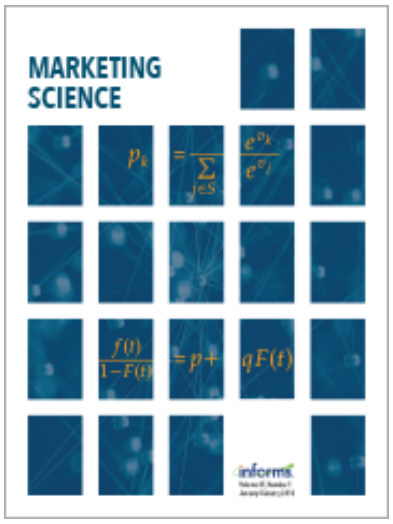


Marketing Science

Publication details, including instructions for authors and subscription information:
http://pubsonline.informs.org

Prediction in Marketing Using the Support Vector Machine

Dapeng Cui, David Curry,

To cite this article:
Dapeng Cui, David Curry, (2005) Prediction in Marketing Using the Support Vector Machine. Marketing Science 24(4):595-615.
https://doi.org/10.1287/mksc.1050.0123

MARKETING SCIENCE
Vol. 24, No. 4, Fall 2005, pp. 595–615
ISSN 0732-2399 | EISSN 1526-548X | 05 | 2404 | 0595

informs

DOI 10.1287/mksc.1050.0123

Prediction in Marketing Using the Support Vector Machine

Dapeng Cui
Ipsos Insight, North America, 111 North Canal, Suite 405, Chicago, Illinois 60606,
dapeng.cui@ipsos-na.com

David Curry
College of Business Administration, University of Cincinnati, Cincinnati, Ohio 45221-0145,
david.curry@uc.edu

Many marketing problems require accurately predicting the outcome of a process or the future state of a system. In this paper, we investigate the ability of the support vector machine to predict outcomes in emerging environments in marketing, such as automated modeling, mass-produced models, intelligent software agents, and data mining. The support vector machine (SVM) is a semiparametric technique with origins in the machine-learning literature of computer science. Its approach to prediction differs markedly from that of standard parametric models. We explore these differences and benchmark the SVM's prediction hit-rates against those from the multinomial logit model. Because there are few applications of the SVM in marketing, we develop a framework to position it against current modeling techniques and to assess its weaknesses as well as its strengths.

Key words: automated modeling; choice models; kernel transformations; multinomial logit model; predictive models; support vector machine
History: This paper was received January 8, 2003, and was with the authors 14 months for 2 revisions; processed by Duncan Simester.

1. Introduction

Many marketing problems require accurately predicting the outcome of a process or the future state of a system. In the past two decades, prediction of consumer choice has attracted the most attention, but prediction of other marketing phenomena plays a fundamental role in modern marketing practice and is essential to accomplish the deeper goals of marketing science. Examples include predicting segment membership, most "switchable" customers, most effective ad, and website navigational choices. This paper presents a systematic study of the strengths and weaknesses of a methodology well suited for a variety of prediction environments in marketing, the support vector machine (SVM). Accurate prediction, though essential, is often hindered by complex relationships between predictor and target variables and an absence of theory to guide model identification. For reasons explained in this research, the support vector machine predicts accurately in such environments.

The support vector machine is a semiparametric technique with origins in the machine-learning literature of engineering and computer science. There are currently no applications of the support vector machine reported in the marketing literature and only one application reported in any major business journal; i.e., Viaene et al. (2002). The SVM is novel, but as Bucklin et al. (2002) emphasize, with today's diverse data sets, "...it may be counterproductive to rely primarily on standard statistical methods. Emphasizing scalable methods and predictive results may enable us to observe a richer set of behavioral phenomena in (...) marketing data."[1]

1.1. Objectives

Though relatively unstudied in marketing, the support vector machine has demonstrated its utility in a variety of other disciplines, including statistics, computer science, agriculture, and engineering. The present research clarifies the strengths and weaknesses of the SVM for the kinds of applications, traditional and future, most relevant to marketing scientists. This goal is accomplished through a combination of analytic discourse and direct empirical comparisons with the multinomial logit model (MNL).[2] In marketing, the logit is the gold standard;

[1] The text has been altered slightly to emphasize marketing data, not just clickstream data, as is the focus of the Bucklin et al. (2002) passage.

[2] We employ several versions—conditional, nested—of the multinomial logit model and recognize the variety of forms that this

it is widely applied and is known to perform well. It provides a well-understood benchmark for the SVM.[3] However, in later sections of the paper, we emphasize the complementary, not competitive, relationship between the SVM, MNL, and other existing models in marketing.

Although marketing scientists have traditionally emphasized structural understanding over predictive accuracy, in many emerging marketing contexts there is a notable absence of theory, and prediction—not structural understanding—is the primary goal. These contexts include automated modeling, intelligent agents, and data mining, to name just a few. In the present study, we engineer environments with properties like those expected in these and other areas relevant to marketing's expanding scope. These environments require individual-level predictions, but data are not collected using a controlled experimental design. More likely, the data are pure one-shot field data that mix information about the prediction target, the individual, and the prediction context.

