Paper 2685-2018

To Show or Not to Show? Using SAS® Solutions to Maximize Patient Scheduling In Medical Clinics

Scott Koval and Mia Lyst, Pinnacle Solutions, Inc.;
Yijie Li, Microsoft Corp.

ABSTRACT

At busy medical clinics, it is important to maximize the number of patients seen each day. Scheduled patients that fail to show are a common problem in medical clinics today that cause a decrease in revenue and disruption in daily operations. In order to combat this issue at several ambulatory clinics in the Southwest, machine learning algorithms were used to help the staff make informed decisions when scheduling patients. Using historical data, predictive models were built to identify patients that are not likely to show up to an appointment. By identifying these patients with high confidence, the staff can quickly fill these open slots to help the clinics serve more people at an efficient pace.

A powerful end-to-end SAS® solution was implemented where SAS® Office Analytics was used to prepare the historical data for modeling and score new data based on the predictive models, SAS® Enterprise Miner™ was used to build, modify and validate the machine learning algorithms, and SAS® Visual Analytics™ provided the mechanism to automatically load scored data into memory and populate informative dashboards.

Medical staff is able to review the reports while scheduling patients and track the overall performance of the models. By using this solution, these clinics are able to serve more patients and produce additional revenue that would otherwise be lost. Due to the success of this project in the initial set of clinics, the solution is going to be expanded to other specialty outpatient clinics across the region.

INTRODUCTION

Managing no-show patients in a medical clinic is an everyday occurrence that results in a potential decrease in patient satisfaction, a drop in the level of patient care, and a disruption of employee schedules. According to an October 3, 2017 MGMA Stat poll, 44% of the respondents indicated that patient no-shows are the biggest challenge with appointments in their medical practice (Harrop, 2017). The bottom line is that no-show patients cost the clinics valuable time and money.

Overbooking patients is a common way that medical clinics try to improve their no-show rates. Although there is a slight reduction in no-shows in the short-term, the long-term effects have been shown to negatively impact overall patient experience and clinic operations.

Instead of “blindly” overbooking patients to try and reduce no-show rates, one of the top pediatric hospitals in the country decided to use advanced analytics to identify patients that have the largest probability of not showing up to their appointment based on past behavior. Provided with this information, their medical clinics can target those patients with intervening actions or messages to try to improve outcomes (e.g. send text message or a reminder call 1-2 days prior to their appointment). In addition, they can selectively overbook only those time slots where the patient(s) is predicted as a no-show if they should also decide to pursue overbooking.

This paper will describe how SAS software provides a turn-key solution to reduce the number of no-show patients for specialty clinics through the use of predictive algorithms, informative dashboards, and performance monitoring tools.

METHODOLOGY

The three main SAS applications used in this project are SAS Office Analytics, SAS Enterprise Miner and SAS Visual Analytics (see Figure 1). The drag-and-drop capabilities of all of these tools help to ensure the client that their internal statisticians could support and maintain the final solution with limited training. Therefore as much of the out-of-box features as possible are utilized in all applications, without compromising performance, to enable ease of solution maintainability.

