2011

Automated self-service modeling: predictive analytics as a service

Kridel, Don; Dolk, Daniel

Springer-Verlag

http://hdl.handle.net/10945/62012

Inf Syst E-Bus Manage (2013) 11:119–140
DOI 10.1007/s10257-011-0185-1

ORIGINAL ARTICLE

Automated self-service modeling: predictive analytics as a service

Don Kridel · Daniel Dolk

Received: 27 September 2010 / Revised: 18 May 2011 / Accepted: 19 November 2011 /
Published online: 13 December 2011

Abstract Research into service provision and innovation is becoming progressively more important as automated service-provision via the web matures as a technology. We describe a web-based targeting platform that uses advanced dynamic model building techniques to conduct intelligent reporting and modeling. The impact of the automated targeting services is realized through a knowledge base that drives the development of predictive model(s). The knowledge base is comprised of a rules engine that guides and evaluates the development of an automated model-building process. The template defines the model classifier (e.g., logistic regression, multinomial logit, ordinary least squares, etc.) in concert with rules for data filling and transformations. Additionally, the template also defines which variables to test ("include" rules) and which variables to retain ("keep" rules). The "final" model emerges from the iterative steps undertaken by the rules engine, and is utilized to target, or rank, the best prospects. This *automated modeling* approach is designed to cost-effectively assist businesses in their targeting activities—independent of the firm's size and targeting needs. We describe how the service has been utilized to provide "targeting services" for a small to medium business direct marketing campaign, and for direct sales-force targeting in a larger firm. Empirical results suggest that the *automated modeling* approach provides superior "service" in terms of cost and timing compared to more traditional manual service provision.

D. Kridel
Department of Economics, University of Missouri-St. Louis,
St. Louis, MO, USA
e-mail: kridel@umsl.edu

D. Dolk (✉)
Department of Information Sciences, Naval Postgraduate School,
Monterey, CA, USA
e-mail: drdolk@nps.edu

Keywords Customer targeting service · Predictive analytics · Service-oriented architecture · Model management · Automated modeling

1 Introduction

Service science management and engineering (SSME) is an emerging research discipline which reflects the dramatic shift in the US and world economy from product-based manufacturing to service-based employment. Interest in the area of service sciences continues to flourish within the academic community, especially in information sciences and marketing (see e.g., Dietrich 2006; Dolk 2008; Maglio et al. 2006; Rai and Sambamurthy 2006; Rust and Miu 2006; Spohrer and Riiecken 2006; Vargo and Lusch 2004). A research domain of particular relevance to the IS community is the design of service systems leveraging Web-based service-oriented architectures. To date, there has been relatively little in the literature linking model-based decision support systems (DSS) to SSME. We describe a service system based upon the *model as a service* concept (Bhargava et al. 1997) that helps bridge this gap by providing an automated modeling capability for customer targeting.

Models as a service derived from the *software as a service* (*SaaS*) movement in the 1990's in which the view of software shifted from being a product to being a service. Typically, SaaS is deployed as server-based commercial software provided as a service to customers across the Internet. Frequently SaaS is often conjoined with grid or cloud computing, which conjoined as a *platform as a service* can be used as a distributed computing environment for executing software. *Models as a service* (*MaaS*) is a subset of SaaS which focuses upon analytical modeling software such as modeling languages, model solvers, and data visualization tools for analyzing results. Analytical modeling applications in this case might entail optimization, regression, decision analysis, and Monte Carlo simulation models, for example.

We refine this distinction even further to consider *predictive analytics as a service* (*PAaaS*) whereby *predictive analytics* (sometimes referred to as *data analytics* or *decision analytics*), we mean a form of data mining which applies statistical analysis to historical data in order to predict future trends and behavior patterns. Predictive analytics is widely used in contemporary business intelligence and customer relationship management (CRM) applications such as credit scoring, customer retention, market basket (cross-sell) analysis, and fraud detection. Predictive analytics can be a powerful business tool but requires significant statistical expertise in order to be done properly. This expertise may not be readily available for all firms, especially small-to-medium businesses (SMBs).

We describe an analytics service which helps bridge this gap by using automated modeling techniques which we have developed in previous research. One of the key characteristics of such a service is the recognition that the end-user clients may know little, if anything, about the model details, whereas the model developers may be unfamiliar with the particular datasets which the model may be called upon to process. To accommodate this disparity in knowledge, we layer the analytics service into *deep structure* versus *surface structure* partitions: a deep structure knowledge

base-driven model development engine to build surface structure model templates, and an automated model workflow process to manage the surface structure requirements of specific users.

From a service-providing firm's perspective, it is increasingly important to "innovate in service provision" and it is reasonable to suppose that much of this innovation is likely to be driven by the automation (or digitization) of services. Our model-driven service system can be used to replace and/or augment part of a firm's current customer targeting process. Customer targeting (hereafter referred to as simply "targeting") has been shown to be effective in increasing response rates for various marketing campaigns (Edmiston and Kridel 2007; Kridel and Dolk 2004; Rossi et al. 1996; Shaffer and Zhang 1995). We demonstrate in this paper that targeting through automated analytics significantly outperforms more traditional methods. This improved performance relates primarily to the ability to model at a much-lower geographic level than has previously been attempted. Further, our "model as a service" can be "internal" providing targeting for in-house initiatives or "external" as a Web-based service sold to a firm's clients to support their targeting.

2 Motivation for predictive analytics as a service

Predictive analytics refers to data mining procedures which use statistical techniques such as multiple regression to make forecasts in support of managerial decision-making. Data mining is typically conducted using software packages which access data warehouses or data marts containing historical and/or cross-sectional information culled from an organization's operational data sources. These warehouses may be augmented by external data sources either publicly available (e.g., economic indicators from the Bureau of Labor statistics), or commercially available (e.g. demographic data from companies such as Experian). Well-known data mining software includes SAS Enterprise Miner™, SPSS Clementine™, and IBM Cognos™.

Effective predictive analytics requires a significant degree of statistical modeling expertise coupled with a thorough understanding of the data which is being used as the foundation for modeling. Several process models for conducting predictive analytics have been proposed, including SAS' SEMMA (Sample, Explore, Modify, Model and Assess), the CRISP-DM (Cross-Industry Standard Process for Data Mining), and the Two Crows model (2005). Table 1 shows the stages of the Two Crows process model which we aggregate into three higher level categories of Problem, Data, and Model.