Such contexts represent a proving ground for the support vector machine. The SVM behaves mathematically in a way that avoids overreliance on particular structural assumptions. It implicitly automates the model identification process, and by so doing, enters the parameter estimation phase with a family of structural possibilities rather than a single possibility. The SVM uses a kernel-induced transformation from the original *attribute space* to a higher-dimensional space to capture relevant features of the data. Through clever use of kernel transforms, the SVM solves a nonlinear problem with a linear model.[4] The approach has certain stability and robustness characteristics lacking in the maximum-likelihood procedure, whether based on actual or simulated likelihood.[5] These characteristics also control overfitting. Thus, in this paper, the term "prediction" always means out-of-sample prediction, not goodness-of-fit.

class of models takes, as well as the sophisticated estimation methods being used and under development. For simplicity, we use the terms "logit model," "logit," or "MNL" to refer to this class of models. In subsequent sections, the precise nature of the models used in this research is clarified.

[3] A simple (OLS) version of the support vector machine has already been shown to outpredict a variety of "soft-computing" techniques, including artificial neural nets, *k*-nearest neighbor, decision-tree, Bayesian learning multilayer perceptron, and tree-augmented Bayes (Viaene et al. 2002).

[4] Kernel transformations are explained briefly in the main text and more thoroughly in Appendices A and B.

[5] Today's newer estimation techniques using the Gibb's Sampler (Allenby et al. 1995, Hofstede et al. 2002), simulated likelihood (Kamakura et al. 2003), hierarchical Bayes (Rossi and Allenby 2003, Andrews et al. 2002), and hybrid logit kernel models overcome this problem to some extent.

1.2. Omitted Topics

This research is defined as much by what it omits as by what it includes. Readers interested in a technical exposition of the support vector machine are directed to Burges (1998). For an exposition in a business-oriented setting, see Curry and Cui (2003). This paper only superficially considers statistical learning theory, the theory of statistical inference underpinning the SVM (Vapnik 1998). We highlight the fundamental result linking machine (model) capacity, sample size, and empirical risk. The bibliography contains abundant references for readers interested in further study of this important topic. Finally, we focus exclusively on one-dimensional, discrete prediction, not predictors with continuous outcomes or 2-*d* or higher classification. The SVM has been successfully applied to 2-*d*-type problems, including face detection (Osuna et al. 1997) and image classification (Chapelle et al. 1999). It has also been adapted to problems with continuous outcomes, such as regression problems (Mattern and Haykin 1999, Müller et al. 1999, Schölkopf et al. 1999, Stitson et al. 1999), principle components analysis (Schölkopf et al. 1999), and density estimation (Weston et al. 1999). However, these areas are beyond the scope of the present research.

The remaining sections of this paper are organized as follows. Section 2 reviews the critical role of prediction in marketing and provides a framework for organizing predictive models. The SVM's position in this framework helps identify the types of problems for which it is well suited. Section 3 outlines major areas where the SVM differs from conventional models. The section addresses topics vitally important to practitioners and modelers alike—including model capacity, boundary bias, and the curse of dimensionality—to build a case for the SVM's superior predictive capability. Section 4 details how and why an SVM works and indicates how we implemented the SVMs used in this research. Section 5 presents the methodology used for the major experimental comparisons presented in this paper, testing the SVM in a variety of environments where it is baselined against the multinomial logit. Section 6 discusses weaknesses of the SVM and shortcomings of our empirical comparisons. Section 7 presents two main directions for future research, both stressing the complementary nature of the SVM and random utility theory (RUT) models. Section 8 offers concluding comments. Appendices A, B, and C in the main text explain technical elements of the SVM, while two online appendices offer additional detail for interested readers (http://mktsci.pubs.informs.org).

2. Prediction in Marketing

More than a half-century ago, Politz and Deming (1953, p. 51) noted the vital role of predictive models

in marketing: "Every (marketing) decision entails the expectation of a specific result. Therefore, every decision, if it is rational, depends on prediction." Because so much is riding on prediction in marketing, taking care to draw the problem in the most refined way possible increases the likelihood of obtaining useful results. To this end, we distinguish four major contexts in which accurate prediction is paramount for marketing success: (a) *pure prediction*, (b) *robust prediction*, (c) *analytic prediction*, and (d) *structural gap analysis*. These categories position the support vector machine against current methods dominant in marketing.

2.1. Pure Prediction

Pure prediction includes all cases in which structural understanding of a phenomenon is neither feasible nor necessary. Management's mission is entirely practical, not intellectual, because a course of action must be determined or a value computed for a large number of cases. In these contexts, the firm can realize higher revenues, lower costs, or both, by using predictive models. In addition to the burden of too many cases to model "by hand," relevant predictors may change from one case to the next, so that human intervention is not only costly, but not useful. These properties define several important frameworks discussed by marketing scholars and implemented to varying degrees by many firms today, including mass-produced models, automated modeling, data mining, intelligent agents, and certain subdomains of other marketing areas, such as direct marketing.

Mass-produced models have already made important contributions in the consumer packaged-goods (CPG) industry, and their importance is likely to expand during the next decade (Blattberg et al. 1994). Similarly, automated models, inspired by IRI's Cover Story™ (Schmitz et al. 1990) and PROMOTER™ (Abraham and Lodish 1987, 1993) are being improved and upgraded (Bucklin et al. 1998, Little 2001). With both mass-produced models and automated modeling, the emphasis is on pure prediction precisely because there are too many cases to consider individually. Although predictions are often based on fairly simple models, they are consistent and "accurate enough," relative to the alternative of human intervention.