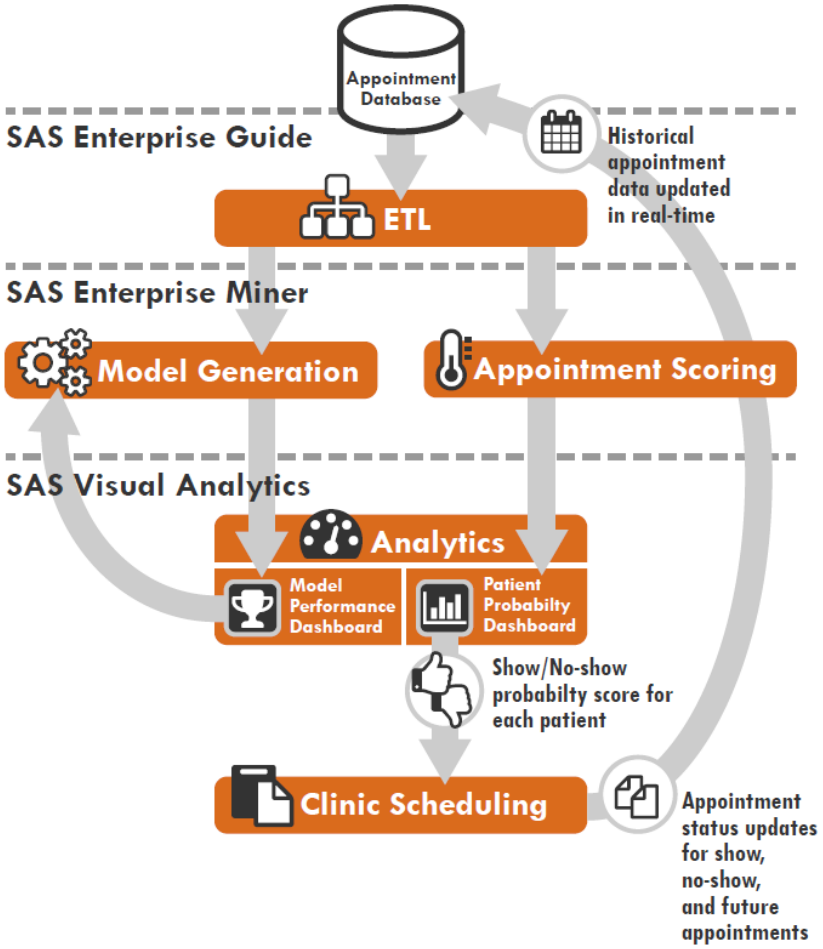


Figure 1. Architecture of Proposed Solution

SAS Enterprise Guide performs all ETL processing required to prepare the historical data for modeling as well as scoring new appointments. Data from the hospital database and other 3rd party data sources are used as input into the ETL processes. SAS Enterprise Miner trains, optimizes and validates all of the predictive models used for the no-show predictions. The resulting SAS score code calculates the probability of a patient no-show for each model. Finally, the scored results and additional appointment information (clinic, patient IDs, appointment times, etc.) are displayed in SAS Visual Analytic dashboards. The dashboards are updated on a daily basis as new data is scored every morning.

MODEL BUILDING

Figure 2 summarizes the proposed model building process. SAS was used to interact with the hospital database and pull in clinic data. After the data is prepared, it is loaded into SAS EM for developers to produce models. During this development phase, feedback by Subject-Matter Experts (SMEs) helped guide the results of the modeling. After testing and review, a champion model is selected and is used to score incoming patient data.

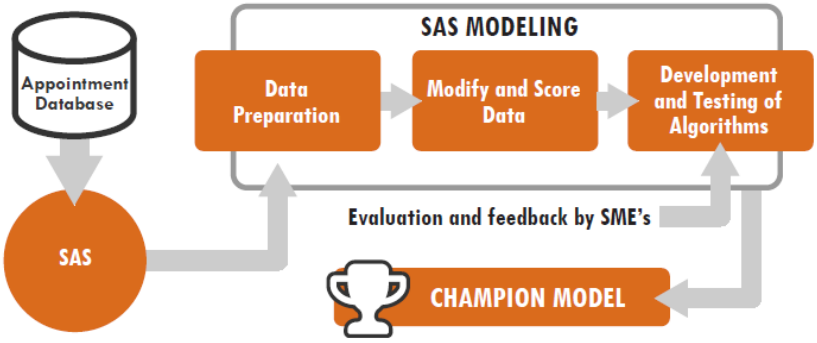


Figure 2. Model Building Process

DATA PREPARATION

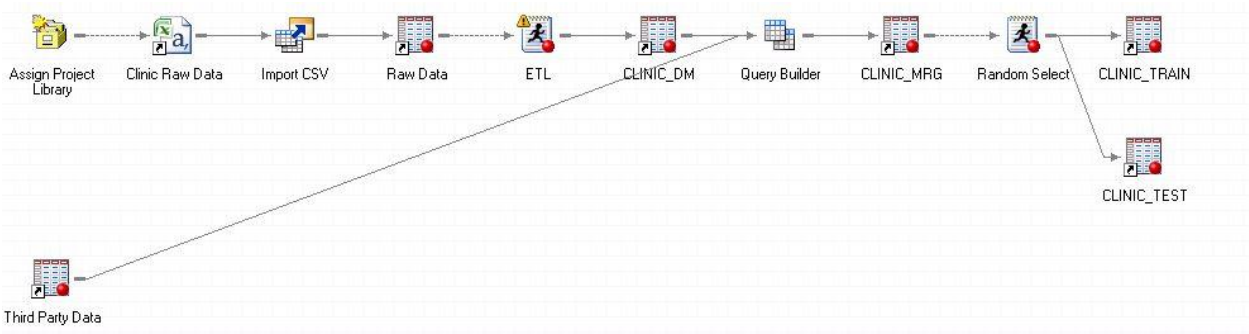
The hospital provided the raw data in the format of one CSV file per clinic that contained approximately one year of historical patient appointment data. Important information on the patient's demographics, scheduled appointment, and diagnosis history is included in this data. Overall, this included 27 categorical variables, 9 numeric variables and 6 date variables. After exploring the data, it was discovered that there were large amounts of categories in some of the predictor variables. Another problem encountered was learning about process changes in each clinic. Because of this, some data needed to be filtered out.

