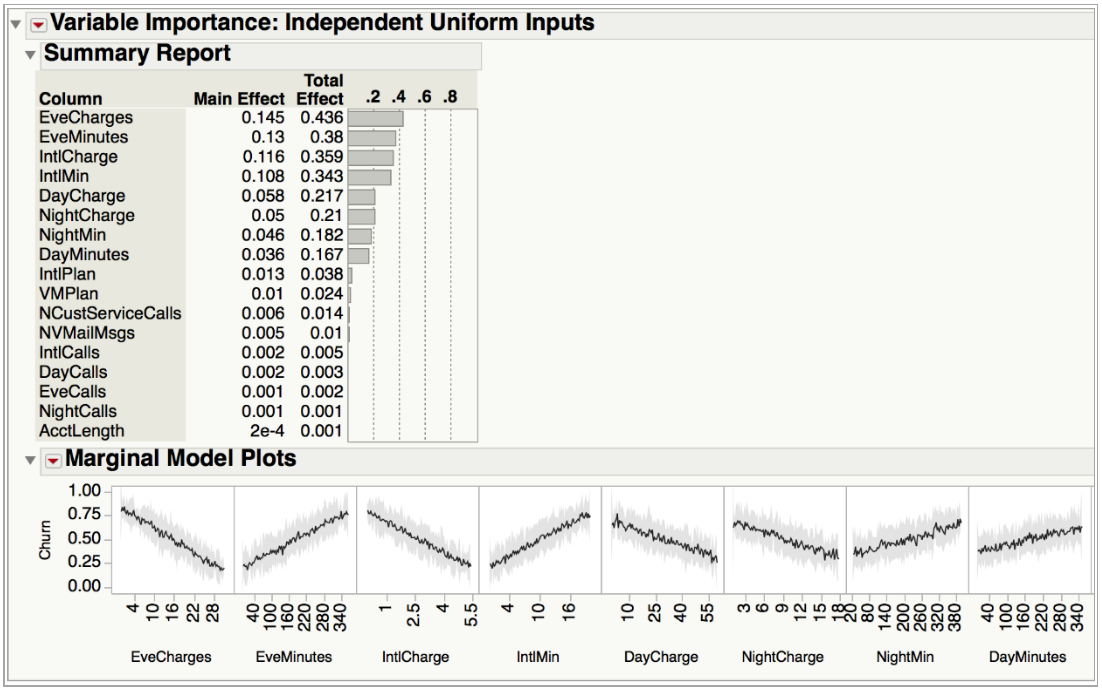
To help us sort through the large set of potentially important variables, we use the Variable Importance option. This feature is available under the red triangle for the Prediction Profiler (select Assess Variable Importance > Independent Uniform Inputs). Variable Importance uses a simulation to estimate the sensitivity of a response to changes in each of the predictor variables (see JMP Help for details)[4].

The Variable Importance Summary Report provides information similar to the column contributions report in the Partition platform. We see that the four most important variables are EveCharges, EveMinutes, IntlCharges and IntlMin (Exhibit 15) and that several of the predictors are not very useful in predicting Churn.

Marginal Model Plots show factors, ordered by total effect in the Variable Importance Summary Report (see bottom, Exhibit 15). Contours in the plot show the average response for each predictor across the distributions of the other predictors. We can see, for example, that higher values of EveMinutes and IntlMin result in higher churn rates.

[4] Since this analysis is based on a simulation, your results may be slightly different.

**Exhibit 15** Variable Importance and Marginal Model Plot



Variable Importance: Independent Uniform Inputs

Summary Report

| Column | Main Effect | Total Effect |
|---|---|---|
| EveCharges | 0.145 | 0.436 |
| EveMinutes | 0.13 | 0.38 |
| IntlCharge | 0.116 | 0.359 |
| IntlMin | 0.108 | 0.343 |
| DayCharge | 0.058 | 0.217 |
| NightCharge | 0.05 | 0.21 |
| NightMin | 0.046 | 0.182 |
| DayMinutes | 0.036 | 0.167 |
| IntlPlan | 0.013 | 0.038 |
| VMPlan | 0.01 | 0.024 |
| NCustServiceCalls | 0.006 | 0.014 |
| NVMailMsgs | 0.005 | 0.01 |
| IntlCalls | 0.002 | 0.005 |
| DayCalls | 0.002 | 0.003 |
| EveCalls | 0.001 | 0.002 |
| NightCalls | 0.001 | 0.001 |
| AcctLength | 2e-4 | 0.001 |

Marginal Model Plots

## Summary

### Statistical Insights

The goal of this study was to build a predictive model for customer churn. We developed a simple neural network, with one hidden layer and three nodes, each using the *TanH* function, and achieved a misclassification rate for the validation set of 9.1 percent. We used only one relatively simple neural network model; however, performance might be improved with a more complicated neural model. We explore other more complex neural models in an exercise.

### Implications

Since the target variable in this case study is categorical, we could also have used other modeling methods, such as logistic regression and classification trees. If we fit multiple models, we need to be able to easily compare the predictive performance of these models. This can be done using the Analyze > Model Comparison platform. (See JMP Help, or refer to *Building Better Models*, Chapter 8 for information on how to compare competing models and identify the "best" model.)

We also did not concern ourselves with variable reduction (choosing only the most important predictors to include in the model) as we generally do with regression analysis. Neural models allow us to use a large number of input variables, even if some of these input variables have a high degree of multicollinearity or have little influence on the response.

### JMP Features and Hints

We used Distribution and Fit Y by X platforms to become familiar with our data. Note that we did not do an exhaustive exploration of the data, nor did we discuss data quality or preparation. The importance of this modeling pre-work cannot be underestimated.

We used the **Neural** platform to fit a model using the default settings. A one-third random holdback sample was used for model validation. The confusion matrix was used to assess model accuracy, the

**Categorical Profiler** was used to graphically explore the model, and the **Variable Importance** option was selected to identify the most important variables.

## Exercises

**Exercise 1**: Use the Churn 2 BBM.jmp data set for this exercise. In the example, we fit a neural network with one hidden layer with three nodes, using only the *TanH* activation function.

a. Fit the model described in this case study.
b. Fit a model with one hidden layer and three nodes, using the *TanH*, *Linear* and *Gaussian* activation functions. How does this model compare to the original model in terms of misclassification?
c. Fit a final model with two hidden layers and several nodes, using the activation functions of your choice. How does this model compare to the two single-layer models fit in part (a) and part (b)?
d. For the original model (fit in this case study), what is the lift at portion = 0.2? What is the lift at portion = 0.10? Interpret lift.
e. Recall that this team is tasked with identifying customers most likely to churn, with the goal of developing strategies or interventions to minimize the risk of churn. How can this model be used toward this goal?

**Exercise 2**: Use the Boston Housing.jmp data set, from the JMP Sample Data Library under the Help menu, for this exercise. Fit a neural network with the continuous variable mvalue as the response and the other variables as the predictors. Use the default settings.

a. What is the validation RSquare and the RMSE?
b. Display the actual by predicted plots. Are there any unusual patterns or observations?
c. Use the prediction profiler to explore relationships between the predictors and the response. Which combination of factor settings leads to the highest predicted mvalue (median value)?
d. Open the variable importance report. Which variables are most important? Describe the nature of the relationship between the top three variables and the response (i.e., use the prediction profiler to explore what happens to the predicted response as the value of each predictor is changed)?