Data-mining applications are prominent in direct marketing (Berry and Linoff 1997, Cooper and Giuffrida 2000) and have rapidly expanded in the CPG environment. In CPG applications, data mining emphasizes predictions about the future coincidence of items in a shopping cart, rather than structural understanding or inference (Bucklin et al. 2002). In direct marketing, data-mining algorithms maximize expected profit from solicitations (Ratner 2003), including cross-selling opportunities (Kamakura et al. 2003).

Although the agent framework is relatively new in marketing, a number of marketing scientists are actively developing agents and exploring their consequences (Ariely et al. 2004, Avery et al. 1999, Diehl et al. 2003, Gershoff and West 1998, Häubl and Trifts 2000, Iacobucci et al. 2000, West et al. 1999). West and colleagues (1999) foresee a wide range of applications for consumers, many of which require accurate prediction of utility, satisfaction level, and price sensitivity on a client-by-client basis with information sets that are unique to each client and change over time.

2.2. Robust Prediction

The second major marketing context where prediction is a key factor falls under the general rubric of marketing engineering, e.g., the "systematic process of putting marketing data and knowledge to practical use" to enhance decision-makers' mental models (Lilien et al. 2002). In marketing engineering, prediction is only part of the equation, though a crucial part. We call this *robust prediction* because, paradoxically, the structural accuracy of the model is not as important as its ability to provide the decision maker with an accurate sense of trends, outliers, boundary conditions, and other elements of the "big picture." In fact, *robust* means structurally naïve but predictively accurate. The models in a marketing management support system (MMSS) (Wierenga et al. 1999, Wierenga and van Bruggen 2000) are valued precisely because they provide intuitive understanding, not scientific explanation (Hunt 1983).

Conjoint analysis is a good example of robust modeling. In hundreds of commercial applications and nearly all published research, no structure deeper than additive is sought for an individual's utility function. The additive model is robust (Dawes and Corrigan 1974); its predictions are accurate enough, despite the fact that many studies of human information integration reveal a wide assortment of noncompensatory behavior (Einhorn 1970, Kahneman 2002, Kardes 1999, Brazerman 1994, Russo and Shoemaker 1989, Tversky 1972). The success of additive conjoint analysis emphasizes that management clearly accepts the trade-off between deep structural understanding and ease of interpretation in light of respondent effort.

2.3. Analytic Prediction

Analytic prediction describes contexts in which a model is developed to solve a particular problem or class of problems or where an existing model is adapted for this purpose. In many cases the model is just beyond inclusion in the robust prediction category, but could end up there as its structure becomes better understood and its robustness a proven commodity. Recent examples include the model of Moe

and Fader (2001) to predict the conversion rate from website visit to online purchase and Sismeiro's and Bucklin's (2004) model to predict how long a visitor will view each webpage on a given website.

Discrete-choice models that operationalize RUT are included in this category. Although these models yield important structural insights, it is worth recalling that in both the transportation and marketing literatures, discrete-choice models were originally designed to support "if-then" analysis for policy implications, a purpose for which accurate prediction is a necessity (Domencich and McFadden 1975). In this spirit, Guadagni and Little (1983) thoroughly tested the out-of-sample predictive ability of the multinomial logit they fit to household scanner data.[6] They emphasize the model's application as a market response tool to assist managers in anticipating market reactions to changes in marketing-mix variables.

Discrete-choice models offer both structural insight and accurate prediction. Regarding structure, however, the main evolutionary path has been to refine structural understanding of the stochastic component of the model, not to seek additional structure in the model's deterministic component. Substantial progress has been made to unpool variance due to consumer heterogeneity and other specialized components, such as correlated attributes. This purifies estimates of the parameters in the deterministic component of the model, adding power to significance tests and enhancing analyst confidence in the magnitude and sign of estimates. However, with rare exceptions, $V_{ij} = \mathbf{X}_{ij}\boldsymbol{\beta}$ remains linear in attributes even though modern applications of discrete-choice modeling venture far beyond the simple choice contexts envisioned by Luce (1959) and Thurstone (1927). Today's models mix attributes of the choice options with attributes of the customer and the choice environment. It is reasonable to expect interactions and higher-order effects between attributes from different domains, e.g., product attributes and decision-maker attributes. Thus, we argue that there are potential gains from including additional structural complexity in V_{ij}, not just in error.

2.4. Prediction and Structural Gap Analysis

Marketing science demands highly accurate predictive models precisely because such models push analysts to increase their structural understanding of a process. Although unambiguous profit motives drive the previous three cases, here financial gain is subordinated to the search for scientific understanding. In his thesis of structural identity, Hempel (1965, p. 367) addresses the idea squarely: "(1) every adequate explanation is potentially a prediction, and (2) every adequate prediction is potentially an explanation." Social scientists sometimes defend a model by claiming that it is designed to explain rather than predict a phenomenon. Hunt bluntly calls this excuse vacuous, since "all adequate explanations must have predictive capacity" (1983, p. 117).

We shall see that the support vector machine, because it predicts so well, has the potential to play a central role in a deeper understanding of marketing phenomena. This role takes the form of presenting the marketing scientist with a prediction gap that leads unequivocally to a structural identification gap. Foreshadowing results, the support vector machine's predictive superiority in nonlinear contexts signals omitted structure in the baseline model. Of course, when this structure is included, the baseline model can predict up to the inherent limits of uncertainty. This is big news, not because the increased predictive power was unexpected, given perfect structural insight, but rather because perfect structural insight is impossible. Finding a reliable technique like the SVM that provides a reasonable target for predictive accuracy supports positivist goals.

