Best Practices for Managing Predictive Models in a Production Environment

Robert Chu, SAS Institute Inc., Cary, NC
David Duling, SAS Institute Inc., Cary, NC
Wayne Thompson, SAS Institute Inc., Cary, NC

ABSTRACT

The widespread use of predictive analytics has enabled organizations to more accurately predict their business outcomes, improve business performance, and increase profitability. As the sheer number of these models in the overall portfolio is coupled with growing requirements to demonstrate external compliance, it is imperative that the organization implements sound model management practices. Model management is not a one-time activity but an essential business process. Models must be well developed and validated to demonstrate that they are working as expected. Outcomes analysis is also necessary to ensure that the scores derived from applying the model to new data are accurate and to verify that model performance remains satisfactory. This paper presents SAS-based strategies for effectively managing predictive models in a production environment and introduced a new product, SAS® Model Manager 2.1 for SAS® 9.1.3.

INTRODUCTION

Organizations today are increasing their use of predictive analytics to more accurately predict their business outcomes, to improve business performance, and to increase profitability. Common and yet also highly strategic predictive modeling applications include fraud detection, rate making, credit scoring, customer retention, customer lifetime value, customer attrition/churn, and marketing response models. As the number of these models increases to support more and more business objectives, so does the requirement to manage these models reliably and securely as valuable corporate assets. Many companies, especially those in the financial services sector, also need to demonstrate adherence to model validation and governance practices outlined by the Office of the Comptroller of the Currency Administrator of National Banks (2000), the Basel II Committee's Accord Implementation Group (Basel Committee on Banking Supervision 2005), and other governing bodies. This paper provides an overview of the model management life cycle process as well as an introduction to SAS Model Manager 2.1 with a focus on recommended practices for the improved management of predictive models in a production environment.

BUSINESS PROBLEMS

Before discussing the model life cycle management process, it is worthwhile to provide examples of key business pains faced by many corporations who develop and deploy large numbers of predictive models. While some corporations can go through the entire process of preparing the analytical base tables, developing models, and deploying the champion model in less than two months, many organizations might take 10 months or longer to deploy a champion model with some models never being deployed. Delays in deploying a model obviously result in lost opportunities and might even result in a model that no longer provides useful predictions due to changes in market and economic conditions. Model deployment for many organizations simply takes too long.

Model deployment setbacks can often be attributed to technical integration issues. For example, many organizations lack a common integrated framework for comparing candidate models to select the champion model. Models are also often developed by more than one data modeler using a host of different algorithms along with potentially different development and test data sources which further add to the complexity of selecting models. The process used to develop the model along with the definition of the development and test data sources needs to be documented and accessible so that others can review and update the model as needed.

The analysts often use a data mining software package that generates model-scoring code in a particular programming language. However, the IT department scoring officer might convert the scoring code to another language either due to policy or to enable integration of the code into an entirely different client system. For example, the score code might need to be integrated into a call center application which is on a different operating system than the one originally used to develop the models. Code conversion introduces the potential for making costly translation errors. A single error in the model-scoring logic results can easily result in an incorrect classifier which can cost the company millions of dollars. Converting the scoring code, which often exceeds a thousand lines, is usually a slow manual process. Some companies even have a multi-step process which requires translating the model into an intermediate document with comments describing the scoring logic which the scoring officer then converts to another language. Scoring also requires that the model inputs be available in the operational data store. A critical requirement for many corporations is a set of good controls for data integrity and binding for integrating the model-scoring logic with the operational data store.

Many corporations also need a central repository for storing models along with detailed metadata for efficient workgroup collaboration. Their model complement has become too large to maintain and review on a consistent basis given resource constraints. The challenge to save all documents that are related to a model and be able to retrieve those documents easily when needed is a common business pain.

Model management also involves a collaborative team of data modelers, data architects, scoring officers, and validation testers. Many organizations are struggling with the process of signing off on the development, validation, deployment, and retirement life cycle management milestones. They need to readily know exactly where each model is in the life cycle, how old the model is, who developed the model, and who is using the model for what application. The ability to version-control the model over time is another critical business need which includes event logging and tracking changes to understand how the model form and usage is evolving over time.

Model decay is another serious challenge faced by organizations. Metrics are needed to determine when a model needs to be refreshed or replaced. Retired models also need to be archived. More reliable management of the score repositories is also a key requirement to ensure that quality representative data is available to evaluate model performance and profitability over time.

SAS MODEL MANAGER 2.1

SAS Model Manager is a new product designed for the selection, maintenance, and continuous enhancement of SAS® analytical models for operational decision making. SAS Model Manager is all about enabling processes to effectively manage and deploy analytical models by delivering all the necessary functionality for each stage of the model life cycle.

SAS Model Manager (Figure 1) includes a secure, centralized repository for storing and organizing models. Models are organized by projects that map to the business purpose or application. Each project contains the champion and challenger models along with extensive metadata, scoring code, data sources definitions, and supporting documentation. Supported models include prediction, classification, segmentation, and rules-based models developed using SAS® Enterprise Miner™, SAS/STAT®, and Base SAS®. Models can be promoted from one version to another within a project as they are applied over time. Event logging and notes correspondence are also supported.