One of the implications of Table 1 is that substantial organizational resources are required for conducting predictive analytics. Critical success factors include an established database infrastructure as well as modeling expertise in the form of statistical analysis and the use of data mining software to perform that analysis. Our motivation for predictive analysis as a service is that there exist many organizations for whom these resources are not available and for whom there is limited or no modeling capability. Our proxy for modeling capability is organizational size where

Table 1 Process stages for predictive analytics (adapted from Two Crows Corporation 2005)

Predictive analytics stage	Aggregated stage
1. Define the business problem	Problem
2. Build a data mining database	Data
3. Explore the data	Data
4. Prepare data for modeling	Model
5. Data mining model building	Model
6. Evaluation and interpretation	Model
7. Deploy the model and results	Model

our assumption is that the smaller an organization, the less likely it is to have resources available for meaningful predictive analytics.

Figure 1 shows a notional service horizon curve for the aggregated predictive analytics stages across three categories of modeling capability represented as organizational size (small, medium, large). The area under the curve represents existing predictive analytics capabilities whereas the area above the curve represents capabilities out of the reach of the respective organizations. The graph shows small businesses as having minimal or limited data and model capabilities for predictive analytics, medium businesses as having slightly more advanced capabilities in these dimensions and large businesses as having the potential for the most capabilities. The motivation for predictive analytics as a service is to move the service horizon curve to the left and upwards, providing increased modeling capabilities, particularly for small and medium businesses. The focus of our paper is to show this can be done for a specific direct marketing (DM) application, customer targeting.

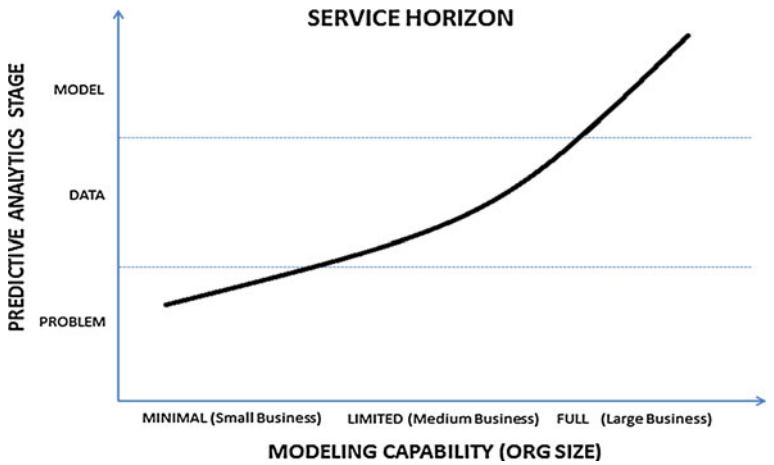

Fig. 1 Predictive analytics service horizon

3 Modeling context: customer targeting

In simplest terms, business targeting is the process employed to determine who to "contact". In much the same way an archer must aim before shooting, a firm must decide who to contact before establishing contact. Firms encounter the targeting problem in a variety of marketing settings:

- Acquisition: the firm wants to convert prospects into new customers;
- Up- or cross-sell: the firm wants existing customers to buy "bigger" (more or different) products and services;
- Response analysis: the firm wants to contact prospects that are more likely to respond positively to a given offer;
- Retention: the firm wants to entice existing customers to "continue" as a customer or prevent the customer from "churning", or canceling service.

In each case, the firm needs a way to limit the number of contacts (or cost) in order for these initiatives to be profit enhancing. For example, an acquisition campaign that contacted every household in any major US city would surely yield additional customers; just as surely, firm profits would fall as the cost of contacting every city-resident would more than the offset the increased revenues from the new customers. This is the essence of the targeting problem: find prospects that are expected to be the most responsive to a given offer (or contact). Beyond who to contact, targeting can also be used to determine "how" to contact (DM, phone, e-mail, mobile advertising, etc.) and "what" to offer (which product, what bundled services, which specific offer, etc.). In the present context, we will limit the discussion to whom to target.

For new customer acquisition, information about existing customers is used to predict which prospects (non-customers) are most likely to become customers if given the opportunity. For up-sell (or cross-sell), information on customers that have behaved in a certain way (e.g., bought a large bundle of services) is used to rank existing customers that have not behaved in that way. Likewise, in the retention case, data from former customers is used to determine which of the current customers are most likely to churn.

3.1 Existing service provision

Targeting typically looks quite different in different-sized firms; further, targeting may mean different things to different companies. Smaller firms generally target informally, while some larger firms dedicate significant resources to targeting.

For SMBs, targeting is typically reduced to intuition or guesswork as these firms are not large enough to afford modelers or modeling services. Typically, the SMB or the firm's DM agent will either

- Purchase "intuitive or dumb lists of names" through a count-and-order system (e.g., firm buys 100 names from a particular set of zip codes with a particular set of demographic characteristics), or

- Purchase names from one of many providers of "specialty lists" (e.g., if target is expectant mothers, there are list providers that specialize in lists comprised of expectant mothers).

For larger firms, more targeting options are available. A firm can utilize count-and-order and specialty-list options as the SMB does, but also may employ modeling staff and/or use consulting services to develop specific targeting models.

Targeting services are generally provided on a manual basis—that is, models are developed by analysts, the models are used to score targets, "smart lists" are generated from those scores (rankings), and finally DM campaigns are undertaken to contact customers on these "smart lists" (fulfillment). Parts of this targeting process have begun to be automated, e.g., "intuitive lists" and fulfillment (creative, printing, and mailing) are available via the web.

Normally, the process begins with internal data collection. Frequently this requires analysts to "pull" their own data (via SQL for example) from internal databases at the company. Once collected, the data will need to be cleansed and organized. Additional exogenous data (e.g., customer demographics) will often need to be purchased and merged into the customer dataset. For many modeling projects, the "pull and cleanse-merge" process may consume as much as one-half of the overall time required to develop the model. This is particularly true for acquisition campaigns since the firm will generally not have a list of non-customers (prospects); this list of prospects will generally be purchased along with the demographic data. A high-level summary of a typical modeling process is provided in Fig. 2.

3.2 System design principles

Our objective is to automate the above process with special focus upon its most complex aspect, namely the modeling dimension. This requires a more flexible approach to model building and execution than has typically been deployed in conventional analytical modeling and predictive analytics. Specifically, we need to be able to rapidly specify and solve on-demand, context-sensitive models using "smart" model formulation techniques that identify data variables, relationships and functional form "on the fly".

Models relevant to predictive analytics may range from simple techniques such as trend analysis or exponential smoothing to more sophisticated models such as logistic regression, maximum likelihood estimation, and neural networks. Commercial software systems such as SPSS Clementine™, SAS Enterprise Miner™ and IBM's Cognos™ provide portfolios of these techniques, and can be thought of as engines for building predictive models. However, building robust predictive models requires advanced levels of statistical expertise and sophistication for which many firms, particularly small to medium businesses, may not have the requisite resources even if they have the underlying engines.

