Print

Demo: Implementing Social Networks

This demonstration illustrates how to build social network classifiers for credit card fraud detection.

Here you can see the CreditCardFraud data set, which I imported into SAS Enterprise Miner. Let’s explore this data set. It has 6684 observations with 11 variables representing the transaction amount; whether the transaction was made in the European Union or Belgium; the recency, frequency, and monetary values of past transactions; two risk category indicators; and the network variable MerchantRelDegree, which is the relative number of past fraudulent transactions associated with the merchant. This is the target fraud indicator, which is 1 for fraudulent transactions and 0 otherwise. You can see that the data contains only a few fraudulent cases resulting in a very skewed class distribution, as anticipated.

Let’s now create a new diagram, name it SNA, and drag and drop the CreditCardFraud data set to the diagram workspace. We add a Data Partition node and connect it to the CreditCardFraud data set. We set the Partitioning Method property to Stratified to make sure that the fraud and no-fraud odds are the same in training, validation, and test data.

The purpose of this demonstration is to contrast the performance of a logistic regression model using only local variables with the performance of a logistic regression model using all variables, including the MerchantRelDegree social network variable.

We add a Metadata node to the diagram workspace and connect it to the Data Partition node. We click the button next to the Train property and set the new role of the MerchantRelDegree variable to Rejected. We now add a Regression node and connect it to the Metadata node. We rename the Regression node to Local Model.

We add another Regression node and connect it to the Data Partition node. We rename it to Network Model.

We are now ready to compare the Local Model with the Network Model. We add a Model Comparison node to the diagram workspace and connect it to both Regression nodes. We set the Selection Statistic property to ROC (or receiver operating characteristic curve) and the Selection Table property to Validation. This will make sure that the model with the highest ROC on the validation data is selected. We run the node and inspect the results.

You can see that SAS has selected the Network Model. You can see that the ROC curve of the Network Model slightly dominates the ROC curve of the Local Model on the validation data. The difference on the test data is more pronounced. You can also see that the lift curve of the Network Model is above the lift curve of the Local Model on the training, validation, and test data. This clearly illustrates the performance benefit of including network variables in the model. Note that the regression model can easily be replaced with other models as well, such as decision trees or neural networks.

Close