Not all techniques used for prediction in marketing fit neatly into the foregoing framework. Examples include scenario analysis, capabilities analysis, and game theory. The SVM fits easily into the class of pure predictors, but has properties, reviewed next, that suggest potential contributions in all four categories of the framework.

3. Model Properties and Predictive Ability

The relative strengths and weaknesses of the SVM can be established by probing four fundamental properties that influence the predictive ability of any model. These properties—decision boundary, structural capacity, boundary bias, and empirical risk—are tightly interconnected. They all contribute to a fifth area, recognition of complexity that involves scientific attitudes and positivist tenets that can stymie methodological progress. We discuss each of these areas in turn to build an understanding of differences between the support vector machine and parametric models in marketing.

3.1. Implicit Decision Boundaries

All parametric models for discrete prediction, including those based directly on the general linear model (GLM), such as discriminant analysis; and those based indirectly on GLM, such as discrete-choice models; involve two basic stages. Stage 1, function estimation, yields continuous decision boundaries that are

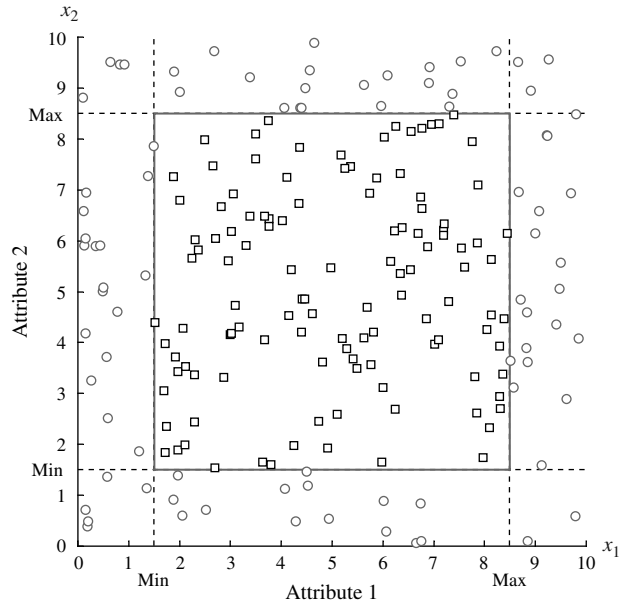
[6] They tested predictions on a new sample (in the same time period), in a new time period (with the original sample), and in single stores (though pooled parameter estimates were derived from data aggregated over stores).

then triggered in Stage 2 when an observation is assigned to a particular discrete class. In the simplest two-outcome, logit model, for example, the boundaries are contours of constant probability of belonging to Class 1, "choose option A," versus Class 2, "choose something else." The statistical prediction rule is based on one of these contours, typically the contour corresponding to 50/50 posterior odds. The logit approximates this contour by using hyperplanes in the space spanned by the predictor variables. The linear combination of predictors is transformed to lie between 0 and 1 in order to approximate probabilities.

Discrete-choice models with richer structure, such as conditional, nested, and mixed; or hierarchical logit models, place additional constraints on these hyperplanes, but do not alter the most fundamental property of the resulting decision boundary; it is functional, not relational, in nature. Thus parametric models face a structural inadequacy when dealing with a *relational* boundary between classes. In consumer choice, such a boundary arises even with very simple information integration rules. For example, if a consumer uses the *Latitude of Acceptance* rule shown in Figure 1, acceptable options (squares) are nested within a ring of unacceptable options (circles) defined by cutoffs on two attributes (West et al. 1997).

Recognizing a nested relation mathematically is difficult for a parametric model based on function approximation. To illustrate, consider the prediction hit-rates for a simple binary logit model versus an SVM for the data in Figure 1. We estimated parameters of both models using a sample of size $l = 400$, then used the model to predict the outcome for $N = 50{,}000$ observations drawn from the same joint density. Over 10 replications of this exercise, the logit predicts on average 51.7%, while the SVM averaged 97.7% correct predictions. The true odds are 49/51, indicating that the binary logit cannot beat chance.

Figure 1 Latitude of Acceptance Model



There are three basic reasons for the disparity, and these reasons exert their influence interactively. First, the structural capacity of the two models differs substantially. Second, the SVM skips the function estimation step and calculates the optimal decision boundary directly, as described subsequently. Finally, the SVM establishes this boundary while preserving degrees of freedom through the application of a unique data transformation that, contrary to common practice, increases rather than decreases the dimensionality of the solution space. How this works and why it should work well are clarified in the following sections.