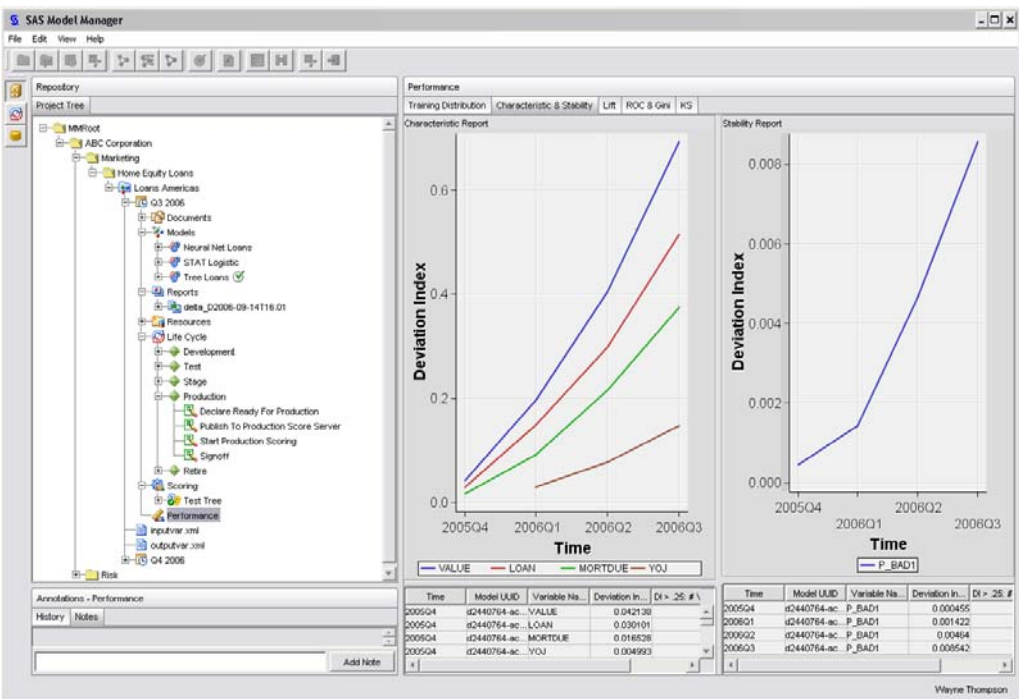

Figure 1. SAS Model Manager (includes a project-based repository for cataloging analytical models and also several model comparison and monitoring reports)

SAS Model Manager includes a set of life cycle templates for pushing the model through the development, validation, deployment, and retirement milestones. The templates include start and completion dates along with completion status reports. Templates can also be customized to meet the organization's specific model management signoff processes.

Prior to deployment in production, scoring officers can develop test scoring jobs to ensure that the model is accurate and produces output as expected without any errors or warnings. Lastly, SAS Model Manager includes performance monitoring reports to determine if the model has degraded as it is being scored over time in the operational environment. A publishing framework is also included to keep key individuals informed of model life-cycle milestone events and performance monitoring alerts.

MODEL LIFE-CYCLE MANAGEMENT

To use predictive models successfully, you need to be able to manage models from the development phase through to your production environment. Model life-cycle management is an efficient iterative process that consists of the following stages:

- Determine the business objective
- Access and manage the data
- Develop the model
- Validate the model
- Deploy the model
- Monitor the model

DETERMINE THE BUSINESS OBJECTIVE

The first stage is to identify what type of model is needed, which type of application the model will be used for, and develop a strategy to ensure that the business units understand how the model is to be utilized once it is deployed. Typical models include those used for predictive modeling of customer behavior, risk management, and credit scoring. Applications can include customer retention, credit origination, transaction authorizations, fraud detection, and marketing campaigns. These concerns drive the data acquisition and model development processes.

ACCESS AND MANAGE THE DATA

The second stage is to manage the vast proliferation of data. Data management and data quality are key components of the mining process; this is where systems spanning multiple platforms and containing multiple data sources (for example, accounts receivable, accounts payable, call reports, demographic data, financial data, card holder data, and credit bureau data) need to be integrated and synchronized into a clear and coherent format. Both the analytic model development and production deployment environments need to be aligned in this data management process to create data consistency and to ensure that a single version of the truth is utilized throughout the model life cycle. SAS provides integrated data quality routines and data management transformations to create a single, consistent version of the truth while allowing for maximum flexibility and scalability across your organization's distributed environment.