A good model-specific delivery system for predictive analytics must take into account the basic dichotomy between an end-user and an analytical modeler. In our environment, the end user client is assumed to be domain knowledgeable but not model knowledgeable. Modelers, on the other hand, are assumed to be model

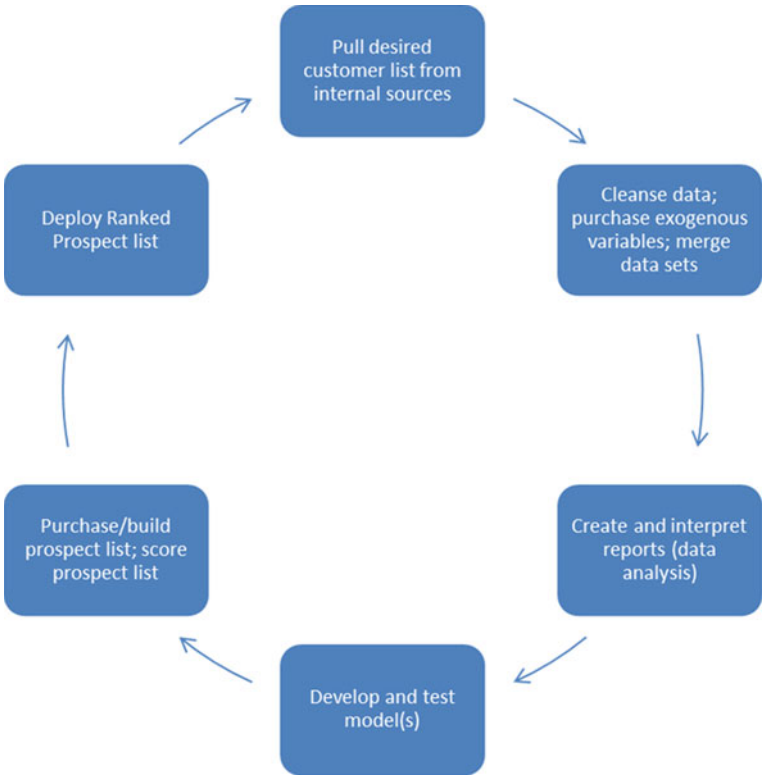

Fig. 2 Representative manual modeling work flow for targeting

knowledgeable but not necessarily domain knowledgeable. Thus the client will be applying dynamically generated models, specifically discrete choice logistic regressions with which they are likely not familiar, to data sets with which they are familiar. The modeler's challenge is to generate models with statistical integrity for datasets which are unknown a priori. This, in effect, requires an automated modeling functionality which in turn requires the ability to generate statistically robust and valid models "on the fly". This dimension of dynamic configurability is implemented as a service-oriented architecture based upon design principles emanating from the distinction between deep and surface structures in modeling systems.

3.3 Deep structure versus surface structure

Deep structure and surface structure are linguistic constructs developed in Chomsky's seminal work on generative, or transformative, grammars (Chomsky 1957). Deep structures are stable foundational schemas which change little if at all, whereas surface structures are various instantiations of deep structure which may take many different forms. This conceptual framework has corollaries to

information system design (Wand and Weber 1995). In relational database systems, for example, schemas defining the table structures remain relatively stable and may be said to comprise the database's deep structure, whereas the fields in the database tables themselves are constantly undergoing modification and are therefore part of the surface structure. From a design perspective, it is critical to identify the deep structure characteristics of a system from the outset, and to model them as accurately and effectively as possible. Changing deep structure artifacts after the fact tends to be very expensive compared to changing surface structure constructs. For example, altering the underlying database schema is a much more costly and complex operation than simply changing one or more values of a data field in a corresponding table.

This linguistic metaphor has also been effectively applied to the design and development of analytical models (Geoffrion and Maturana 1995). In this context, deep structure refers to the components of a model which remain relatively stable and exhibit low volatility during the model lifecycle, whereas surface structure embodies those aspects of a model with relatively high volatility. In the case of optimization models which the authors use as an example, the minimal mathematical representation of the model comprises the deep structure whereas the data values, index set contents and derived intermediate outcome variables constitute descending levels of surface structure from higher to lower. Two normative design relationships apply: (1) Volatility—the more volatile a component, the easier it should be to change (and vice versa); and (2) Modularity—changes to components at any level of structure should not affect components at the next lower level. Thus, adding or deleting data values should not affect the index set contents, changes in which, in turn, should not affect the outcome variables, and so on.

We adopt the analytical model version of the deep structure paradigm to the domain of predictive analytics in order to design a flexible model-based service. There are three levels of structure in our automated modeling system: data surface structure (we do not consider database schemas), model surface structure, and model deep structure (Table 2). The overall process of data mining and predictive analytics typically occurs in environments with very high levels of data volatility. Data surface structure is defined by the user who provides the relevant client databases, possibly augmented by proprietary national databases such as Experian™. Model surface structure is less volatile than data surface structure but nevertheless can be quite dynamic compared to more conventional modeling applications. For example, as datasets change, the "goodness of fit" of a particular model may deteriorate over time. The functional form of a regression model may need to be altered in a dynamic fashion to "keep up" with the data stream of updated transactions. Additionally, it may be necessary to add or remove independent variables from the model to keep it "in synch" with the current dataset. Thus, we have developed a dynamic automated model builder that is driven by a knowledge base of rules governing model formulation. Our model builder constantly monitors the match between the existing data environment and the current executable model to ascertain the integrity of its statistical inference. If and when a change is mandated, the automated model builder will formulate a new model that more accurately reflects the parameters of the evolving dataset. Thus,

Table 2 Structure hierarchy for automated model system

Component	Structure level	Volatility
+Data	Surface	High
Model instances	Mid-surface	Medium to high
Knowledge base (model schema)	Deep	Low

this model surface dimension is less volatile than its data surface counterpart but nevertheless quite dynamic in nature. The true deep structure in our system is the knowledge base which captures expertise about building econometric forecasting models, specifically discrete choice logistic regressions, which are statistically coherent and relevant. The knowledge base guides which variables to include in the logistic regression as well as what functional form should be adopted.

3.4 Automated service-provision process

The implementation of our deep versus surface structure partition is shown in Fig. 3 which is a high-level schema of the workflow underlying the automated process. The process begins with customer list from a client. Depending on the how the service is provided, the customer list may come from a CRM system (e.g., SalesForce.com), a user-created view of the customers of interest within the system itself (for enterprise clients) or uploading a file directly by the user (typical for SMBs). The list is matched (and geo-coded) to a national database to obtain demographics or firmographics depending on whether the firm sells to consumers (B2C) or other firms (B2B).[1] The modeling process (driven by the modeling knowledge base) develops a dynamic logistic regression model which is used for scoring. The ranked prospect list, along with profiles and maps, are available for downloading and/or on-line viewing. For some self-service portals, the user can also be "fulfilled" (e.g., the firm can select and mail the desired DM piece to the provided list).The same basic workflow is employed for larger clients who interact through a richer GUI with their data already loaded and matched. However, from the service provision perspective, the essence of the automated targeting process is unchanged.