3.2. Machine Capacity

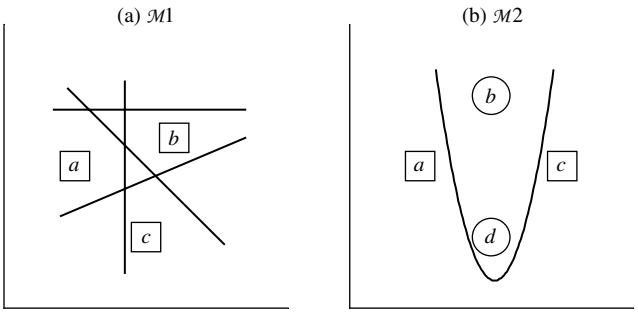
Let us elaborate the *model as machine* analogy through the idea of a machine's *capacity*. Capacity is akin to the Fisherian notion of structural error caused by model misspecification (Fisher 1950), but is indexed quantitatively in the theory of the support vector machine. Consider the following four functional forms for a regression model.

$$\mathcal{M}1:\ y = \beta_0 + \beta_1 X_1,$$
$$\mathcal{M}2:\ y = \beta_0 + \beta_1 X_1 + \beta_{(2)} X_1^2,$$
$$\mathcal{M}3:\ y = \beta_0 + \beta_1 X_1 + \beta_2 X_2,$$
$$\mathcal{M}4:\ y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_{12} X_1 X_2.$$

These forms differ in certain explicit properties such as *number of inputs* and *number of parameters*. Several of these models are nested; e.g., $\mathcal{M}1 \subset \mathcal{M}2$ and $\mathcal{M}1 \subset \mathcal{M}3 \subset \mathcal{M}4$. Among nested models, we naturally think of the super-model as having greater capacity than a submodel in the sense that it must exhibit superior *in-sample* fit. For nonnested models with the same number of parameters, their relative capacity is not as clear. For example, $\mathcal{M}2$ and $\mathcal{M}3$ have the same number of parameters, but $\mathcal{M}2$ involves only one input while $\mathcal{M}3$ involves two. In such cases, our intuitive notion of capacity is confounded by econometric questions such as multicollinearity, e.g., X_1 versus X_1^2. These questions in turn engender others about the nature of the data collection process (field data or experimental data; i.e., are X_1 and X_1^2 orthogonal?), and whether the likelihood or simulated likelihood surface is unimodal, a property that can strongly influence the quality of estimation.

Rather than use nesting, degrees of freedom, or more generally, an object-oriented assessment of observable model properties, Vapnik and Chervonenkis (1964, 1971) focused on an unobservable but more fundamental property from which these others follow. This property, known as the *VC-dimension* of

Figure 2 **Capacity to Shatter 2^p Points**



a model, is based on the concept of *shattering*. A function is said to shatter a set of p points if it can be instantiated so as to subdivide the points into all 2^p possible subsets. The function's *capacity is the maximum number of points it can shatter*. For example, three points can be partitioned eight ways into two groups. Figure 2(a) shows that the linear boundary can shatter three points because it can identify all eight partitions. However, the linear boundary cannot shatter four points, because in panel (b) it cannot achieve the partition $\{ac \mid bd\}$. This partition can be achieved using the quadratic boundary, $\mathcal{M}2$, which has capacity 5. ($\mathcal{M}2$ can shatter four and five points, but not six.) Put another way, no matter how well we estimate the parameters of the linear machine $\mathcal{M}1$, as a decision boundary it lacks the (structural) capacity to solve problem (b).

Capacity measured in this way would be, at most, an interesting addendum to the area of structural analysis in modeling if the story ended here. However, Vapnik and colleagues were able to derive a unifying relationship between *capacity* (h), *sample size* (l), and *empirical risk* (R_{emp}) (Vapnik and Chervonenkis 1971). To define empirical risk, consider the following characterization of the discrete prediction problem. Suppose we are given l observations drawn independently from an unknown distribution $P(\mathbf{x}, y)$. (P symbolizes the cumulative probability function; vectors and matrices appear in boldface type.) Each observation consists of an n-dimensional vector $\mathbf{x}_i \in R^n$, $i = 1, \ldots, l$ and an associated output y_i. For binary choice situations, y_i is either 0 or 1. For multiclass problems, y_i is an element of an index set.

Suppose that a real-valued function underlies the data-generating process; g: $\mathbf{x}_i \to y_i$. The researcher wants to "build" a machine to approximate this function and use it to predict outcomes. The machine is defined by a family of possible functions $f(\mathbf{x}, \boldsymbol{\alpha})$, where $\boldsymbol{\alpha}$ is a set of adjustable (hyper)parameters of the family. For a fixed $\boldsymbol{\alpha}$, the machine is deterministic. It will generate the same output $f(\mathbf{x}, \boldsymbol{\alpha})$ if the same $\mathbf{x}$ is input. However, $f(\mathbf{x}, \boldsymbol{\alpha})$ may or may not equal the true value of y. Under these conditions, the actual (expected) empirical risk is the expectation of prediction error for a perfectly trained machine; i.e., $R(\boldsymbol{\alpha}) = \int \frac{1}{2}|y - f(\mathbf{x}, \boldsymbol{\alpha})|\, dP(\mathbf{x}, y)$. However, because we do not know the function P, $R(\boldsymbol{\alpha})$ can only be approximated. This approximation produces *empirical risk*, defined as the observed mean error rate on a training set for a fixed number of observations

$$R_{\text{emp}}(\hat{\boldsymbol{\alpha}}) = \frac{1}{2l}\sum_{i=1}^{l} |y_i - \hat{f}(\mathbf{x}_i, \hat{\boldsymbol{\alpha}})|.$$

Note that R_{emp} is based on an approximation to the true f. For a fixed sample size, there are infinitely many such realizations. Thus, the approximation $\hat{f}$ is a realization of the random variable f. The realization varies as a function of the observations in the training set. Note further that $f(\mathbf{x}, \boldsymbol{\alpha})$ may not belong to the same class of functions as the actual data-generating process g. In the best case, $f \subseteq g$, but it may not be so, because the class of mathematical functions is not ordered. For example, a quadratic can be used to approximate $y = \beta_0(1 - e^{-\beta_1 x})$, but neither function is a special case of the other.

