Paper 137-2012

A New Age of Data Mining in the High-Performance World

Jared Dean, David Duling, and Wayne Thompson

SAS Institute Inc, Cary, NC, USA

ABSTRACT

Today's businesses are challenged with both analyzing huge data volumes and improving and accelerating their predictive modeling. SAS® High-Performance Data Mining technology is based on the SAS® High-Performance Analytics model for distributed processing alongside the database to make use of multiple processors and vast sums of memory across multiple machines. Data mining features cover the full spectrum of building a predictive model, including data selection, exploratory analysis, transformations, feature selection, dimension reduction, linear and nonlinear modeling, and model performance comparison. New SAS High-Performance Data Mining procedures ease the transition for SAS programmers. This paper discusses the options and methods available for use in High-Performance Data Mining and uses real data for performance benchmarks.

INTRODUCTION

"Big Data" is a popular topic that has been gaining attention from the high-performance computing niche of the information technology market. Big Data is an enormous amount of data from which it is extremely difficult to manage and glean information. Big Data provides both challenges and opportunities for quantitative analysts to develop improved predictive and descriptive models. Provided that your analytical computing environment does not hinder the size of data you can analyze, Big Data enables you to analyze more of the population with a broad range of classical and modern analytics. Analyzing more complete data, versus relying on sampling, enables you to isolate hard-to-detect signals in the data and can also produce models that generalize better when deployed into operations. Enterprises are also moving towards creating large multipurpose analytical base tables that can be used by several analysts to develop a plethora of models for risk, marketing, and so on. Customer dynamics also change quickly, as does the underlying snapshot of the historical modeling data. So the analyst often needs to refresh (retrain) models at very frequent intervals. Now more than ever, analysts need the ability to develop the models in minutes, if not seconds, versus hours or days. Using several champion and challenger methods is critical. Analysts should not be restricted to using one or two modeling algorithms. Model development (including discovery) is also iterative by nature, so data miners need to be agile when they develop models. The bottom line is that Big Data is only getting bigger and data miners need to significantly reduce the cycle time it takes to go from analyzing Big Data to creating ready to deploy models.

SAS launched SAS High-Performance Data Mining in December 2011 to enable you to analyze more data faster than ever before possible. Based on the SAS High-Performance Analytics model for distributed in-memory processing, SAS High-Performance Data Mining is delivered SAS software that uses Teradata or EMC Greenplum hardware. A subset of SAS High-Performance Analytics, SAS High-Performance Data Mining has truly revolutionized the model-building and model-scoring processes. The massively parallel in-memory algorithms enable organizations to derive highly accurate and timely data mining models in minutes, not hours or days, to make better-informed business decisions.

SAS High-Performance Data Mining provides full-spectrum data mining, including data discovery and summarization, variable transformations and reduction, linear and nonlinear modeling, and integrated model comparison and scoring. You can leverage your existing SAS® programming skills to analyze data by using SAS procedures and macros. Alternatively, you can use SAS® Enterprise Miner™ nodes that are specifically designed for high-performance analysis to develop repeatable and shareable process flows without requiring SAS programming knowledge. Score code is generated as a SAS DATA step fragment that requires only Base SAS for deployment on a SAS server or personal workstation. SAS Enterprise Miner model packages can be created and imported into SAS® Model Manager for automated and rigorous promotion to SAS and other database scoring platforms.

The remainder of this paper presents more details about the SAS High-Performance Data Mining components and architecture along with detailed analysis that uses SAS programming and SAS Enterprise Miner nodes. A brief overview of planned enhancements is also outlined. Contact your SAS Account Representatives or SAS Technical Support for more information.

APPLIANCE ENVIORNMENT

The high-performance data mining environment is unique among other solutions and products at SAS. In addition to

the existing tiers, a massively parallel processing (MPP) distributed database and installed SAS components, which constitute an appliance, is used for computation. The appliance consists of one of more server racks, each about the size of a refrigerator. Each rack contains 16 blade servers, and each blade server has multiple processors with multiple cores and many gigabytes of memory. This all adds up to a system that in aggregate contains hundreds or even thousands of CPU cores and hundreds of gigabytes to terabytes of memory. The appliance environment, an x64-Linux platform, is supported by the Message Passing Interface (MPI). Figure 1 shows how the appliance environment relates to the existing SAS Enterprise Miner client/server topology. Communication is needed between the SAS products and the appliance environment, but the client and middle tiers remain unchanged. Keeping the client and middle tiers the same enables existing SAS Enterprise Miner customers to add high-performance functionality while their existing topology remains virtually unchanged.