Developing a predictive model will involve collecting appropriate data, sampling, aggregating data attributes, performing segmentation analysis, and conducting other in-depth data analyses. For example, data sources could be brought together at the customer or account level from many disparate sources. This can include billing and payment transactional data, demographic figures, financial data, and so on. Transformations are then applied to further enrich the data. Transformations include:

- Computing rollup variables such as the average, maximum, and total balance across all accounts for a given customer
- Computing interval variables such as how long an individual has been a customer, value of the last purchase, and so on
- Applying the logarithm to heavily skewed data such as median household incomes
- Transforming multimodal distributions to categorical values
- Filtering outliers and replacing missing values
- Segmenting customers into groups with common attributes
- Creating interaction terms

Many organizations will attempt to create a standard data source for all modeling activities. This process can help to enable auditing activities, standardize best practices, and simplify deployment with varying degrees of success. For business objectives that do not change, this can be very effective. However, when presented with new target variables, the data modeler often will return to the source data systems to acquire new data not already found in the model training data mart. The most effective guidelines allow both types of activities and provide tools that capture new data acquisition logic and incorporate it into the model training data mart. SAS Model Manager can aid this practice by both specifying the model development data to be used for new projects and tracking the model training data actually used in individual models. A prototypical data build for modeling data is shown in Figure 2.

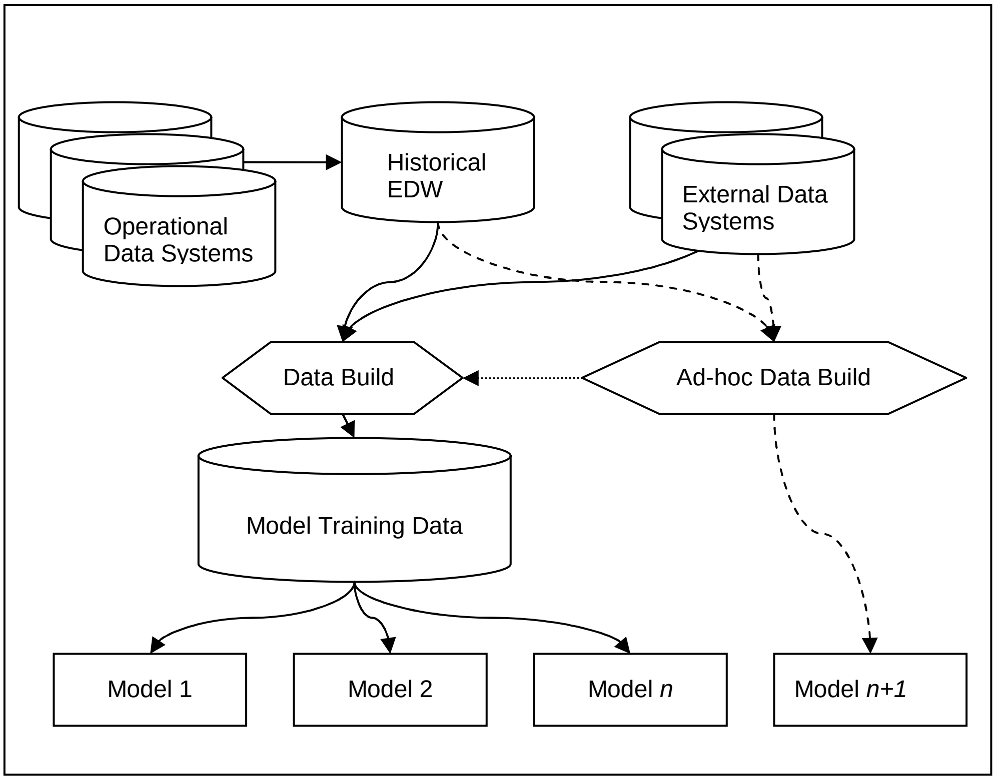

Figure 2. Prototypical Data Build for Modeling Data Incorporating both Routine Model Creation and New Model Creation

The resulting data aggregation code produced in the data management stage of the financial model will be further enriched as it is integrated into the model development code. For a complete discussion on data preparation methods and strategies, please see *Data Preparation for Analytics Using SAS®* (Svolba 2006).

DEVELOP THE MODEL

The third stage is to build the model based on the representative training data source defined in the data management stage. This process involves the application of exploratory statistical and visualization techniques, variable transformations, filtering outliers, data replacement, segmentation, clustering, predictive modeling algorithms, and model validation as examples.

The use of these techniques will be guided by the business objective. Application credit score models are highly regulated in their use of data and model forms by both U.S. fair lending guidelines and Basel II regulations. Behavior credit score models are used to rate existing loan portfolios and may be subject to both Basel II and Sarbanes-Oxley regulations. Many businesses have their own best practice guidelines which state that certain variables must be included, combinations of others must be excluded, and that a priori segmentation schemes must be honored.

Definition of the dependent variable (in this case typically, loan default) can have many forms such as time since last payment, number of missed payments, ratio of accrued interest, or formal loan cancellation. Customer acquisition and cross-sell models are typically based on response to previous promotions within some time period and/or through some channels. Manufacturing root cause analysis might be based on reported failure rates-per-unit lots. To accurately track model performance over time by following accuracy rates, the definition of the dependent variable must be consistent over time. If the dependent variable definition has changed, this should be noted as a new model form, and a new set of tracking statistics should be generated.