Vapnik (1998) refers to the use of $R_{\text{emp}}(\hat{\boldsymbol{\alpha}})$ to approximate the function $R(\boldsymbol{\alpha})$ as the *empirical risk minimization* (ERM) principle. R_{emp} is computed without reference to a probability distribution. It is a fixed number for a particular choice of $\boldsymbol{\alpha}$ and a given set of training data $\{\mathbf{x}_i, y_i;\ i = 1, \ldots, l\}$. Various models and associated computational methods have been proposed to approximate $R(\boldsymbol{\alpha})$ based on $R_{\text{emp}}(\hat{\boldsymbol{\alpha}})$. However, minimizing empirical risk (*in-sample fit* conditional on f) can be, but need not necessarily be, equivalent to minimizing expected total risk. The difference $R(\boldsymbol{\alpha}) - R_{\text{emp}}(\hat{\boldsymbol{\alpha}})$ is bounded and is the subject of the second contributor to overall risk, *structural risk*, caused by the choice of f in relation to the true g. At the heart of the SVM approach is the goal of optimally trading-off empirical and structural risk. Unlike parametric approaches that address this task sequentially—i.e., fix structure first, then minimize empirical risk—the SVM solves the problem simultaneously.

The unifying relationship between capacity (h), sample size (l), and empirical risk was originally developed by Vapnik and Chervonenkis (1971) for losses taking either 1 or 0. Using $0 \le \eta \le 1$, they show that the following bounds hold with probability $1 - \eta$.

$$R(\boldsymbol{\alpha}) \le R_{\text{emp}}(\hat{\boldsymbol{\alpha}}) + \sqrt{\left(\frac{h[\log(2l/h) + 1] - \log(\eta/4)}{l}\right)}. \quad (1)$$

The right-hand side (rhs) of (1) is the *risk bound* and the term under the radical is called *VC confidence*. *VC*

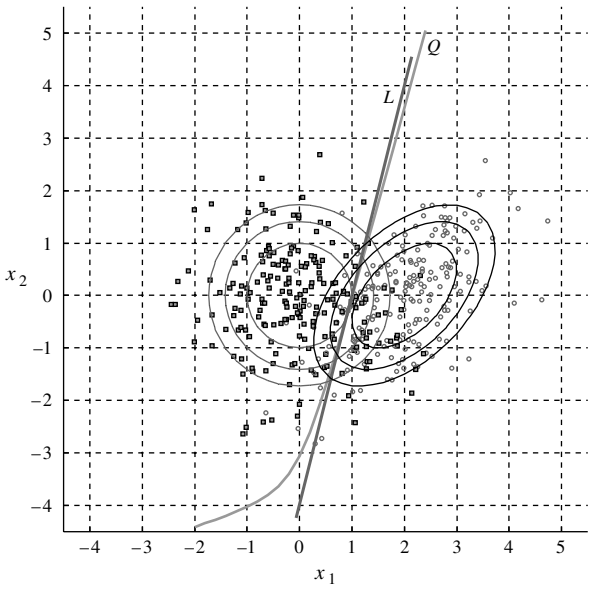
confidence, for a fixed sample size l, monotonically increases in capacity (h). For a fixed capacity, VC confidence shrinks as $l \to \infty$. As this value shrinks, the rhs of (1) more tightly bounds expected risk; i.e., the approximation $R(\boldsymbol{\alpha}) \approx R_{\text{emp}}(\hat{\boldsymbol{\alpha}})$ becomes better. Thus, with large samples, fixing structure and focusing on empirical risk minimization can give good results. More precisely, for a given sample, if the ratio l/h is "large" (sample size is much larger than capacity), expected risk is dominated by empirical risk. In this case, minimizing empirical risk is roughly equivalent to minimizing expected risk. This is the path followed when using classical criteria such as the likelihood ratio statistic, AIC, or BIC. However, each of these criteria depends on the asymptotic properties of the probability distribution presumed to generate the data, whereas (1) does not. Furthermore, if the ratio l/h is "small," then empirical risk and expected (actual) risk are quite different. What one means by "small" is, therefore, of paramount importance for assessing the quality of a model.

The trade-offs in (1) serve as defining logic for the support vector machine. We illustrate using models from the classes, $\mathcal{M}1$ and $\mathcal{M}2$, mentioned earlier, and their application to a two-group prediction problem. Let $\mathbf{x} = (x_1\ x_2)^T$ be drawn with probability 0.5 from either $N_1(\mu_1, \Sigma_1)$ or $N_2(\mu_2, \Sigma_2)$, where $\mu_1 = (0\ 0)^T$, $\mu_2 = (2\ 0)^T$, $\Sigma_1 = I_{2\times 2}$, and $\Sigma_2 = \begin{bmatrix} 1 & 0.5 \\ 0.5 & 1 \end{bmatrix}$. Applying Bayes' theorem to classify $\mathbf{x}$ based on the maximum posterior probability of group membership leads to a quadratic boundary (Q) between the two groups in the x_1, x_2 attribute space, as shown in Figure 3. Pooling the variance-covariance (vcv) matrices leads to the structurally incorrect, linear (L) boundary. Q is the Bayesian (optimal) rule, while L is computed using Bayes' theorem, but based on less information.