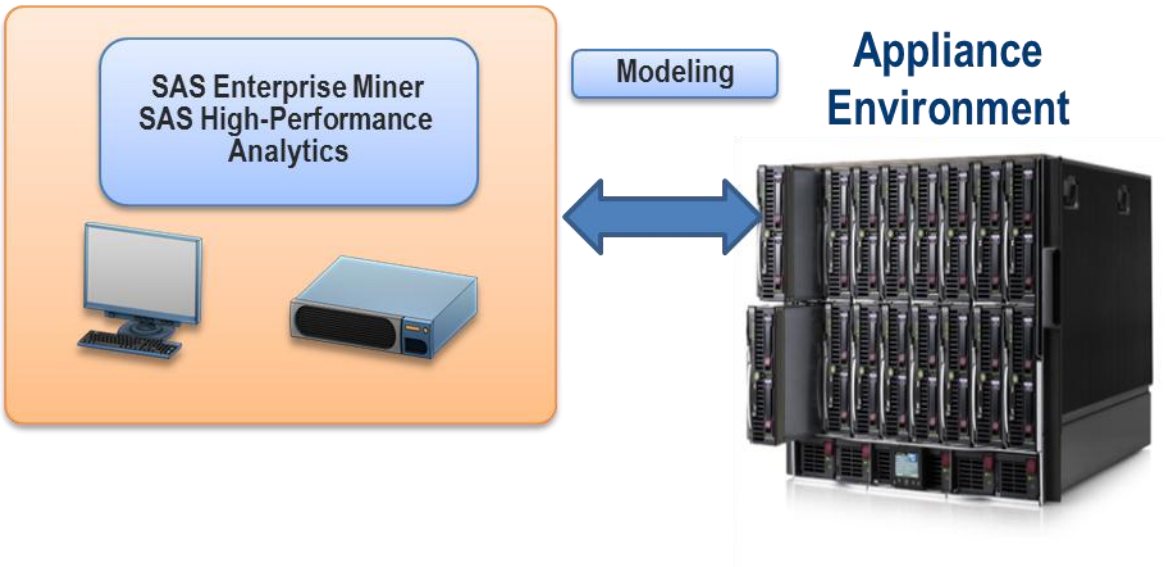


Figure 1. SAS High-Performance Topology

Currently both Teradata and EMC Greenplum databases are supported as high-performance appliances.

SAS HIGH-PERFORMANCE PROCEDURES

Table 1 shows the procedures that have been developed to take advantage of the multithreaded distributed hardware of the appliance environment.

Procedure Name	Functionality
HPDMDB	Summarize data
HPDS2	Parallel execution of DS2
HPFOREST	Random forest
HPLOGISTIC	Logistic regression
HPNEURAL	Neural network modeling
HPNLIN	Nonlinear regression
HPREDUCE	Unsupervised variable selection
HPREG	Regression
HPBIN	Variable Binning
HPSAMPLE	Sampling and data partitioning
HPIMPUTE	Imputation
HPSEVERITY	Severity models
HPCOUNTREG	Regression of count variables
HPSUMMARY	Summarize data

Procedure Name	Functionality
HPLMIXED	Mixed linear models
HPATEST	Test operational status of system

Table 1. SAS High-Performance Analytics Procedures

CONVERTING PROCEDURE OUTPUT INTO SAS SCORE CODE

SAS High-Performance Data Mining provides macros that convert the output from the HPNEURAL, HPLOGISTIC, and HPREG procedures into SAS score code. Table 2 shows the mapping of procedures to macros for creating SAS score code.

Procedure	Macro
HPNEURAL	%HPDM_create_scorecode_neural
HPREG	%HPDM_create_scorecode_reg
HPLOGISTIC	%HPDM_create_scorecode_logistic

Table 2 Macros for SAS Score Code Creation

You can use these macros to create SAS score code that can be deployed to score data anywhere the SAS System is installed. The SAS score code can be converted to DS2 statements and used to score on the appliance environment, or it can be deployed into a relational database using SAS® Scoring Accelerator. (See Figure 2.) The following statements create score code from the HPLOGISTIC procedure and score data on the appliance environment:

```
 *---- create scorecode ----*;
filename lrf '/file/path/lrf.sas';
%HPDM_create_scorecode_HPLOGISTIC(indata=&train.,model=lr_est,modelinfo=lr_info,fil
eref=lrf);
/*=============================================================================*/
*---- run scores on appliance environment ----*;
title4 "HP Logistic Grid Scoring on Hold Out Sample";
proc dstrans ds_to_ds2 in='/file/path/lrf.sas' out='/file/path/lrf2.sas' aster
nocomp; run;
proc hpds2  in=&valid. out=&scores. ; %include '/file/path/lrf2.sas'; run;
```