In order to resolve the problem of having a large number of categories, modelers worked with SMEs to reclassify some of the data. For example, there were many different types of insurance. After some discussion, this data was recoded into the following categories: government, charity, self-pay and commercial. Another instance of this problem included patient visit codes. There were many different categories containing information on why the patient made their appointment. Once again, SMEs assisted in simplifying these into two groups: new patients and follow-ups.

New variables were also created based on the data provided. For instance, a variable based on the distance a patient must travel to arrive at the clinic was calculated using address information. Data on a patient's history of no-shows and cancellations was provided, but these numbers were revealed to be inaccurate. New rates were calculated using the DATA step on sorted data. The retain statement and lag function were used to help create more recent and accurate moving averages of known appointment outcomes. Example code of this process is as follows:

```
data clinic_dm;
    set clinic_dm;
    by id appt_date_time;
    retain NoShow_Sum Cancel_Sum;
    if first.id then do;
          Count = 1;
          NoShow_Sum = NoShow;
          Cancel_Sum = Cancel;
    end;
    else do;
          Count+1;
          NoShow_Sum+NoShow;
          Cancel_Sum+Cancel;
    end;
    NoShow_Avg = NoShow_Sum / Count;
    NoShow_Avg_Lag = lag1(NoShow_Avg);
    Cancel_Avg = Cancel_Sum / count;
    Cancel_Avg_Lag = lag1(Cancel_Avg);
    if first.id then do;
          noshow_avg_lag = 0;
          cancel_avg_lag = 0;
    end;
    drop noshow_sum cancel_sum noshow_avg cancel_avg;
run;
```

In order to possibly help improve model performance, the data was also enriched by additional third party sources. These additional variables provided information based on the scheduled patient's zip code. After this data was merged, modelers now had access to surrounding demographic information, such as income, population, and nearby healthcare facilities. A final step in the process was to randomly split the data into a training dataset (80% of the data) and a test dataset (20%). The test dataset was withheld from any of the modeling process and used as a baseline to measure modeling performance. Display 1 provides a screen capture of a final process flow developed for a clinic.