At a summary level there is very little difference in Figs. 2 and 3. In the first step, a customer must be collected; this step is essentially identical in both the automated and manual processes.[2] Some of the list-cleansing will still need to be done in the automated process (as part of the first step); the remaining cleansing activities along with the merging of lists will be automated (and removed from the list of activities

[1] For SMB file—uploads and CRM "on-the-fly" targeting, Fig. 2 displays the basic workflow. For larger enterprise clients, the customer data would already be loaded (first two steps would be complete). In the enterprise case, the user would create a customer segment of interest (rather than upload it); otherwise, the process would be displayed as in Fig. 2.

[2] In the case of an enterprise installation, the automated process would ease subsequent segment creation for subsequent modeling projects. For the initial model, however, there is little difference between the automated and manual processes.

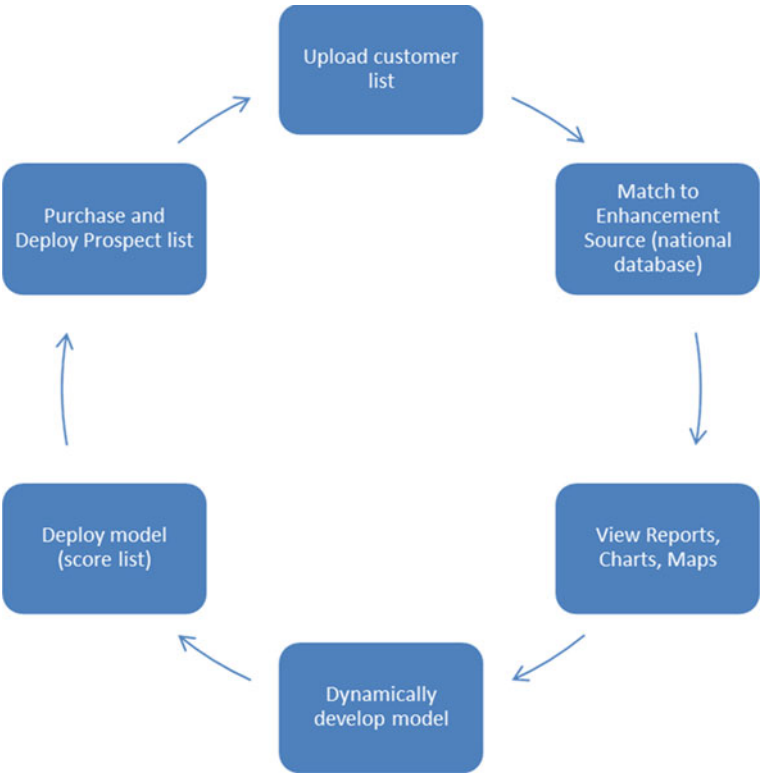

Fig. 3 Basic automated work flow for targeting (SMB or self-serve version)

required of the user). Assuming the user employs the same software for merging, cleansing, and reporting (e.g., SAS), report creation and presentation will be similar in both processes, though the user likely has more control in the manual process. Model development in the manual process requires expert statistical training and significant time; this process is totally automated and much faster (minutes vs. days or weeks) in the automated process.[3] Scoring (ranking) is similar in both processes though the automated process will offer some time-savings as typical selection-processes are incorporated that would have to programmed specifically in the manual process (e.g., select randomly from the top five demi-deciles). Deploying the list will be virtually identical in both processes although the "purchase" happens at different stages: during data-set building (merging) in the manual process and at deployment in the automated process.[4]

In Fig. 3, surface structure activities occur at each step except the modeling process (Step 4). All data manipulation activities are surface structure including the

[3] Of course, the deep structure embedded in the modeling templates requires this same expert statistical training, but the automated system allows "re-use" of this training by other users.

[4] Purchase terms may differ but typically for manual models the entire set of 0's (along with demographic data) would need to be purchased before modeling while for the automated process only the actionable ranked list is purchased for deployment.

match between the client's database and the national database. Only the model generation process accesses deep structure. In the following section, we delve into the deep structure knowledge base and describe in detail how it generates dynamic logistic regressions to adapt to the client's requirements.

4 Automated modeling workflow

There are many challenges associated with providing an automated workflow that builds dynamic models for potentially untrained statistical users. Besides the usual interface and presentation issues associated with presenting model results to non-technical users, there is the important task of model-building itself. In particular, these templates need to flexible enough to handle a wide range of situations, but specific enough to deliver useful models to real users. Kridel and Dolk (1991) describe automated dynamic model-building techniques derived from research in the areas of active decision support and automatic model generation (Castillo et al. 1991; Dolk and Kridel 1993; Manheim 1988) to generate context-specific customer lists.[5]

Building on the model manipulation language developed by Dolk and Kridel (1991, 1993) for the active DSS or the "symbiotic econometrician", the modeling template extends the features of conventional modeling systems. While the modeling templates are not learning-based, the templates do provide services consistent with Manheim's notion of an active DSS, i.e., "a DSS which can usefully do more than its users explicitly direct it to do" (El-Gayar and Deokar 2011). In much the same way that the active DSS "takes over" from the user to help solve a specific econometric estimation problem, the modeling template "takes over" for the user and provides the user with a "high-level business service" rather than a specific "estimation task". Parts of the active DSS modeling process (Dolk and Kridel 1991) are contained explicitly in the workflow via the modeling template (e.g., simulation is effectively replaced by step-based and model-based rules). Other processes (e.g. the inference processor) are explicitly built into the specific modeling template by the statistical expert. As such, there is a premium on statistical training and practical system experience for the specific template builders. Nonetheless, this expertise is then "available" to non-technical users and is delivered in the form of useful targeted lists. For SMBs especially, this knowledge transfer is a valuable service as there are no practical (cost-effective) alternatives.

It should be noted that parts of this overall analytic-workflow can be obtained from other providers:

- A variety of list providers offer "targeted leads" either through direct sales or via the web;
- Automated modeling is now available through data-mining tools generally offered by statistical software providers (e.g., SAS Enterprise Miner);

[5] Kridel and Dolk (2003) provides examples of system artifacts and a simplified version of a modeling template (in flow-chart form).