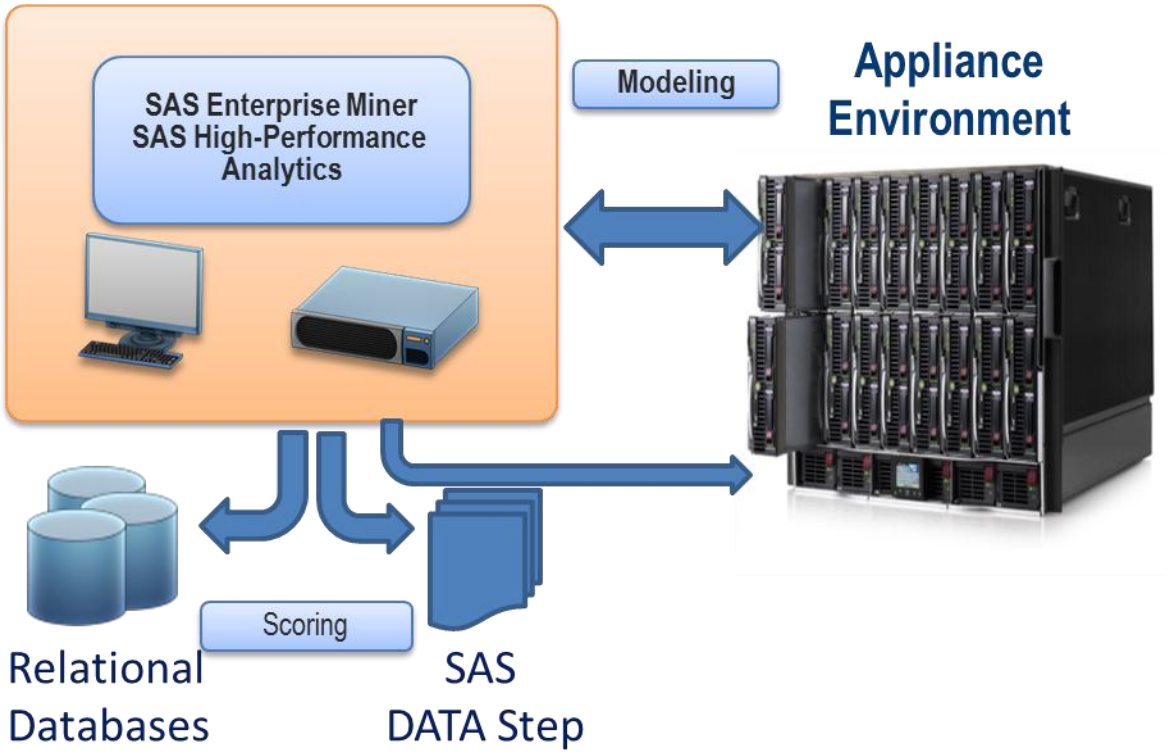


Figure 2. SAS Enterprise Miner "Deploy Many" Scoring

By using other SAS offerings, such as the SAS® Scoring Accelerator, you can also deploy these models to be scored in your database. Figure 2 shows this "deploy many" ability from SAS Enterprise Miner. The models created in the appliance environment can be scored on the same appliance, into a relational database, or to SAS score code. You continue to have the flexibility to score models where you need them.

AIRLINE DATA EXAMPLE

This paper uses a simple example to illustrate how you can use either SAS command-line statements or the SAS Enterprise Miner graphical user interface to perform a high-performance data mining analysis.

The data for this example are records of the US domestic flights from 1987 to 2008. The data include about 123 million observations and has 29 variables. The variable Late (the target predictor variable) evaluates the arrival delay for each flight. If the arrival delay was greater than 15 minutes, then Late equals 1; otherwise, Late equals 0.