Display 1. Enterprise Guide Process Flow for ETL

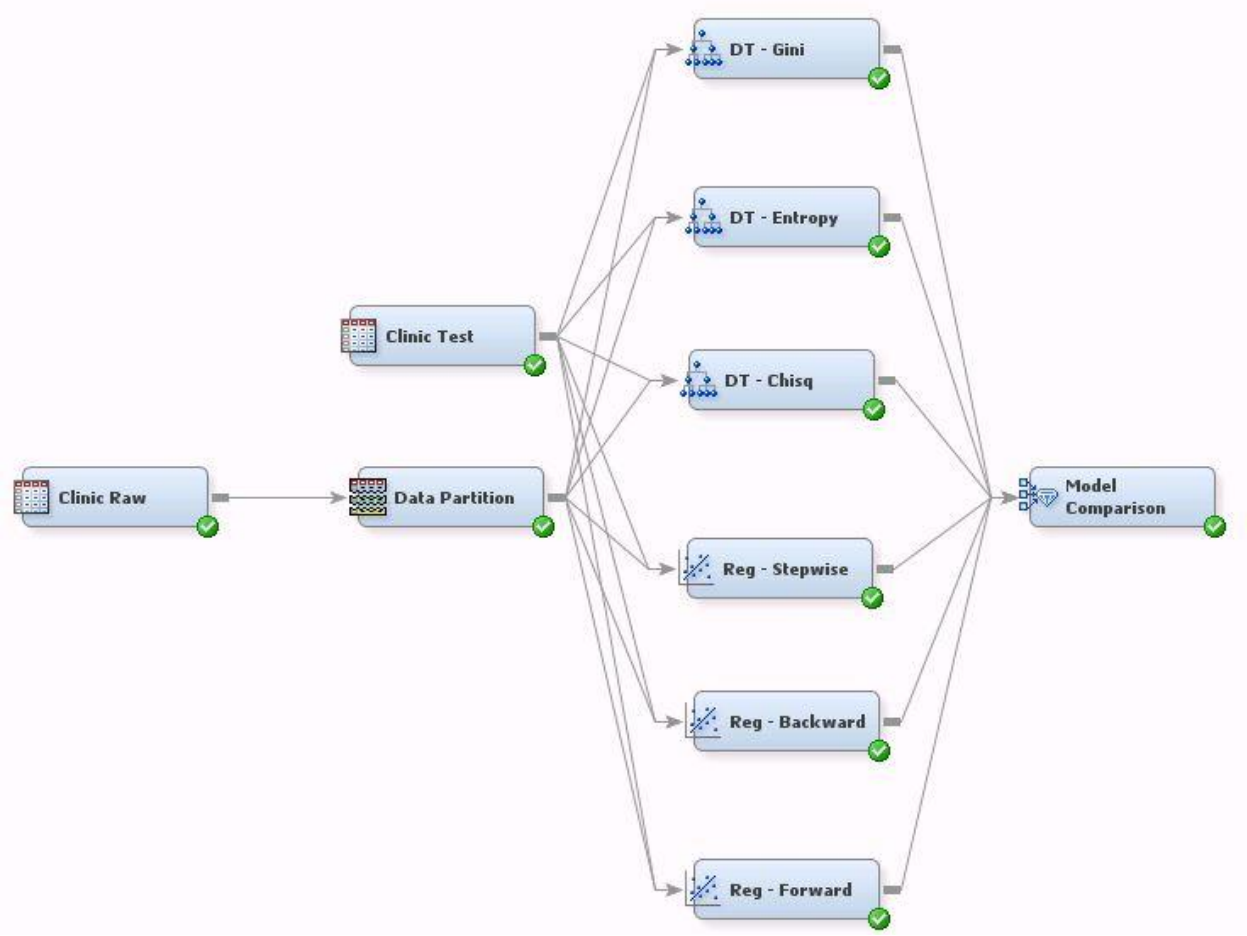
MODELING

Models for this project were built using SAS Enterprise Miner. This solution offers a wide variety of modeling techniques and was utilized to help overcome some of the challenges associated with this project. The first main challenge present was to make sure that the results are accurate enough to minimize the instance of double booking schedules only to have two patients show up. For this, a goal of 80% precision rate was set as a minimum for models developed. The second main challenge was that models must be able to identify what can be rare events at certain clinics. Not all of the clinics that models were developed for had similar no show rates. Some of them had rates that were less than 10%. In order

to help overcome this, separate models were built for each individual hospital clinic. Finally, the models created have to be able to hold up over time. Changes in behavior of both patients and clinic personnel can vary over time. Any models put into production must be robust enough to account for these variations.

After the data was prepared and process flows finalized, the resulting tables were loaded into SAS EM to build models. Separate projects were created for predicting no show rates at each individual clinic. The first step to model building was to randomly partition the raw data into training and validation datasets. In this case, 70% of the data was used to train models and the remaining 30% was used to validate them. This helped ensure that the models created were robust enough to classify new and unknown records.

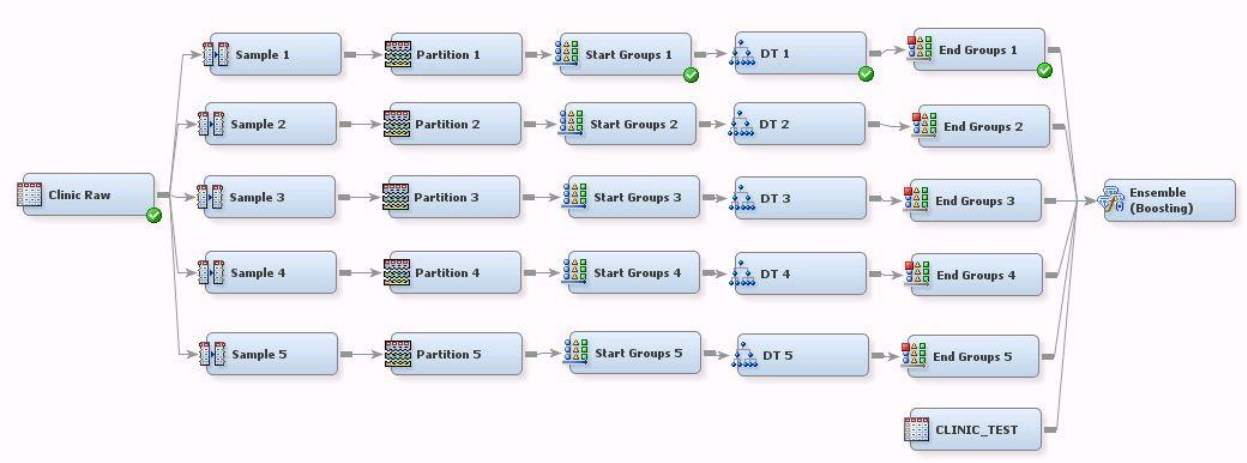
The initial phase towards model development relied heavily on traditional modeling techniques. This included creating models based off logistic regression and decision trees. Display 2 provides a screenshot of the original project diagram.