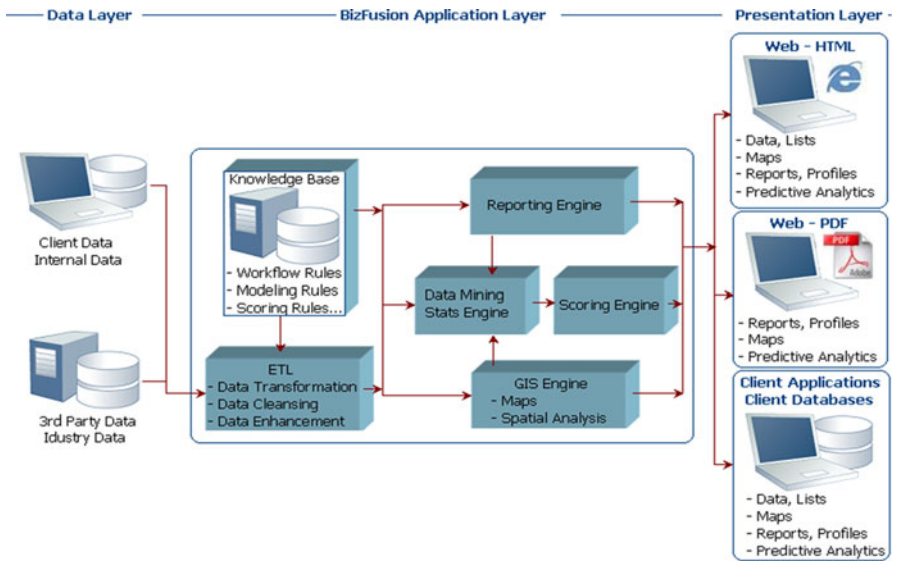

Fig. 4 Automated modeling workflow (from Kridel and Dolk 2009)

- Consulting firms offer end-to-end analytic services for targeting (and other marketing problems).

As a practical matter, these piece-parts offer little to SMBs (and even some larger firms). The targeted lists are little more than guesses or hunches based on intuition; the general poor performance of these lists is one of the main drivers for the demand for analytic services. SMBs are unlikely to have access to either the budget or skill-set required to utilize the data-mining tools. Further, even if the SMB had access, data preparation and deployment of model results would typically be well beyond their analytic resources of the typical SMB. Consulting services are simply too expensive and too time consuming to be useful on an on-going basis.

Figure 4 details the architecture of the work-flow process. The (deep structure) knowledge base contains sets of "rules" embedded in templates that are used to develop the (surface structure) model(s). The knowledge base is used to build executable model templates which are instantiated dynamically from client-supplied data. The template contains rules for data-filling and transformations, "include" rules for deciding which variables to test in model development, and "keep" rules for deciding whether the tested variables remain in the final model.

Examples of "include" rules for categorical variables are:[6]

- ALWAYS: always test this variable in the model;
- PLI (Purchase likelihood index) based rules: e.g., $80 < \text{PLI} > 120$ or $\text{PLI} > 125$; the PLI is derived from the univariate analysis that is performed as part of the reporting service prior to model development;

[6] Note these include rules can apply to individual categories (within the variable) or to all categories (within the variable).

- Count: allows exclusion of small sample sizes by either counts or percentage-counts or the proportion of missing and/or filled;
- Some combination of the above (as multiple rules are allowed), e.g., a variable must meet PLI and Count rules to be considered in the model.
- For continuous variables, the "include" rules allow testing of functional form, e.g., linear, quadratic, logarithmic, etc.

There are three types of "keep" rules:

1. variable-based rules that apply to specific variables:[7]
   a. ALWAYS: always leave variable in the model
   b. Statistically based rules: e.g., $-2 > t\text{-stat} > 2$, or $t\text{-stat} > 1.65$
2. step-based rules that apply to specific model steps:
   a. likelihood ratio tests (is Step 3 a statistically significant improvement over the existing model from Steps 1 and 2)
   b. likelihood improvement (is model at some %-amount better than previous step)
3. model-based rules that apply to the entire "model":
   a. size-based rules (e.g., are any coefficients, *t*-statistics, or elasticities too "large"?)
   b. rules based on the hold-out samples (e.g., are training and hold-out lift charts "different"?)

Model templates may contain as many steps as desired and each step may contain a virtually unlimited number of variables. For example, one of the standard retail templates includes five steps and approximately 50 variables (of which five are continuous). "Standard" modeling templates (e.g., retail acquisition) are available through the system by default; custom templates may be built—either because of statistical "preferences" of the client or because the situation is unique (often when internal data from the client is being utilized in addition to data from the national database). For self-service clients, typically SMBs, the "default" templates are utilized since the necessary expertise to alter templates is likely not available (and customer lists would typically contain only name and address). Similarly, the modeling templates for enterprise clients are richer since these customer lists typically include internal RFM (Recency of purchase, Frequency of purchase, and Monetary amount) variables. The modeling template "fully controls" the development of the dynamic model-building process.

In much the same way the symbiotic econometrician (Dolk and Kridel 1991) had to interpret what the modeler was attempting and assist in problem-solving, the

[7] These are slightly more general in the sense that the rules can be applied for the entire variable (or by components of the variables). In the case of categorical variables, the rule can be applied to all categories (keep all if any are significant) or by category (keep only the categories that satisfy the keep rule). For continuous variables, the same options can be applied (keep all functional terms if any pass keep rules or keep only the functional terms that pass keep rules). In addition, the keep rules can be applied at each step or only in the current step.

modeling templates need to drive the estimation and provide "protection" for less sophisticated users and/or unusual data sets. For example, consider a modeling template for mobile response campaigns. This template would likely contain attributes of the subscription, attributes of the mobile device, attributes of the campaign itself and attributes of the subscriber. This type of modeling template could be quite effective for a general mobile campaign. Suppose, however, that a user attempts to model a campaign directed only to iPhone users. In this case, the attributes of the mobile device could lead to model failure as near-perfect multicollinearity would occur (as all observations would possess the same attributes for the mobile device variables). In some cases these types of problems can be handled directly when loading the data by checking for variation in the variables. Many times, however, there is enough variation in the data (through coding errors for example) that the simplest methods (e.g., max = min, if yes, do not load variable) will not effectively eliminate the problem. It is specifically for these types of issues that the model and step keep rules were added; careful design of modeling templates assists in this regard. For example, adding these variables in later steps (and grouped in a single step) will ease the detection issues associated with problems of this type.

To demonstrate how the process works, we consider a highly stylized acquisition modeling template that contains three separate modeling steps:

1. Standard demographics (income, age, length of residence)
2. Lifestyle segments or clusters (e.g., MOSAIC codes)[8]
3. More-esoteric demographics (presence of children, previous mail responder).

Further, assume for ease of discussion that each variable in the first modeling-step has five categorical values, that there are 50 MOSAIC segment-codes, and that each of the variables in the third modeling-step is binary (yes/no). For this simple example, assume that the modeling template has PLI and Count%-based include rules for every variable and that each variable has a standard *t*-stat keep-rule.[9] Table 3 displays the stylized modeling template.