USING SAS STATEMENTS TO DEVELOP A HIGH-PERFORMANCE DATA MINING ANALYSIS

1. To gain access to the data through the SAS/ACCESS® engines for Greenplum or Teradata, use a LIBNAME statement similar to one of the following, depending on whether you have Greenplum or Teradata hardware:

```
libname hpdm greenplum server="myserver" user=foo password=bar database=hps;
libname hpdm teradata server="myserver" user=foo password=bar database=hps;
```

2. Set the following new options to take advantage of the appliance environment:

   - `GRIDHOST` identifies the domain name system (DNS) or IP address of the appliance node to which the SAS High-Performance Analytics software connects to run in massively parallel processing mode.
   - `GRIDINSTALLLOC` identifies the directory in which the SAS High-Performance software is installed on the appliance.
   - `GRIDDATASERVER` identifies the database server on Teradata appliances as defined in the HOSTS file on the client. This data server is the same entry that you usually specify in the SERVER= entry of a

LIBNAME statement for Teradata. For more information about specifying LIBNAME statements for Teradata and other relational databases, see the *SAS/ACCESS Interface* documentation for the specific database.

Use the following SAS statements to set these options for a server with a DNS of `hpa.sas.com`, the SAS appliance environment executables in `/opt/TKGrid`, and the data server equal to `myserver` (notice that this name matches the server name in the LIBNAME statement in step 1).

```
option set=GRIDHOST      ="hpa.sas.com";
option set=GRIDINSTALLLOC="/opt/TKGrid";
option set=GRIDDATASERVER="myserver";
```

For ease of readability, specify the following macro variables to set up training data and validation data and to assign variables to class and numeric roles.

```
%let train = hpdm.airline_t;
%let valid = hpdm.airline_v;
%let scores= hpdm._scores;
%let vars  = ACTUALELAPSEDTIME DAYOFMONTH DISTANCE MONTH YEAR;
%let class = DAYOFWEEK UNIQUECARRIER;
%let tr_tran = hpdm.airline_t_trans;
```

3. Use the HPATEST procedure to check the number of operational nodes and get a baseline for a single-pass read time for the time it takes to read through the data one time.

```
%let hpa_enabled=;
proc hpatest data=&train;

performance nodes=all details;
run ;
%put NOTE: HPATEST RETURNS: &hpa_enabled;
```

All of the examples in this paper contain a PERFORMANCE statement. The PERFORMANCE statement is common to all SAS High-Performance Analytics procedures; it enables you to write details about execution timing or to set the execution details for that procedure run. Some of the items that are controlled by the PERFORMANCE statement are the number of nodes or threads that are used in the execution.

4. Use the HPDMDB procedure to summarize the data.

```
proc hpdmdb data=&train. classout=c varout=v;
var &vars ;
class &class late;
run;
```

Look at missing values, frequency distributions, and basic summary statistics. If a tabular view of the data is insufficient, you can create a sample and move it to the SAS client where you can use SAS/GRAPH to create plots as shown in step 5.

5. Use the HPSAMPLE procedure to create a 0.1% sample of the data, and write it to the `work.sample` data set:

```
proc hpsample data=&train. out=sample samppct=0.1 seed=1 ;
performance details ;
class DAYOFWEEK UNIQUECARRIER late ;
var ACTUALELAPSEDTIME DAYOFMONTH DISTANCE MONTH YEAR ;
target late ;
run ;
```

The TARGET statement causes the sample to be stratified by the Late variable, which predicts whether your flight will arrive late. (Late is a prediction variable.) Sampling enables you to avoid transferring large amounts of data across your network.

The GCHART procedure creates bar charts of the Late, DayOfWeek, UniqueCarrier, and Distance variables from the sample created by PROC HPSAMPLE.

```
proc gchart data=sample ;
hbar late /discrete ; run ;
hbar dayofweek /discrete ;  run ;
hbar uniquecarrier ; run ;
hbar distance ; run ;
quit ;
```

6. Use one or more of the HPBIN, HPIMPUTE, or HPDS2 procedures to transform the data in preparation for modeling. This example uses PROC HPDS2 as follows:

```
proc hpds2 in=&train. out=&tr_tran.;
data DS2GTF.out;
 method run();
 set DS2GTF.in;
if ^MISSING(DISTANCE) & DISTANCE ^= 0.0 then AIRSPEED =ACTUALELAPSEDTIME
/ DISTANCE;
 else AIRSPEED = .;
_return: ;
end;
 enddata;
 run;
quit;
```

The HPDS2 procedure enables you to make custom transformations. A variable named Airspeed is created based on the values of ActualElapsedTime and Distance.