Model input terms have similar regulatory and deployment restrictions. Data used in building the model must be updated and available during the lifetime of the model deployment and scoring processes. Demographic and transactional data sources might change their field definitions and codes during the lifetime of a model; therefore, the modeler should build models that have robust missing value and range and value-checking logic. Avoid terms that are known to be unstable or might not be consistently available.

Highly correlated input variables can lead to model sensitivity issues where slight changes in relative values of data fields can cause indefensibly large changes in predicted values. This is a typical concern when a model is expected to be in production for a long time. Aggressive variable selection, use of exploratory correlation statistics, and variable clustering can be effective in reducing long-term instability.

Many businesses implement a champion challenger strategy. The champion is often a model that has already been in production and is based on data from several previous time periods. Newer challenger models are built on data from more recent time periods. In high-throughput environments where thousands of models are generated, the functional form is most likely fixed and the models are simply retrained. In other environments, a data modeler might spend weeks developing a set of new challenger models. For both model monitoring and business auditing purposes, both champion and challenger models should be archived into a model database. They should be compared on blind test data that represents the most recent or most stable samples.

Both SAS Enterprise Miner and SAS/STAT provide complete predictive model-building capabilities to address all listed issues, and users of both systems can add models to the SAS Model Manager. SAS Enterprise Miner users can use GUI capabilities to register a model to the SAS metadata server where it can then be accessed by SAS Model Manager users. SAS/STAT users can use model registration macros provided by SAS Model Manager to automate the model registration process. Once registered, models from both products are represented by the following essential attributes:

- Model score code written in the SAS language
- A list of essential input variables required by the model
- A list of output variables created by the model
- Data aggregation code in the SAS language used or developed by the data modeler
- The functional form of the model, such as logistic regression, neural network, decision tree
- The training data used to develop the model
- The validation data used to control model over-fitting

Once registered, the expected performance of multiple models created by both SAS/STAT and SAS Enterprise Miner users can be compared on blind test data sets. SAS Model Manager will score the models and compute statistics such as single-point error measures as well as lift and captured response values. Figure 3 shows an example of a dynamic lift chart.

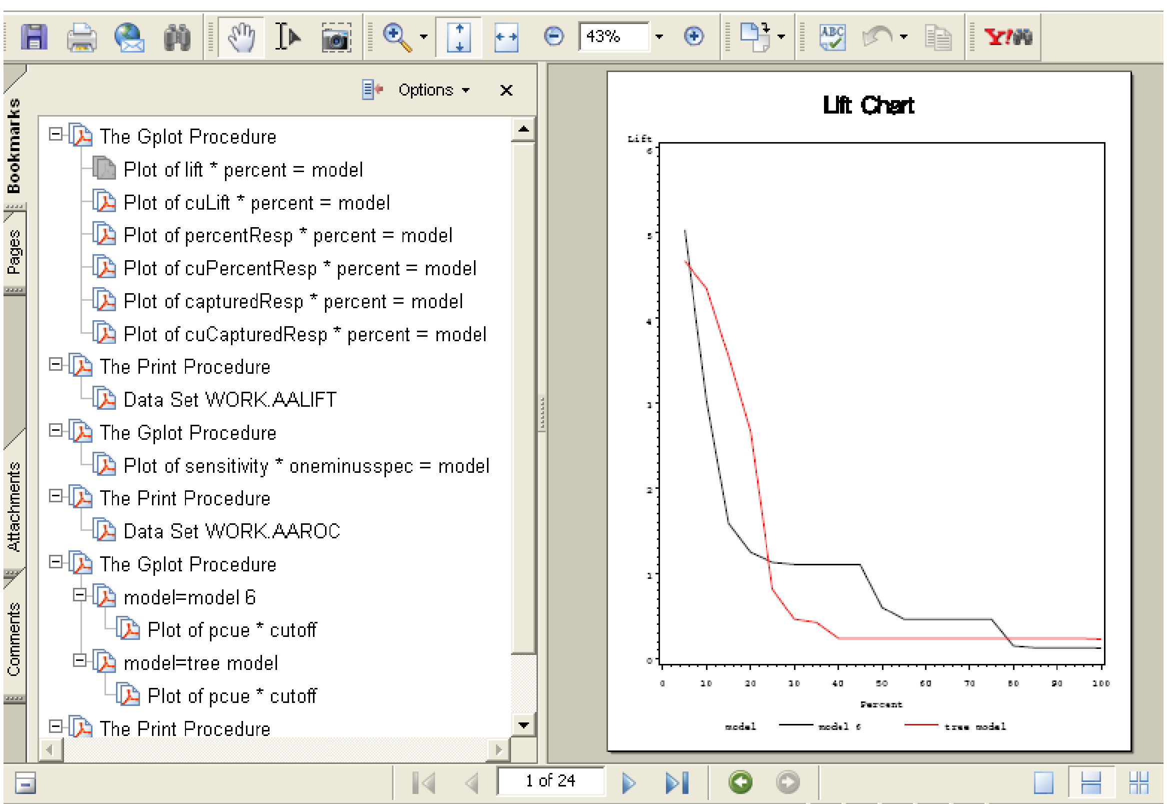

Figure 3. Dynamic Lift Chart

VALIDATE THE MODEL

With increased adoption of analytical methods for driving business decisions, predictive models are being viewed as important intellectual assets for organizations. Each production model is important and could have significant impacts on corporate bottom lines, compliance, and legal/economic risks. This phenomenon is forcing more and more organizations to establish model validation as a formal business process.