The model process begins by using the customer list and generating a compliant non-customer (prospect) list. While configurable, the reference prospect list is generally generated by geography, e.g., for every zip code where there is a customer, select all other households in that zip as prospects. In essence, this step "builds" the binary dependent variable (the 1's are current customers, the 0's are the prospects or would-be customers).

A market penetration report is developed for the eight variables included in the modeling template, a portion which is displayed in Table 4.

[8] MOSAIC is a geo-demographic segmentation system developed by Experian. (There are several alternative segmentation systems available from other vendors.) Each of the nearly 250,000 block groups in the US are categorized into one of 12 groups which can be further broken down into 60 segments (MOSAIC codes), e.g., A01 = America's Wealthiest, A02 = Dream Weavers, etc.

[9] When there are continuous variables, the process also tests for functional form; that is, does the variable enter as linear, quadratic, logarithmic, etc. The test may be based on categorical equivalents (the "shape" of the univariate report) or likelihood-based tests based on Box-Cox transformations.

Table 3 Summarized modeling template

Variable	Include	Include type	Keep	Keep type	Step	Model
LOR	90 < PLI > 110; %Count > 3	Ind category	−2 > *t*-stat > 2	Ind category this step	None	None
Income	95 < PLI > 105; %Count > 3	Ind category	−2 > *t*-stat > 2	Ind category this step	None	None
Age	95 < PLI > 105; %Count > 3	Ind category	−2 > *t*-stat > 2	Ind category this step	None	None
MOSAIC code	90 < PLI > 110; %Count > 2.5	Ind category	−2 > *t*-stat > 2	Ind category each step	None	None
Presence of children	90 < PLI > 110; %Count > 3	Ind category	−2 > *t*-stat > 2	Ind category this step	None	None
Previous responder	PLI > 110; %Count > 3	Ind category	*t*-stat > 2	Ind category this step	None	None

Table 4 Market pen report for length of residence (LOR) variable

LOR	Description (years)	Customer #	Customer (%)	Prospect #	Prospect (%)	PLI
A	A = <1	43	15.58	39,145	9.60	162.3
B	B = 1–5	190	68.84	256,825	63.30	108.8
C	C = 5–10	30	10.87	75,490	18.60	58.4
D	D = 10–15	3	1.09	16,893	4.20	25.9
E	E = >15	10	3.62	17,377	4.30	84.3

The model builder then queries the modeling template for the include rules. For each categorical value (e.g., A) of each variable included in the template (e.g., LOR), the rules engine checks which variables "pass" the include rules. In this case, categories A, C, and E would pass the include tests (B is excluded by the PLI rule and D is excluded by the size rule). The same procedure for income, age, MOSAIC, presence of children and previous mail-responder would be utilized.

The "starting point" for the model would include all categories from all variables that satisfied these include rules. Assume that four income and age categories, 25 MOSAIC codes, and both binary variables satisfied the include rules. The initial regression for Step 1 would contain 11 variables: four income categories, four age categories and three LOR categories. After this logistic equation has been estimated, the keep rules are applied. If all 11 variables satisfied their respective keep rules, the process would proceed to Step 2. On the other hand, if only eight of the eleven variables satisfied the keep rules, additional iterations in Step 1 would be processed. Typically (though this is also configurable via the modeling template), the least significant variable (lowest absolute *t*-statistic) would be dropped and the logistic equation would be re-run with ten categorical variables. This process would continue until all variables satisfied their respective keep-rule. In Step 2, the 25 MOSAIC variables would be added to the "working" equation that resulted from Step 1. The same process would be employed: run iterations until all remaining

variables satisfy their respective keep-rules. Step-rules (if in place for Step 2) would then be employed; for example, a likelihood ratio test would determine whether Step 2 was "kept" or the model reverted back to the Step 1 variable list. At this point, Step 3 variables would be added with the same iterative process employed. Finally, once all variable and step rules have been satisfied, model rules (if specified in the template) would be employed. At this point, the model would be complete (or final) and ready for deployment.[10] The final model would be utilized to score or rank all prospects with the top ranking prospects qualifying as the best candidates for being contacted. The system can recommend the number of prospects to target (based on the scoring results) or the users can select the number of prospects (presumably based on available budget).

5 Service-oriented architecture

The automated workflow system for targeting is based upon a service-oriented architecture (SOA) summarized in Fig. 5.[11] The system is designed around a variant of the model-view-controller paradigm. The view represents a visualization of the model; the model is the object being manipulated, and the controller is the manipulator. The controller consists of a set of independent services. The controller engages specific (lower level) services to manipulate the behavior and structure of the model. In this way, lower-level services are utilized through the workflow to provide desired actions (higher-level services).

The basic services are organized as follows:

- administration: permissions, authentication, etc.
- data management: usual data-base and data-uploading functions
- analytics: statistics and regression [logistic, multinomial logit (MNL), and ordinary least squares (OLS)]
- scoring: using mode(s) to rank or score candidate prospects
- counts: segment creation and counting
- mapping: simple mapping functionality (points-of interest, areas-of-interest, etc.)
- reporting: generates customer and prospect reports

The targeting service, described in the previous section, is comprised of more basic reporting, modeling, and scoring services. In this way, a business service is delivered to end-users through the workflow. We note that this is a very specialized implementation of the more general architectures for service-based Web-based model management as described, for example, in Bhargava et al. (1997), El-Gayar and Deokar (2011).

[10] This assumes the model test was satisfied; if not, the final model would be altered based on the model rules in place and parts of the iterative process would be repeated.

[11] The targeting platform was initially called *Bizfusion*, a proprietary product of CopperKey, Inc. In 2009, KAST acquired the IP and renamed the service. Since the original submission of this article, KAST has been acquired by Adenyo, Inc. and the platform will likely be renamed once again.