7. Use the HPNEURAL procedure to create a model. The following statements build a model on your training data by using both the numeric and character variables to model Late with four hidden neurons:

proc hpneural data=&train. ;

```
PERFORMANCE DETAILS ;
input &vars / level=int ;
input &class / level=nom ;
target late / level=nom ;
hidden 4 ;
train outmodel=nn maxiter=50 ;
score out=&scores. ;
run ;
```

The procedure trains for up to 50 iterations if the stop criteria are not met. The SCORE statement outputs a data set that includes the actual outcome and the probability of each event level. The target, Late, has a binary response, so the score data set includes variables p_late0 and p_late1 both of which indicate the predicted probability of the event. If the target variable had more levels, then more variables would appear in the score data set.

8. Use the HPLOGISTIC procedure to produce a competing model. The syntax of PROC HPLOGISTIC is very similar to the syntax of the LOGISTIC and DMREG procedures. The following SAS statements use a stepwise selection method perform a logistic regression on the Late variable. Once again the scores are output.

```
proc hplogistic data=&train. itselect nostderr ;
PERFORMANCE DETAILS ;
id id late ;
class &class late ;
selection method=stepwise ;
model late(desc) = &class &vars ;
ods output ParameterEstimates=lr_est ModelInfo=lr_info ;
```

```
output out=&scores. predicted ;
run ;
```

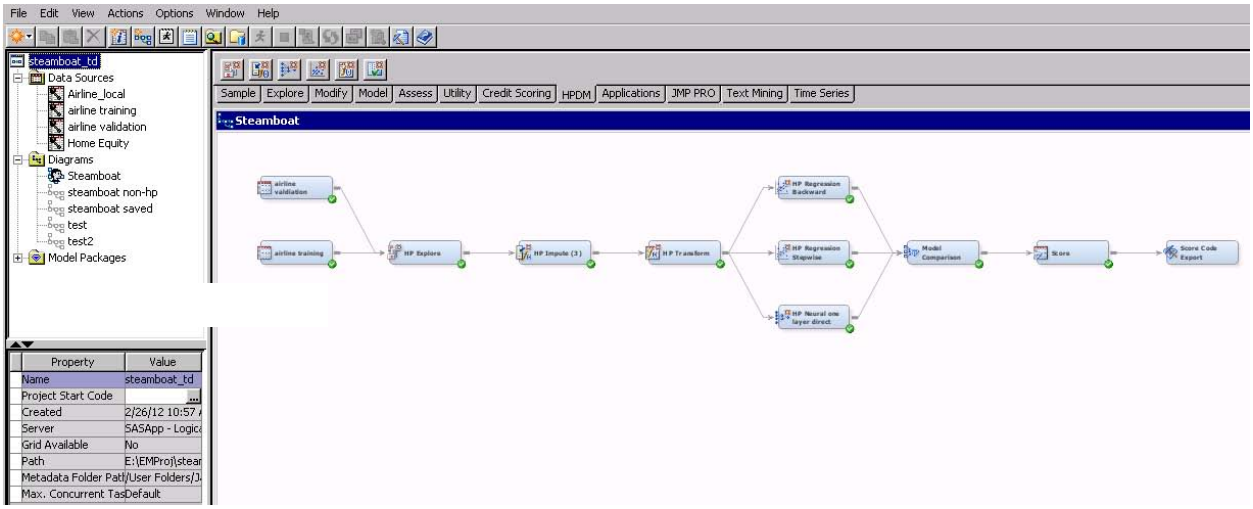
9. With these two models created, you need a way to assess the quality of each model and compare the models to each other. SAS High-Performance Data Mining provides the %EM_new_assess macro to compute model assessment measures. This macro uses a mixture of the DATA step, Base SAS procedures, SAS Enterprise Miner procedures, and SAS High-Performance Analytics procedures. The following statements call the %EM_new_assess macro and the subsequent macro %EM_new_report:

```
%em_new_assess(data=&scores.,out=lrbins,target=late,event=1,var=pred,from=late,
into=predict,hpds2=1);

%em_new_report(bins=lrbins,from=late,into=predict) ;
```

USING THE SAS ENTERPRISE MINER USER INTERFACE TO DEVELOP A HIGH-PERFORMANCE DATA MINING ANALYSIS

The SAS High-Performance Data Mining functionality is added to the existing SAS Enterprise Miner user interface to build upon the significant set of Enterprise Miner tools and to give existing Enterprise Miner customers a familiar interface for leveraging the appliance environment. Display 1 shows the Enterprise Miner user interface with the new *HPDM* toolbar.



Display 1. Enterprise Miner User Interface with High-Performance Data Mining Toolbar

SAS High-Performance Data Mining provides six nodes exclusively for the appliance environment. These nodes use a number of new procedures that are designed to take full advantage of the memory and computing power of the appliance. Table 3 shows the relationship of the high-performance nodes and the procedures they use. In addition to the procedures in Table 3, the HPDS2 procedure is used extensively to prepare and process data as it moves through the process flow.