Display 2. SAS EM Diagram of Initial Modeling

Multiple models were constructed using each of these methods. For logistic regressions, variable selection algorithms differed between each of the models. This would include the forward, backward, and stepwise variable selection techniques. The decision tree models differed in parameters relating to tree width, depth, and evaluating splitting rules. All of these models were then compared using the model comparison node. Champion models were selected by which ones provided the best results on the data withheld for testing purposes.

An additional technique was developed in order to further improve model performance on the test data (See Display 3). This method involved including a boosting technique in order to adjust model outcomes over various iterations (see Wexler & Easterling, 2015). A series of sampling nodes, each with unique seeds, were used to create new training datasets. For each sample, a Start Groups node was used with the mode parameter set to Boosting. A Decision Tree node was then placed afterwards, and the parameters adjusted to be similar to prior trees. Finally, the End Groups node was connected to each modeling path to close off the loop and allow for the iterations (n = 10) to take place. This series of models was then averaged into an ensemble and compared with the rest.



Display 3. Ensemble Model Averaging Boosted Decision Trees

MODEL COMPARISONS

The model performance was measured using the Model Comparison node in SAS EM. The diagnostic statistic used to select a champion model was the misclassification rate on the withheld test data. The results can be found in Table 1.

		Train		Valid		Test	
Champion	Model	Misc.	AUC	Misc.	AUC	Misc.	AUC
Y	Ensemble (Boosting)	2.00%	99.60%	5.35%	97.10%	7.80%	80.10%
	DT - ChiSq	9.44%	58.00%	9.79%	57.10%	8.69%	58.10%
	DT - Gini	9.33%	72.70%	9.65%	68.20%	8.82%	69.20%
	DT - Entropy	9.41%	71.30%	9.74%	65.80%	8.88%	69.30%
	Reg - Forward	10.45%	69.60%	10.48%	68.80%	9.65%	72.20%
	Reg - Stepwise	10.45%	69.60%	10.48%	68.80%	9.65%	72.20%
	Reg - Backward	10.47%	69.50%	10.50%	68.90%	9.71%	72.00%