Validating a model is not a one-time task but a continuous process. It typically includes the following key tasks:

- **Validate predictors for legal issues:** While certain variables might be good predictors, they may result in unwanted risk exposure for an organization when used in a predictive model. For example, there can be legal reasons why variables such as age, gender, and/or race might not be used in the model-building process.

- **Validate data distributions:** It is important to understand the initial distribution of target and predictor variables in order to recognize distribution shifts over time. If distribution shifts are detected, it might be necessary to retrain the model on new data.

- **Validate analytical algorithms:** The algorithms chosen to generate the models need to be validated in light of the potential use of the predictive model. For example, certain models such as decision trees produce results that are easy to be interpreted. They enable you to answer questions like “Why was this customer denied credit?” Other models such as neural networks do not provide this sort of simple interpretation and hence might be inappropriate for certain applications. The trade-off between interpretability and prediction accuracy must be carefully considered at this stage.

- **Compare model-prediction accuracy:** For a particular data mining project, modelers might use multiple tools to create a set of potential models. It can be difficult to compare models across tools, and model comparison might need to be done based on an independent data source that was not used in the model creation process.

- **Audit validation processes:** Validation processes can vary over time. One thing that is consistent is that every step of the validation process needs to be logged. For example, who imported what model when; who selected what model as the champion model, when and why; who reviewed the champion model for compliance purposes; and who published the champion to where and when.

- **Perform pre-deployment scoring testing:** Before a new champion model is published for production deployment, an organization might want to test the model for operational errors. This type of pre-deployment checking is important especially when the model is to be deployed in real-time scoring environments.

- **Monitor model performance:** Once a champion model is published, it can be deployed repeatedly in a production environment. Typically, model performance degrades over time. Organizations need to systematically detect performance degradation in order to weed out obsolete models and build new ones. Without appropriate design for monitoring automation in the first place, model performance monitoring processes could be time-consuming and error-prone.

Model validation is a business process. It cannot be accomplished in an efficient and manageable manner without good software tools. SAS Model Manager is designed with the above tasks in mind.

DEPLOY THE MODEL

Once a model is validated, the business needs to put the model into production. This requires implementing a scoring system where the model function is applied to new data that might not have a dependent variable. Most scoring systems are batch style where thousands to millions of records are input to one or more models. If the business has efficiently defined a common modeling data set, a single data-building job can construct one table that serves many models. For a direct-marketing campaign, this job might be run ad-hoc by the modeling or IT staff and the scores sent to campaign execution staff. For model monitoring purposes, the model might be scored every month when new values for the dependent variable are available and real performance is compared to expected performance.

For models that contribute to operational business decisions such as an application score model or a cross-sell model, the target system is typically an operational or transaction handling system. Transactional systems have a defined throughput requirement which can impact the amount of input data that is used and the complexity of code that is executed. Often, these systems rely on pre-computed scores based on the most recent offline data. In this case, batch scores are loaded onto data systems that provide fast access to the transactional systems. Often, a significant number of models are needed for scoring, and scoring results are loaded before the next transactional time period. With the ability to read model definitions from SAS Model Manager, access virtually any data source, schedule, and grid-distribute these jobs, SAS® Data Integration Studio is an ideal candidate for this task. The transaction system can then use an indexed lookup to retrieve the model score for a known customer. Figure 4 illustrates this case.