High-Performance Node	Function	Procedures Used
HP Explore	Provides summary statistics and graphics to aid in your basic data exploration.	HPDMDB
HP Impute	Includes options for class and interval variables and can create indicator variables in the same manner as the Impute node.	HPBIN, HPDMDB
HP Neural	Creates a neural network model for binary, nominal, or interval targets.	HPNEURAL

High-Performance Node	Function	Procedures Used
HP Regression	Creates a regression model for binary, nominal, or interval targets.	HPREG, HPLOGISTIC
HP Transform	Supports many standard transformations for interval variables and provides options for binning of interval variables, which is important in many business applications. In addition to the many included transformations, the HP Transform node includes a DATA step editor so you can specify custom transformations.	HPBIN, HPDS2
HP Variable Selection	Selects important variables through either supervised or unsupervised methods. This node also has a sequential selection that runs unsupervised followed by supervised variable selection.	HPREDUCE, HPLOGISTIC

Table 3. High-Performance Nodes and Their Corresponding Procedures

The creation of projects and diagrams are the same as in previous releases. The following steps detail how to explore, transform, model, and score the same Airline data as are used in the previous section.

1. To gain access to the data through the SAS/ACCESS engines for Greenplum or Teradata, use a LIBNAME statement similar to one of the following, depending on whether you have Greenplum or Teradata hardware:

```
libname hpdm greenplum server="myserver" user=foo password=bar database=hps;
libname hpdm teradata server="myserver" user=foo password=bar database=hps;
```

2. Set all of the options shown in step 2 in the previous section and set one more macro variable, HPDM_WORK, as shown in the following example.

```
%global HPDM_WORK;
%let HPDM_WORK = hpdm;
```

   The HPDM_WORK macro variable identifies the library on the appliance environment that should be used as a work area. The value of HPDM_WORK should match the name of the library that was established through the SAS/ACCESS engine in step 1.

3. You must make the data set to be analyzed visible to SAS Enterprise Miner. The most common ways to make it visible are by using library assignment statements in the project start code or as preassigned libraries in metadata. In the previous step, the HPDM library was added in the start code.

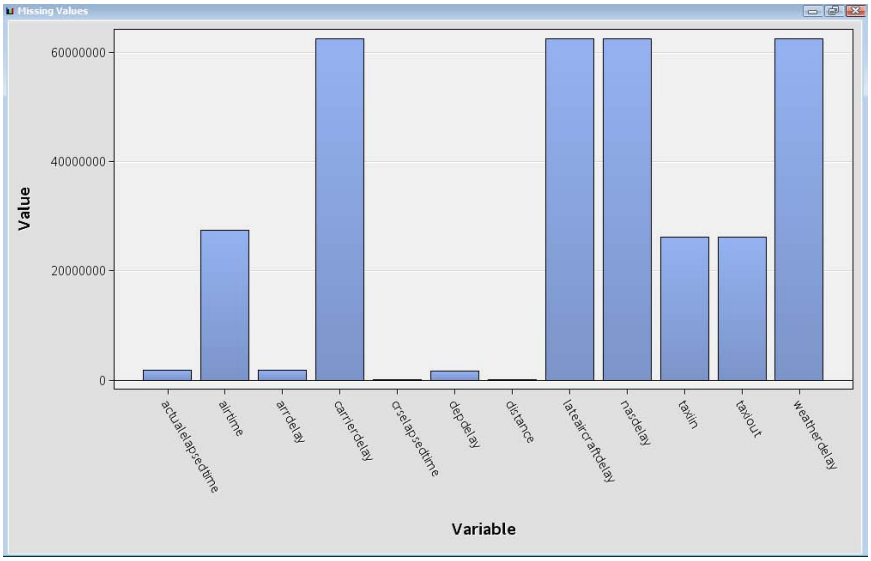
   Usually the data to be analyzed is loaded on the appliance environment outside of the SAS Enterprise Miner process by using either a vendor's bulk loading tools or the HPDS2 procedure. The process for registering a data source is the same as the process currently supported by SAS Enterprise Miner.

   Because not all SAS Enterprise Miner nodes work in the appliance environment, a sample is automatically taken when the data source is registered in case it is needed in the future. This sample is created by using the HPSAMPLE procedure and is stratified by a nominal or a binary target if one exists; otherwise, a simple random sample is taken.