These rules are virtually identical in the areas where probability mass is more highly concentrated. Thus, intuition suggests that their empirical counterparts should have very similar prediction hit-rates. More important, nothing from Figure 3 suggests that the structurally misspecified model L should outperform the structurally correct model Q. The relation (1) implies, however, that with small sample sizes, empirical risk and true risk will differ significantly. This means that with smaller sample sizes, empirical realizations of the structurally misspecified model will outperform those from the structurally correct model. We offer additional explanation of why this should be so after viewing results.

Table 1 shows results when estimation sample size (l) is varied over the values $\{10, 50, 100, 200, 400\}$. Within each sample size, 100 trials were run. In each trial, both the linear and quadratic discriminant models were estimated using the same l observations, balanced to reflect equal priors. Each instantiated

Figure 3 Sample Size, Capacity, and Generalization



model was then used to predict the group membership of $N = 1{,}000$ fresh draws from the population, again balanced 500/500. For each trial, Table 1 (column (a)) shows the frequency of times each model wins; column (b) shows each model's mean error rate (*misses* $= 1 -$ *hit rate*); and columns (c) and (d) show the standard deviation and range, respectively, of these error rates over the 100 trials for the sample size in question. Column (a) shows that for smaller estimation samples, the linear model resoundingly outpredicts the structurally correct model, despite the apparent similarity in their decision boundaries. When $l = 10$, Q wins in only 15 cases. Even though Q generates the data, its prediction error rate is 26% higher on average than the error rate of the linear machine. When $l = 200$, L still wins by a 3:2 margin, though hit-rates are nearly equal. This pattern

Table 1 Sample Size, Capacity, and Generalization

Est. sample	Model type	(a) Freq (Wins)[a]	(b) Mean	(c) Std	(d) Range min	(d) Range max
$l = 10$	*L*	85	0.1894	0.0475	0.1360	0.4120
	Q	15	0.2388	0.0769	0.1300	0.5050
$l = 50$	*L*	69	0.1575	0.0123	0.1295	0.1970
	Q	31	0.1611	0.0121	0.1355	0.2030
$l = 100$	*L*	66	0.1524	0.0080	0.1340	0.1710
	Q	34	0.1547	0.0093	0.1240	0.1780
$l = 200$	*L*	60.5	0.1509	0.0096	0.1330	0.1800
	Q	39.5	0.1522	0.0088	0.1310	0.1745
$l = 400$	*L*	47	0.1501	0.0088	0.1240	0.1725
	Q	53	0.1501	0.0088	0.1245	0.1720

[a]Ties were split evenly between *L* and *Q*. Ties occurred when $l = 100$ (4 ties), $l = 200$ (7), and $l = 400$ (6).

persists up to $l = 400$, where Q finally wins in slightly more than half the trials. Even with this "large" sample size—relative to the number of parameters estimated—the linear machine's mean empirical error rate is virtually identical to that of Q.

3.3. Positivist Habits, Structural Diagnosticity, and Prediction

This simple example shows that a marketing scientist who focuses on uncovering the "real" latent structure generating data will be misled. In all cases above, it must be true that Q fits better *in sample* than L, because $L \subset Q$. However, neither this fact nor the fact that L outpredicts Q can be trusted to find the actual model generating the data. As engineers of the data-generating process, we have the benefit of knowing the real process. The marketing scientist trying to reverse engineer the data to judge whether L is structurally right or wrong would be misled (especially for smaller sample sizes) by L's superior in-sample fit (adjusted for df), its superior out-of-sample prediction rate, and its relative simplicity, which satisfies the scientific standard for parsimony.

3.4. Decision Boundary Bias

In discrete-class prediction, decision boundary bias interacts with the ERM principle to further complicate the scientist's ability to recognize true structure. Recall that empirical risk is calculated using $\hat{f}(\mathbf{x}_i, \hat{\boldsymbol{\alpha}})$, an approximation of the function used by a perfectly trained machine. This approximation is one realization of the random variable, f. Figure 3 shows the true boundaries L and Q for the engineered process. However, in each sample summarized in Table 1, the estimated boundaries $\hat{L}$ and $\widehat{Q}$ differ from their population counterparts. (Q is the real process, thus using the analogy $g \equiv Q$ and $f \equiv L$, we see that $\hat{f}$ is a double approximation to g. In the present case, $f \subset g$, but this fact is specific to our example.)