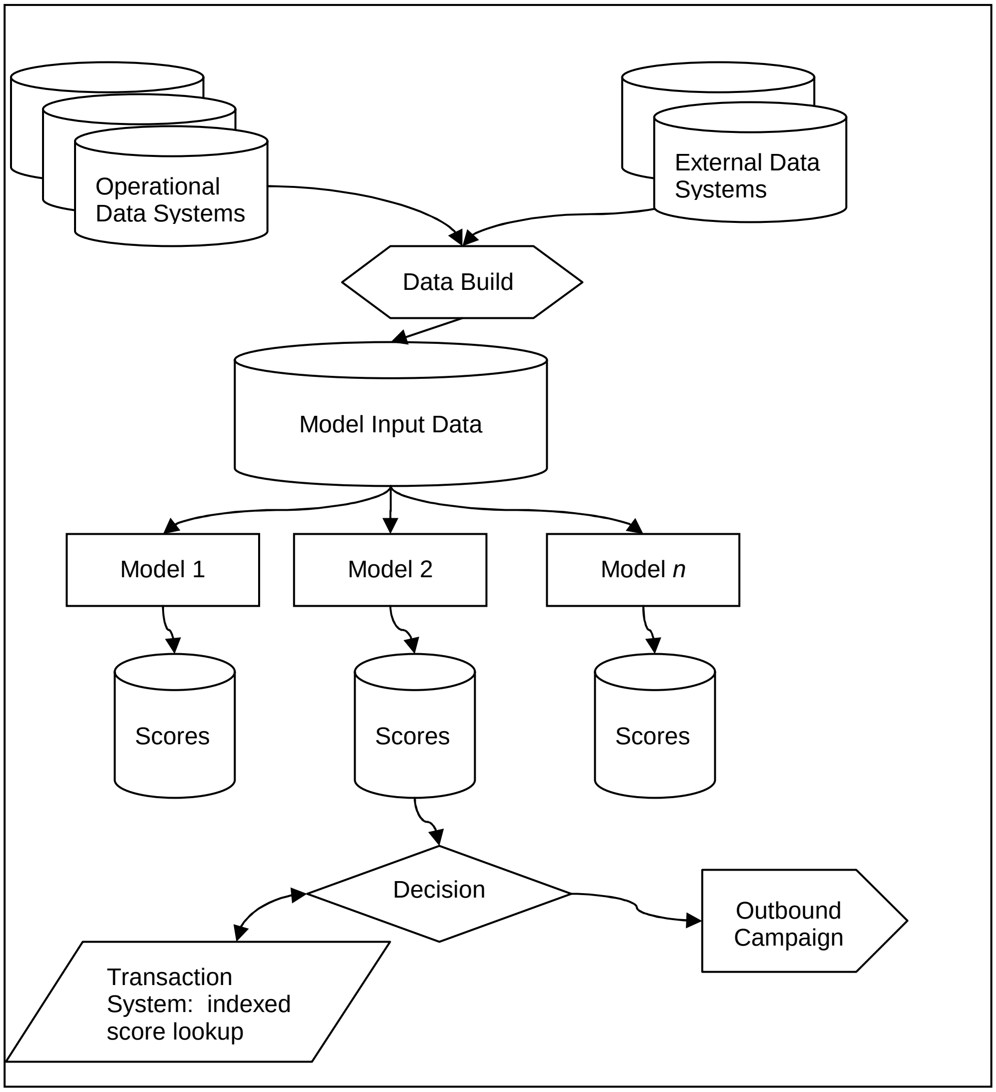

Figure 4. Batch Scoring. This diagram represents the flow of logic for a batch-scoring scenario. A data build combines operational data with external data systems to create one master model input data that can serve multiple models. In a large organization, there might be many master data tables created. Model scores are then loaded into offline or online systems for efficient access. For high throughput needs, SAS supports concurrent execution through use of MP/Connect software.

For on-demand systems such as front office application-processing and call centers, the Message Queue (MQ) capabilities of the SAS® Integration Technologies product can be used to asynchronously service model-scoring requests. The scoring client forms the MQ data packet and sends it to the MQ server. The scoring server retrieves the MQ data packet, executes the corresponding score code, and posts the output scores to the MQ server. The client then retrieves the indexed output from the MQ server and continues processing. Message Queue systems are common in many industries. Figure 5 illustrates this case.