4. You can create a diagram using the same process currently supported by SAS Enterprise Miner and begin to build a process flow to leverage the appliance environment by adding nodes to the diagram and connecting them in your desired order.

5. Use the HP Explore node to explore your data. Display 3 shows the default results for the Airline data set. You can see that many variables have missing values and some variables have a significant proportion of missing values. An imputation step is probably wise. You also learn from the results of the HP Explore node that overall 18% of aircrafts arrive late, Saturday is the lowest flight day, the longest flight is almost 5,000

miles (Newark to Honolulu), and there are flights from JFK to LGA (about a 20-minute drive according to Google Maps).



Display 3. Missing Value Bar Chart

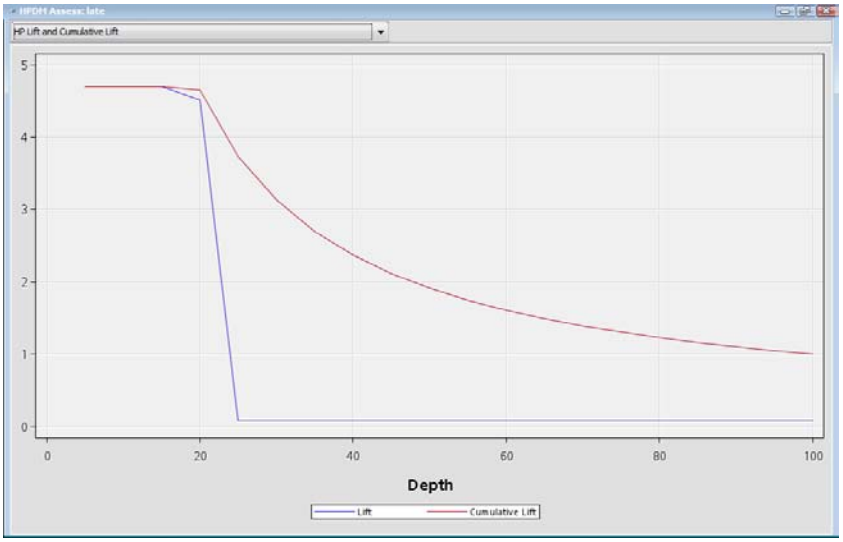
You can also explore the data by using the existing SAS Enterprise Miner nodes (Stat Explore, Graph Explore, Multiplot, and so on), but don't forget that these nodes run against the sample of the data that was created when the data source was registered, not the entire data set.

6. Use the HP Impute and HP Transform nodes to generate DATA step statements which are translated to DS2 statements prior to modeling. These nodes enable you to continue to write in the DATA step you are familiar with and reap the benefits of processing in the appliance environment.

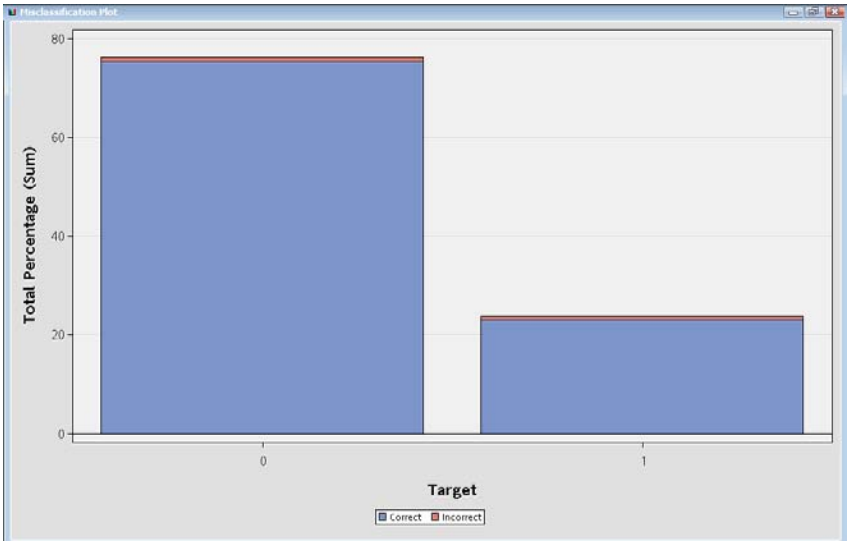
7. Use the HP Neural node or HP Regression node to model whether a flight will arrive late. The HP Regression node calls the HPLOGISTIC procedure, and the HP Neural node models a nominal target. If the problem were recast to predict how many minutes late the flight would arrive, these same nodes would be used, but instead the HPREG procedure would be called and the HPNEURAL procedure would be called to model an interval target. Both nodes include a full set of diagnostic and reporting output to aid in model development and decision making. The tabular and graphical outputs are created by using results from the %EM_new_access macro.