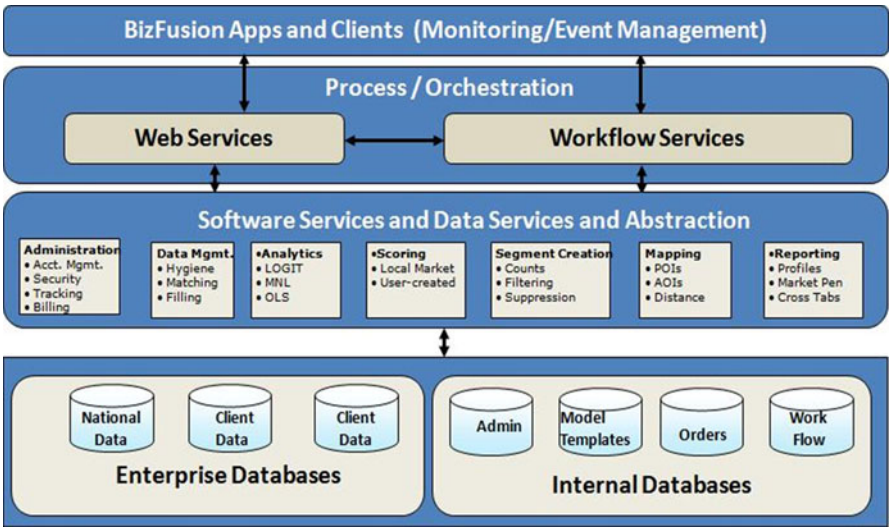

Fig. 5 Service-oriented architecture (SOA) for targeting workflow system

6 Evaluation of the system

6.1 Statistics

Evaluation of service systems under variability is also a critical research issue in SSME (Kannan and Proenca 2008). In addition, there are two other service provisioning issues to be addressed: (1) does the ASP-delivery of analytic services meet customer expectations; and (2) does the automated workflow of analytics "work"?

With respect to service provision, there seems to be limited satisfaction with ASPs (at least anecdotally). Susarla et al. (2003) identify a conceptual framework for modeling and evaluation of customer satisfaction for ASP services. Like any ASP, PAaaS could benefit understanding and delivering said services in a manner that would meet (or exceed) customer expectations. Nonetheless, in the present case, we will focus on whether the outcome of the PAaaS process "works". In particular, two questions will be addressed:

(1) Does the system provide service in the form of viable "clients"?
(2) Are the services provided useful in an economic sense?

While increases in the former would seem to imply economic utility, a more formal evaluation of (2) is provided.

With regards to (1), the service system seems to be gaining traction in a wide variety of service-provision settings. Experian provides a private-labeled enterprise version of the automated-modeling platform (branded as Business Market Analyzer). They have recently added several new clients from retail, regional banking, and financial and marketing services. These firms typically utilize the

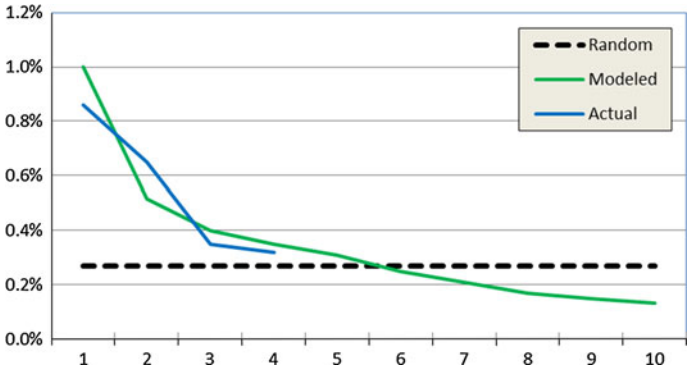

Fig. 6 Actual versus predicted response rates (from Edmiston and Kridel 2007)

service for internal targeting, in other words, targeting services to augment (or replace) current targeting practices for acquisition and retention. AT&T Yellow Pages is using the platform for "internal" targeting services as discussed below.

Kridel and Dolk (2004) report on a case study that compares "dumb list via count-and-order selection" and "smart list obtained from automated modeling portal" for an SMB. The lift generated by the models is about 560% (7.8% vs. 1.4%). The immediate direct ROI is 120% (vs. −90% for the "intuitively selected list").[12]

Edmiston and Kridel (2007) describe in some detail the process that AT&T employed to determine whether to replace its manual modeling process with the automated process described here. The authors report that ROI exceeds 3,000% (the calculation includes *only* the direct cost of the vendor's contract—not the total Cost of Goods Sold). During 2005 and 2006, response rates increased on average approximately 285%. Typically, this meant market conversion rates increased from the 0.25–0.5% range to the 0.75–1.5% range. Figure 6 compares, for deciles that were contacted, the actual and predicted response rates from the automated model for a test market. For this market the actual lift was 2.02 while the predicted lift from the model was 2.10. The results indicate not only that the model performed well, but that actual performance was very close to predicted performance.

6.2 Limitations and generalizability of the system

An obvious limitation of our analytics service implementation is that it only treats a single class of discrete choice models, namely logistic regression, for the specific application domain of customer targeting. In the world of predictive analytics, there are many statistical techniques available to the modeler, for example, decision trees, discriminant analysis, neural networks, multiple regressions, cluster analysis and genetic algorithms to name just a few. When and how to apply and interpret these

[12] It is worth noting that the modeling profile was used for the "dumb or intuitive" selects. As a result, the "intuitive" list is better than it would have been had the business owner simply had to guess at the appropriate selects.

techniques for different application contexts is a distinct challenge, requiring experienced model analysts. In our automated modeling environment, it is absolutely essential that the model templates comprising the deep structure of the system be developed by highly knowledgeable analysts. In the general case, expert modelers may try many different techniques to see which performs the best for a particular dataset.

In our system, the methodology, or technique, is fixed and the knowledge base guides the eventual formulation of the model structure as a logistic regression. To support a more general exploratory data analysis, one might envision a portfolio, or library, of such rule bases, each developed for determining a statistically robust model, or class of models, for a specific technique. This would free the modeler, at least to some degree, from determining model structure, and instead allow focusing upon the comparison of various techniques.

Devising a rule base to guide this kind of general exploratory analysis is significantly more difficult than the narrowly circumscribed knowledge base we have described. For this situation, what is required is a higher level conceptual model and associated language, an experimental design model (EDM) and language (EDL) that allows modelers not only to specify, solve and compare multiple model structures and methodologies, but also to translate the results into model templates similar to what we have constructed for the targeting domain. Experimental design capabilities are conspicuously absent in the analytical software domain, not only in statistical analysis but also in optimization and other classes of management science and operations research modeling. Although scripting languages are available which can be adapted by skillful programmers and modelers to create and execute experimental designs, these languages lack an overarching conceptual model for experimental design which seriously limits their generalizability.

Several data mining languages have been proposed or implemented which could serve as the basis for research into EDM/EDL (see for example, DMQL (Han et al. 1996), XML–DMQL (Feng and Dillon 2005), TDML (Muthukumar and Nadarajan 2007), Microsoft's SQL Server-based data mining language, DMX (Data Mining Extensions (DMX) Reference. Microsoft Corporation. http://msdn.microsoft.com/en-us/library/ms132058.aspx), and the predictive modeling markup language, PMML (Guazzelli et al. 2010) created by the Data Mining Group). Many of these languages are extensions of SQL and carry the benefit of being directly applicable to relational databases and obviating the need to preprocess these databases for data mining applications. However, SQL is too restrictive and lacks the expressiveness for the robust kinds of experimental design processes that we envision as necessary to move to the next level of automated modeling.