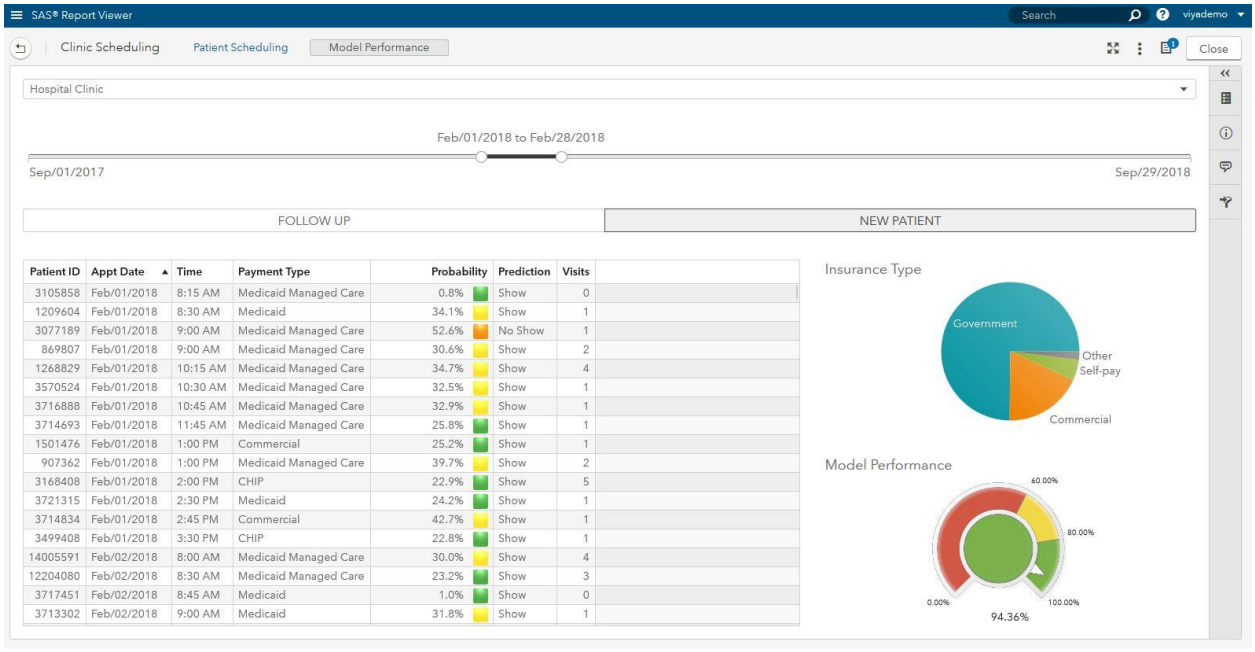
Table 1. Comparison of Models based on Test Data Misclassification Rates

The model selected by this technique was the ensemble model that averaged the boosted decision trees. When compared to other techniques, this model had only a 7.80% misclassification rate and an area of 80.10% under the ROC curve. In general, the Decision Trees outperformed the Linear Regression models.

In order to help fine-tune model outcomes to achieve model performance goals, the model had its probability cutoff rate adjusted. Doing so helped ensure that of the patients predicted to not show up, a majority of these classifications would be correct. The Cutoff node was used to help adjust the model's decisions. By using this approach, the model's precision rate improved from 71.11% to 80.21%. This step helped create a more conservative model and help limit the possibility of two patients showing up at the same time to a clinic.

REPORTING

New patient data is updated on a daily basis at this hospital. A process was set in place that combines the ETL and scoring together in order to load patient data into a SAS Visual Analytic LASR server. Reports were built in SAS Visual Analytics using data scored with the champion models for each clinic. An example of a SAS VA Report can be seen in Display 4.



Display 4. Screen Capture of SAS VA Report.

Medical staff required a dashboard that provided the ability to quickly and easily identify patients with high probability of not showing up for their appointments. The dashboards allowed the staff to filter the data by variable parameters, such as the type of clinic and appointment, a specified date range, hourly time slots, and patient id. These reports also allowed the staff to monitor model performance and track results over time.

CONCLUSION

Making informed decisions about patient scheduling is an important task at medical clinics and hospitals. The solution presented in this paper helps demonstrate how SAS software can handle the data management, analytics, and reporting required to improve the patient scheduling process. By easily identifying patients that are most likely to no-show to their appointments, clinic personnel can use this information to strategically overbook appointments and reduce lost revenue.

The outcome of this project has been successfully implemented in the trial clinics and is currently expanding to include other clinics across the region. This solution is scalable to individual client needs, and can be upgraded to include the SAS® Viya™ platform.

REFERENCES

Harrop, C. 2017. “MGMA Poll: Practice leaders say no-shows, availability are their top appointment challenges.” Accessed January 23, 2018. https://www.mgma.com/data/data-stories/practice-leaders-report-their-biggest-challenges-w.

Wexler, J. and Easterling, P. 2015. “Machine Learning With SAS® Enterprise Miner™: How a Team of SAS® Modelers Created and Determined a Champion Model to Predict Churn Using KDD Cup Data.” SAS Institute, Inc. Accessed January 23, 2018. https://www.sas.com/en_us/whitepapers/machine-learning-with-sas-enterprise-miner-107521.html.

CONTACT INFORMATION

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

Mia Lyst
Pinnacle Solutions, Inc.
(317) 423-9143
mia.lyst@thepinnaclesolutions.com
www.thepinnaclesolutions.com

Scott Koval
Pinnacle Solutions, Inc.
(317) 423-9143
mia.lyst@thepinnaclesolutions.com
www.thepinnaclesolutions.com

Yijie Li
Microsoft Corporation
yijie.li@microsoft.com