Paper AA-12

# Predictive Modeling Using Artificial Neural Networks in SAS® Enterprise Miner

Kechen Zhao, Department of Preventive Medicine, University of Southern California

## Abstract

A neural network is a set of connected input/output variables where each connection has a weight with it. Neural networks take non-linear functions of linear combinations of input variables. This is a powerful and very general approach for regression and classification, and has been shown to be the best machine learning method on many problems. Neural networks are especially effective in problems with high-noise ratio and settings where prediction without interpretation is the goal. It is nowadays one of the most popular data mining and pattern learning tools frequently used by many companies like Google. SAS Enterprise Miner implements tools for modeling and utilizing neural networks. However, literatures on neural network modeling using SAS Enterprise Miner are limited. This paper is a step-by-step introduction to neural network modeling using SAS Enterprise Miner 13.2. This paper will provide an introduction to the Neural Network node in SAS Enterprise Miner 13.2. It will also address steps in training a neural network, accessing model fit and utilizing a trained neural network to classify outcomes in SAS Enterprise Miner 13.2.

## Introduction

Neural networks are a class of parametric models that can capture a variety of nonlinear relationships between a set of predictors and a target variable than can a traditional logistic regression model. Building a neural network involves two main stages. First, one must define the network configuration or structure; and then iteratively trains the model based on the given network structure.

A neural network is more complicated to explain than a regression or a decision tree. However, we know that we prefer a stronger predictive model, even if the model is more complicated. Here, we will run a neural network model and compare it to a logistic model, a decision tree and gradient boosting in this paper.

Neural networks are very flexible and SAS Enterprise Miner has two nodes that fit neural network models: the Neural Network node and the AutoNeural node. The Neural Network node trains a pre-specified neural network configuration. This node is preferred when you know a lot about the subject knowledge of the model that you want to define. The AutoNeural node searches over several network configurations and finds one that best captures the relationship in a data set and then trains the model. The example in this paper will use the Neural Network node.

## Data Source

The example data we will use in this example is called “donor_raw_data” and can be downloaded from SAS website: http://support.sas.com/documentation/onlinedoc/miner/. The data set includes 40 quantitative input variables, 8 categorical input variables, one binary outcome variable and one quantitative outcome variable. Details about these variables can be found from the URL: http://support.sas.com/documentation/cdl/en/emgsj/61207/HTML/default/viewer.htm#p0gbbwhw4eszv5n1rko3o6x34b3g.htm. In this example, we will only use the binary outcome variable TARGET_B.

**Exploratory Data Analysis (EDA)**

It is always a good idea to first explore the statistical properties of the variables in the data set before model building. This helps us learn the nature of the data and detect any abnormalities in the data. Running the **StatExplore** node returns basic descriptive statistics for each variable (min/max, median, mean, etc.), information about missing values and the top 20 input variables by their chi-squared statistics.

In predictive modeling, a method for assessing the generalization of the trained model is to partition the data source. A portion of the data, called the training data set, is used for building the preliminary model. The validation data set is used to prevent a model node from overfitting the training data and to compare the models. Run the **Data Partition** node to allocate 55% of the data source to the training data set and the rest to the validation data set.

In this example data set, the variables SES and URBANICITY are input variables for which the value “?” denotes a missing value. DONOR_GENDER has the value “A”. This value is the result of a data entry error. Use the **Replacement** node to replace values “?” and “A” with missing value.

In SAS Enterprise Miner, models such as regressions and neural networks drop observations altogether that contain missing values, which reduces the size of the training data set. Less training data can affect the prediction accuracy of the trained model. To overcome this drawback, we can impute missing values before we fit the models. We can use the **Impute** node to impute the missing values. Within the **Impute** node, select Tree Surrogate and Median as the Default Input Method for categorical variables and quantitative variable, respectively.

Sometimes, transformed input data is more informative. Variables transformation can be used to stabilize variance, remove nonlinearity, improve additivity, and remedy non-normality. Hence, transformations of the input data can improve model fit. We use **Transform Variables** node to transform some of the original input variables. Specifically, select Log-10 form for the following variables:

- FILE_AVG_GIFT
- LAST_GIFT_AMT
- LIFETIME_AVG_GIFT_AMT

- LIFETIME_GIFT_AMOUNT

and select Optimal Binning form for the following input variables:

- LIFETIME_CARD_PROM
- LIFETIME_GIFT_COUNT
- MEDIAN_HOME_VALUE
- MEDIAN_HOUSEHOLD_INCOME
- PER_CAPITA_INCOME
- RECENT_RESPONSE_PROP
- RECENT_STAR_STATUS

### Predictive Modeling Using Neural Networks

Before creating a neural network, we reduce the number of input variables with the **Variable Selection** node. Performing variable selection reduces the number of input variables and cuts down the computational time for training a neural network. We then use the **Neural Network** node to train a neural network. In the **Neural Network** node, select Direct Connection and set Number of Hidden Units to 5. This example trains a multilayer perceptron neural network with 5 units on the hidden layer. The network has connections directly between the inputs and outputs in addition to connections via the hidden units.