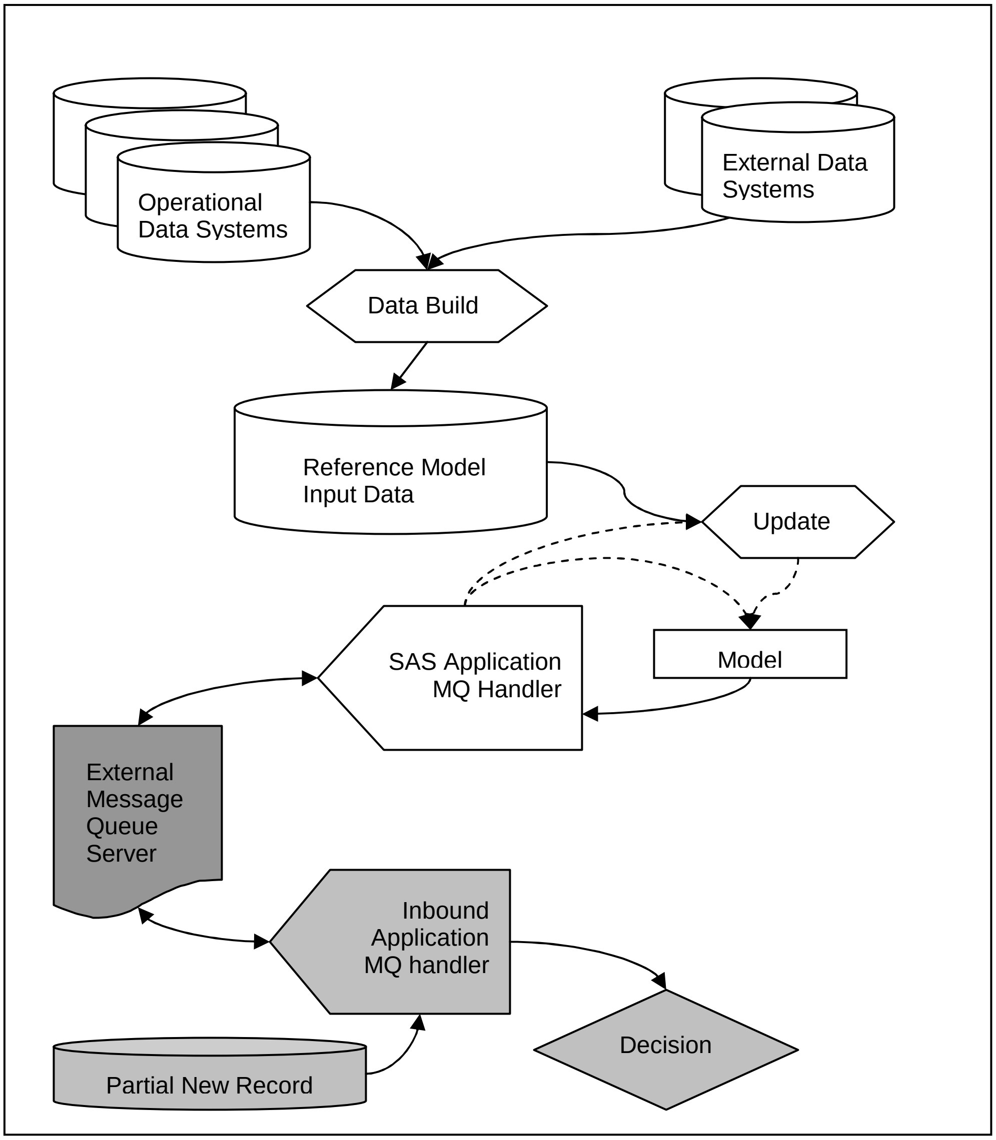

Figure 5. On-Demand Scoring. In this case, Message Queue technology is employed to provide a single record of data to the SAS system for model scoring. Optionally, a reference model input table can be used to provide data fields that are not provided by the client application. In that case, the SAS server updates the full data record with the new partial data record, runs the score code, and returns the score or resulting decision to the client application through the asynchronous MQ server. This design is appropriate for many on-demand scoring applications.

The most important point here is that by using the SAS system, a single-score code program can service all of these environments: ad-hoc, batch, and on-demand. Because the program does not have to be recoded and tested in a new execution environment, organizations can save valuable deployment time. Models that are put into production faster will show a better ROI than models that take longer to deploy. SAS Model Manager provides tools to help modelers and IT staff deploy predictive models to multiple environments.

MONITOR THE MODEL

The final stage is to manage the model in a production environment. This includes moving the executable to an operational platform, running model performance reports, distributing the generated reports, and revalidating the model. Periodically, the life cycle of the model repeats itself with the data management stage when the customer base has evolved sufficiently to require new sample data, or the factors that contribute to predictability have changed.

A champion predictive model is deployed for a period of time in production environments. Its predictive performance often degrades over time. A champion model needs to retire when its performance degradation hits a certain threshold. Therefore, model monitoring tasks should be routinely done to identify underperforming models in time to prevent problems caused by obsolete models. When there could be so many active production models, the following question arises: “How does an organization manage model performance monitoring tasks in an effective and efficient manner?” A good model monitoring tool can help to answer this question. The tool should:

- **Automate the monitoring tasks.** It should be possible to schedule model monitoring tasks to run at a variety of time periods.

- **Support portable performance indexes.** Performance indexes should be portable in the sense that they can be applied to many models. These indexes can then be used to define performance degradation thresholds. For example, SAS Model Manager provides the following portable performance indexes: lift5Decay, lift10Decay, lift15Decay, lift20Decay, and so on. For a more specific example, the current lift of the top 5% for model A is 4.5. Is the lift value good or bad? Well, you don’t know unless you also know the top 5% model lift value (for example, 5.0) computed on the development test partition. Lift5Decay, in this particular case, is equal to (5.0-4.5)/5.0. Because lift5Decay is a portable performance index, it can be used by many models for performance monitoring purposes.

- **Allow user-defined multi-level degradation thresholds.** Users should be able to define multiple levels of performance thresholds so that warning notifications are generated before alert notifications. (Note: An alert condition is more serious than a warning condition.) If users receive a warning notification regarding a model performance and build a new champion model in response to correct the model degradation problem, then they might not receive alert notifications at all down the road when the new replacement champion is deployed.

- **Notify interested parties when at least one threshold is crossed.** A good model monitoring tool should notify users via email, for example, when at least one user-defined threshold is crossed.

- **Log execution events of monitoring tasks.** The events of model monitoring tasks should be logged so that the entire model monitoring process is traceable.

- **Track input and output variable distribution shifts.** In addition to tracking model performance, the tool should also help detect the variable distribution shifts for both target and predictor variables. A well-thought-out tool should make sure that the binning definitions for a continuous variable are reused for comparing variable distribution and that outlier effects for binning are avoided or reduced.

- **Provide monitoring trend charts or reports.** Charts showing how the model degradation evolves over time are critical for the model monitoring process. The lift chart in Figure 6 shows how model lift degrades over time.