8. Now you need to choose a model to predict which flights will be late. Display 4 shows a lift of nearly 5 using logistic regression with backward selection. Display 6 shows the misclassification plot from the HP Neural node. As you can see, both models perform very well in predicting which flights will arrive late.

9. Use the Model Compare node (which is unchanged from previous releases) to compare these models. The Model Comparison node creates additional fit statistics and then informs you of the best model based on your selection criteria.



Display 4. Lift Chart from the HP Regression Node



Display 5. Misclassification Plot from the HP Neural Node

10. Use the Score node and the Score Code Export node to generate both DATA step score code for the Airline training data and a holdout validation sample and to prepare the SAS score code to be exported for the SAS Scoring Accelerator. Score code can also be deployed back to the appliance environment as shown earlier or as DATA step code as follows:

```
Filename HPDMMDL "/path/to/model/scoreCode.sas";
DATA mydatabase.TableToScore;
%include HPDMMDL;
run;
```

FUTURE DIRECTION

SAS is fully committed to delivering high-performance data mining extensions. Jim Goodnight, CEO of SAS, has set a goal that no analytics problem shall be too large or complex for SAS to help its customers solve. One of the key high-performance areas for future development is text mining. Text represents over 80% of all collected data, whether large or small. SAS will include in-memory tools for text parsing (accumulation into a term by document matrix and singular value decomposition to support textual clustering), classification, and enriched integrated predictive modeling.

Another focus is enabling customers to automate the development of large numbers of models. SAS High-

Performance Data Mining will support distributed BY-group processing to support the training of separate models in parallel for class variables, such as store, region, or product. An automated decision tree ensemble will also be included as another challenger model class. SAS is also developing an automated rapid predictive modeling tool to ease the process of creating efficient, accurate, and robust data mining models for both statisticians and business analysts. This new tool will automatically treat the data to handle outliers, missing values, rare target events, skewed data, collinearity, variable selection, and model selection. Advanced users can customize the model settings.

Another planned model is based on scorecards, which are easy to interpret and are a standard for credit risk assessment. Scorecard models require additional advanced binning. Additional transformations, such as capping extreme values, will be also be added.

CONCLUSION

The current buzz of activity around Big Data is motivated by genuine business needs to process greater amounts of information in less time to produce predictive models in a data mining environment. These requirements are met by SAS High-Performance Analytics software. SAS users continue to work inside their familiar environments while SAS High-Performance Analytics procedures and macros execute functions seamlessly on the SAS high-performance appliance. The Airline flight delay examples demonstrate how users can work effectively either with the SAS language (with full access to data preparation, model building, evaluation, and scoring functions) or inside SAS Enterprise Miner. The examples show a full predictive modeling analysis that produces score code that can be executed in multiple batch and real-time environments.

Furthermore, SAS High-Performance Analytics also facilitates a larger change for data miners. When you use conventional tools on very large data, you need to laboriously explore your data, construct small representative samples, and carefully schedule model-building jobs to run for extended periods of time. Now, you can work in a more natural interactive style where you can generate a hypothesis and test with near immediate feedback. You can spend less time building models and have more opportunities to explore relationships, generate and select features, and compare multiple models. You will certainly be able to address more modeling projects. Thus, you will indeed enter a new age of data mining.

RECOMMENDED READING

SAS High-Performance Analytics 1.3: User's Guide Cary, NC: SAS Institute Inc.
SAS Enterprise Miner 7.1 High-Performance Data Mining Procedures and Macro Cary, NC: SAS Institute Inc.

SAS Enterprise Miner High-Performance Data Mining Node Reference Help. Cary, NC: SAS Institute Inc.

ACKNOWLEDGMENTS

The authors would like to thank Dominique Latour and Ye Liu for their contributions to this paper. They are also grateful to Anne Baxter for her editorial contributions.

CONTACT INFORMATION

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

Jared Dean
S6120
SAS Institute Inc.
SAS Campus Drive
Cary, NC, 27513
Jared.Dean@SAS.com

David Duling
S6100
SAS Institute Inc.
SAS Campus Drive
Cary, NC, 27513
David.Duling@SAS.com

Wayne Thompson
S6102
SAS Institute Inc.
SAS Campus Drive
Cary, NC, 27513
Wayne.Thompson@SAS.com