The EDM/EDL modeling environment we envision would be the equivalent of a higher level knowledge base for conducting more comprehensive predictive analytics across a wider range of applications. Without proceeding any further into the details of conceptual models or modeling languages, we suggest that the rule base component comprising the deep structure of our system could be expanded through the use of an EDM and associated EDL to encompass a more general and powerful approach. In this scenario then, the EDM/EDL would constitute an even deeper structure than the knowledge base and extend the structure hierarchy by

Table 5 Structure hierarchy for automated model system with EDM/EDL

Component	Structure level	Volatility
Data	Surface	High
Model instances	Mid-surface	Medium to high
Knowledge base (model schema)	Mid-deep	Low to medium
Experimental design model and language (EDM/EDL)	Deep	Low

another level (Table 5). One of our current research objectives is to develop this conceptual model for experimental design.

7 Conclusions

We have described an automated self-service modeling system for targeting which leverages predictive analytics to put statistically robust and sophisticated discrete choice models in the hands of "model naïve" clients. We achieve this by embedding the required statistical expertise in a deep structure knowledge base which drives the dynamic formulation and application of configurable surface structure predictive models to client databases. This SOA provides a model-driven targeting service which, in preliminary trials, has shown significant increases in lift over traditional manual service provision.

Spohrer et al. (2007) describe service systems as "collections of resources that can create value with other service systems through shared information." Further, they suggest that "open" service systems are

1. capable of improving the state of another system, and
2. capable of improving its own state by acquiring external resources.

The *predictive analytics as a service* targeting workflow system we describe satisfies both aspects of this definition in the sense that the workflow interacts with other systems (both internal and external) to provide improved targeting which, in turn, leads to increased revenues and/or decreased costs. For example, in an enterprise setting the targeting workflow would augment and enhance existing BI and CRM systems to improve targeting models. As a result, customers receive fewer "nuisance contacts", more "offers of interest", or both.

Although not a perfect substitute for full-time modeling, our *model as a service* does significantly reduce the amount of time required for a full-time analyst to target a market effectively in more detail than would otherwise be possible. It further shows the promise of shifting our view of decision and model technologies from commodities to services as suggested by Bhargava et al. (1997). This not only better positions modeling technologies with respect to service-based applications, but also holds the potential for extending the utility and relevance of analytical and computational models.

References

Bhargava H, Krishnan R, Muller R (1997) Decision support on demand: emerging electronic markets for decision technologies. Decis Support Syst 19:193–214

Castillo D, Dolk D, Kridel D (1991–1992) GOST: an active modeling system for costing and planning NASA space programs. J Manag Inf Syst 8(3):151–169

Chomsky N (1957) Syntactic structures. Mouton Press

Dietrich B (2006) Resource planning for business services. Commun ACM 49(7):62–64

Dolk D (2008) Introduction to decision technologies and service sciences track. In: Proceedings of the 41st Hawaiian international conference on system sciences, January 2008. IEEE Computer Society

Dolk D, Kridel D (1991) An active decision support system for econometrics. Decis Support Syst 7:315–328

Dolk D, Kridel D (1993) Towards a symbiotic expert system for econometric modeling, Chap. 7. In: Blanning RW, King DR (eds) Current research in decision support technology. IEEE Computer Society Press

Edmiston E, Kridel D (2007) Automated modeling and sales targeting: case study for AT&T advertising and publishing. In: Proceedings of DMA07 (Direct Marketing Association conference and exhibit), Chicago, IL, October 2007

El-Gayar O, Deokar A (2011) An ontology-based model management architecture for service innovation. In: Dolk D, Granat J (eds) Modelling for decision support in network-based services. Lecture Notes in Business Information Processing. Springer, Berlin (to appear)

Feng L, Dillon T (2005) An XML-enabled data mining query language: XML-DMQL. Int J Bus Intell Data Min 1(1):22–41

Geoffrion A, Maturana S (1995) Generating optimization-based decision support systems. In: Proceedings of the 28th Hawaii international conference on system sciences. IEEE Computer Society

Guazzelli A, Lin W, Jena T (2010) PMML in action: unleashing the power of open standards for data mining and predictive analytics. CreateSpace

Han J, Fu Y, Koperski K, Wang W, Zaiane O (1996) DMQL: a data mining query language for relational databases. In: Proceedings of the ACM SIGMOD workshop on research issues on data mining and knowledge discovery, June 1996

Kannan P, Proenca J (2008) Design of service systems under variability: research issues. In: Proceedings of the 41st Hawaiian international conference on system sciences, January 2008. IEEE Computer Society

Kridel D, Dolk D (2003) An on-line marketing consultant for small and medium businesses. In: Proceedings of the 32nd WDSI conference, Lihue, HI, April 2003

Kridel D, Dolk D (2004) Using intelligent profiling to generate smart lists: an empirical test. In: Proceedings of the 33rd WDSI conference, Manzanilla, Mexico, April 2004

Kridel D, Dolk D (2009) A self-service automated targeting portal: an example of model as a service. In: Proceedings of the 38th WDSI conference, Lihue, HI, April 2009

Maglio P, Srinivasan S, Kreulen J, Spohrer J (2006) Service systems, service scientists, SSME, and innovation. Commun ACM 49(7):81–85

Manheim M (1988) An architecture for active DSS. In: Proceedings of the 21st Hawaiian international conference on system sciences, vol III, January 1988. IEEE Computer Society, pp 356–365

Muthukumar A, Nadarajan R (2007) TDML: a data mining language for transaction databases. Fourth international conference on fuzzy systems and knowledge discovery, pp 81–86

Rai A, Sambamurthy V (2006) Editorial notes—the growth of interest in services management: opportunities for information systems scholars. Inf Syst Res 17(4):327–331

Rossi P, McCulloch R, Allenby G (1996) The value of purchase history data in target marketing. Mark Sci 15(4):321–340

Rust R, Miu C (2006) What academic research tells us about service. Commun ACM 49(7):49–54

Shaffer G, Zhang Z (1995) Competitive coupon targeting. Mark Sci 14(4):395–416

Spohrer J, Riiecken D (2006) Services science. Commun ACM 49(7):31–34

Spohrer J, Maglio P, Bailey J, Gruhl D (2007) Towards a science of service systems. Computer 40(1):71–77

Susarla A, Barua A, Whinston AB (2003) Understanding the service component of application service provision: an empirical analysis of satisfaction with asp services. MIS Q 27(1):91–123

Two Crows Corporation (2005) Introduction to data mining and knowledge discovery, 3rd edn. http://www.twocrows.com/intro-dm.pdf
Vargo SL, Lusch RF (2004) Evolving to a new dominant logic for marketing. J Mark 68(1):1–17
Wand Y, Weber R (1995) On the deep structure of information systems. Inf Syst J 5:203–223