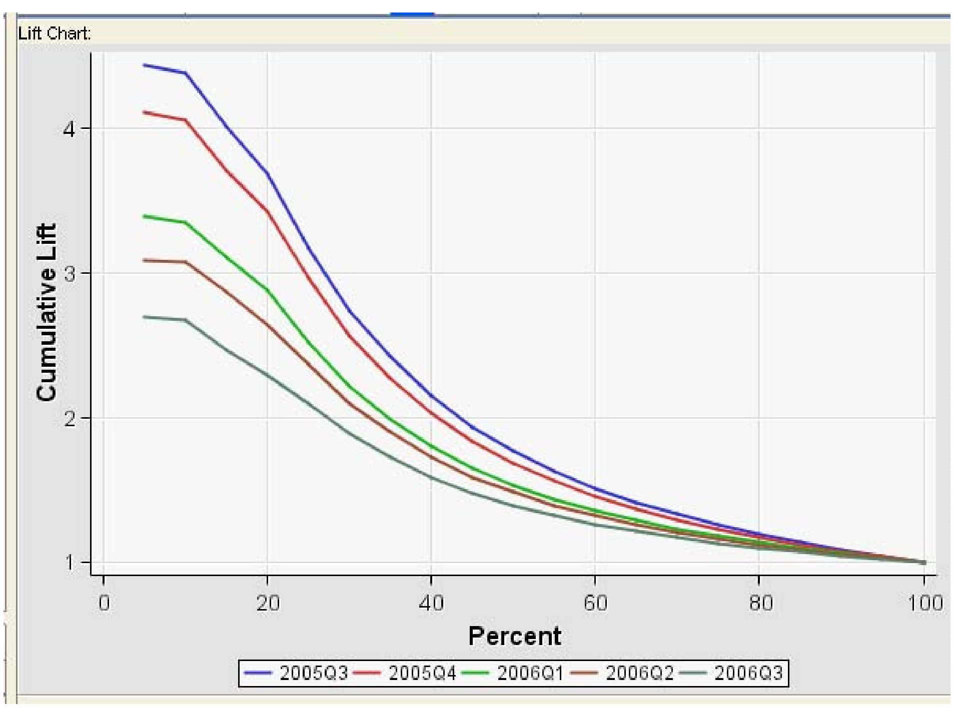

Figure 6. Model Degradation Lift Chart

CONCLUSION

Predictive models are important corporate assets. More and more predictive models are created by all sorts of organizations to gain competitive advantages. Managing such predictive models becomes a critical business process. This paper addresses general model management issues and introduces what the new product SAS Model Manager can do to facilitate model management processes. However, this paper is not intended to cover the entire feature set of SAS Model Manager 2.1. For more information about SAS Model Manager, please check the Model Management and Deployment Web site at www.sas.com/technologies/analytics/modelmanager/index.html.

REFERENCES

Basel Committee on Banking Supervision. 2005. "Update on work of the Accord Implementation Group related to validation under the Basel II Framework." Basal Committee Newsletter No. 4. (Accessed March 3, 2007) Available at www.bis.org/publ/bcbs_nl4.htm.

Comptroller of the Currency. Administrator of National Banks. 2000. OCC 2000-16. OCC BULLETIN. (Accessed March 3, 2007) Available at www.ffiec.gov/ffiecinfobase/resources/retail/occ-bl2000-16_risk_model_validation.pdf.

ACKNOWLEDGMENTS

Appreciation is extended to the entire SAS data mining and model management family (Development, Testing, Technical Support, Education, Publications, Marketing, and Field Strategy/Support) for helping bring this product to market. We also appreciate Tonya Balan, Kathleen Walch, and Mike Boyd for reviewing the paper. We are also especially grateful to our development partners for the great feedback and testing of the product. Thank you.

RECOMMENDED READING

SAS Institute Inc. 2004. SAS Institute white paper. "A Strategy for Fast Deployment of SAS® Models". Cary, NC: SAS Institute Inc.

Svolba, Gerhard. 2006. Data Preparation for Analytics Using SAS®. Cary, NC:SAS Press.

UNISYS. 2005. UNISYS White Paper. "SAS® Parallel Scoring Optimization." Cary, NC: UNISYS and SAS Institute Inc.

Wielenga, Doug. 2007. "Identifying and Overcoming Common Data Mining Mistakes." *Proceedings of the SAS Global Forum 2007 Conference*. Cary, NC: SAS Institute Inc.

CONTACT INFORMATION

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

Robert Chu, Senior Software Manager
SAS Institute Inc.
S6120 SAS Campus Dr.
Cary, North Carolina 27513
Work Phone (919) 531-7696
Email: robert.chu@sas.com

David Duling, Development Director
SAS Institute Inc.
S6102 SAS Campus Dr.
Cary, North Carolina 27513
Work Phone: (919) 531-5267
Email: david.duling@sas.com

Wayne Thompson, Product Manager
SAS Institute Inc.
S6100 SAS Campus Dr.
Cary. North Carolina 27513
Work Phone: (919) 531-7696
Email: wayne.thompson@sas.com