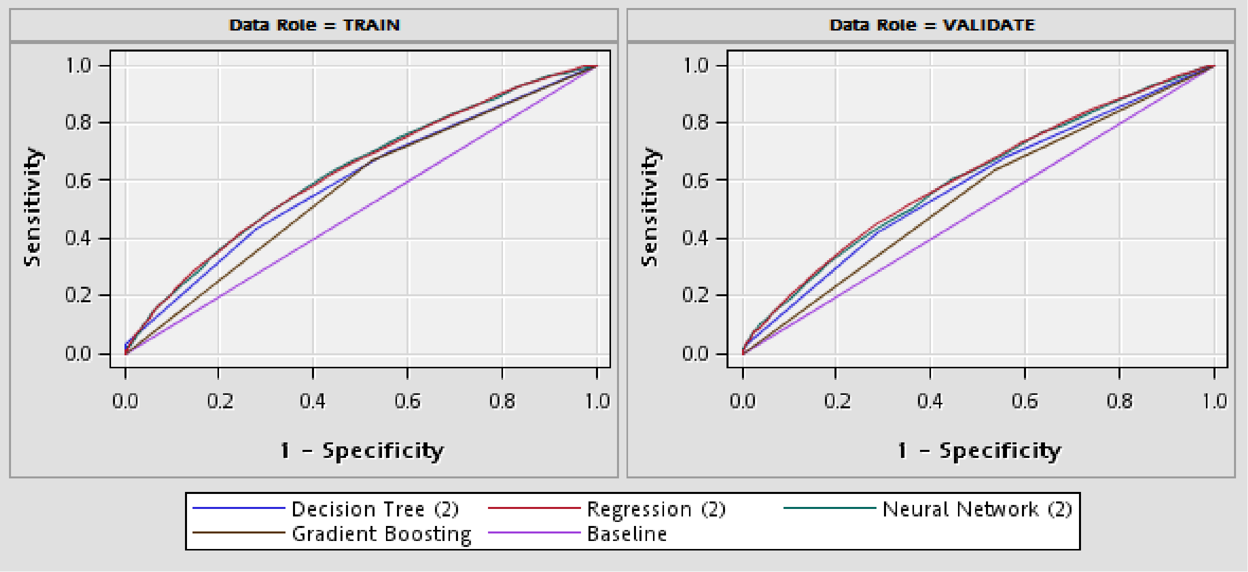

**Fig. 1** – ROC charts from training and validate data sets

Fig. 1 displays the ROC curves obtained from 4 different models for training and validation data sets, respectively. In the training and validation data sets, Regression (2) and Neural Network (2) are slightly better than Decision Tree (2) and Gradient Boosting.

### Score New Data

The final step in most predictive modeling problems is to create scoring code that you can use to score new data. To do this, we can combine the preferred model with the **Score** node to compute predicted value of the outcome variable.

## Conclusions

This paper provides an introduction to neural network modeling using SAS Enterprise Miner 13.2. The paper compares the performance of a neural network to other commonly used predictive models. It addresses steps in training a neural network, accessing model fit and utilizing a trained neural network to classify outcomes in SAS Enterprise Miner 13.2.

## References

1. Kevin P. Murphy (2012), Machine Learning: A Probabilistic Perspective, MIT Press.
2. Trevor Hastie, Robert Tibshirani and Jerome Friedman (2013), The Elements of Statistical Learning, Springer.
3. SAS (2013), Getting Started with SAS Enterprise Miner 13.1.

## Acknowledgments

The author would like to thank Cindy Lee for her encouragement and support.

## Contact Information

Your comments and questions are valued and encouraged. Contact the author at:

Name: Kechen Zhao
Enterprise: University of Southern California,
Department of Preventive Medicine,
Division of Biostatistics.
Address: 2001 N. Soto Street, Los Angeles, CA 90032
Mailbox #117
Phone: 510-584-1950
E-mail: zhao_kechen@hotmail.com