The discrete predictions generating the hit-rates for Table 1 are obtained using the classic two-phase process, function estimation followed by discrete classification. It is well known that the function estimation phase improves monotonically with increasing sample size. However, using a clever decomposition of variance, Friedman (1997) shows that, surprisingly, this is *not* the case for improvements in the discrete classification phase. In particular, given a training sample (of size l), the error rate, *averaged over all future predictions at a given point* $\mathbf{x}$, depends on whether the classification rule is Bayes' rule (the optimal rule) or not. If it is, then the error rate is the irreducible error associated with Bayes' rule (i.e., class overlap in 2-d).[7] If not, there is an added component of variance that depends on the variance of the estimated decision boundary; e.g., $V(\hat{f}) = E\lfloor \hat{f} - E(\hat{f})\rfloor^2$. The classification error rate and the function estimation error rate are influenced completely differently by this variance. Decreasing $V(\hat{f})$ decreases classification error when decision boundary bias is negative, but *increases* it when bias is positive. Thus, focusing on improved function accuracy; i.e., minimizing $V(\hat{f})$ can actually make the discrete-choice prediction phase less accurate. In such cases, a simpler (but incorrect) model will outpredict the structurally correct model if it is biased on the "correct side" of $\mathbf{x}$ more often than not. This is precisely the case for L in our example. Table 1 shows that the paradox persists to the point where the sample size is large enough to reduce empirical risk and boundary bias (of $\widehat{Q}$ in this case) below that of $\hat{L}$. This usually requires very large samples, where large is indexed by the ratio of sample size to capacity, i.e., the VC-dimension of the model.

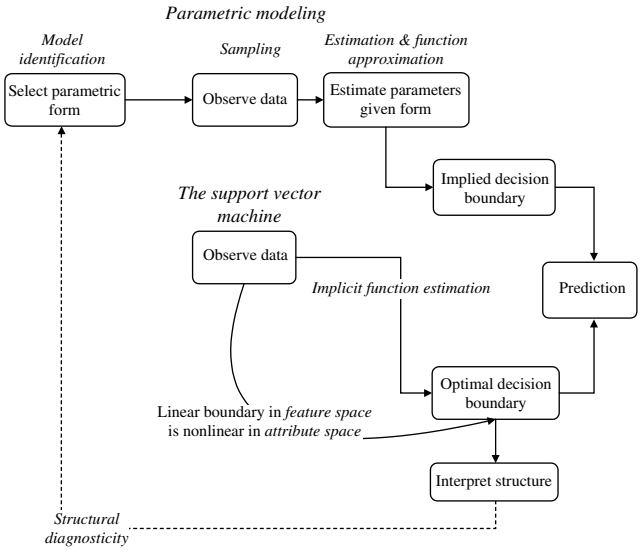
3.5. Dimensionality, Parametric Estimation, and Prediction

As Ben-Akiva et al. (1997, p. 279) noted, "When the utility function has many covariates, the data points are sparse over the high dimensional space and the estimation becomes unstable . . . . As the number of dimensions increase (sic), an exponential increase in sample size is needed to maintain reliable estimation." This trade-off between model complexity, available data, and empirical risk (Bellman 1961) interacts with decision boundary bias in parametric models as suggested in Figure 4. Ultimately, the selected decision boundary is subject to a three-link chain of instability. These links involve (a) the hypothesized parametric form(s) selected to model various stochastic elements of the process, (b) the relationship between the dimensionality of the problem space and the number of data points (and their sampling properties) when empirically approximating the identified form(s), and (c) properties of the estimation technique (analytic or numeric), i.e., estimation in RUT entails finding the global optimum of the likelihood or simulated likelihood surface.

Figure 4 suggests very generally how the SVM avoids these potential pitfalls. Foremost, the decision boundary is found directly from available data, not indirectly through function approximation. (This derivation is shown in the next section.) In fact, Vapnik's (1979) original work focused on finding the optimal decision boundary for linearly separable outcome classes. The result, known as the *maximum margin optimal classifier*, does not depend on any

[7] Class overlap in 2-d is irreducible error in (x_1, x_2), but not necessarily globally irreducible error because in higher dimensions the overlap may disappear. This, of course, is a restatement of the deeper question of whether or not nature contains inherent randomness.

Figure 4 Dimensionality and Prediction



parametric assumptions about the process generating the data, but requires only that samples be drawn independently from this process. Boser et al. (1992) extended this result to nonlinear problems using a kernel function to map attribute space into a higher-dimensional feature space. In feature space, a linear boundary is still sought. However, classes that are linearly separable in feature space may be nested or otherwise highly intertwined in attribute space. The resulting boundary, when translated back to attribute space may, therefore, be nonlinear. Cortes and Vapnik (1995) then extended the solution to soft-margin classifiers to deal with data that are not perfectly separable, even in feature space.

The parameters of the optimal linear classifier in feature space are solutions to a constrained quadratic programming problem. As such, these estimates are provably globally optimal and are not conditioned on a particular parametric form or chain of forms.[8] They are mathematical in nature (like OLS or minimum chi-square), not statistical in nature (like maximum likelihood). We return to this point because it clearly forces additional reflection about the—not necessarily mutually exclusive—goals of prediction and structural diagnostics. The resulting decision boundary is not subject to decision boundary bias because it is not the discretized version of a function approximation exercise, but rather a direct attack on the optimal boundary problem. As Figure 4 implies, structural interpretations are possible with the support vector machine, but they are made post hoc once the model's predictive capacity has been fully utilized. Figure 4 links the SVM and classic parametric paradigms to emphasize their complementary rather than competitive nature.

[8] For example, Vapnik (1995) documents the failure of analytic maximum likelihood to solve a mixture of normal distributions. Train and Sándor (2002), while exploring various draw techniques for mixed logit models (e.g., pseudorandom draws, Halton sequences, orthogonal arrays, and Latin hypercube draws) identify cases where the maximum of the simulated likelihood function is never found.

4. Implementing an SVM

The support vector machine combines concepts from abstract Hilbert spaces with modern optimization techniques. We concentrate in this section on the main steps required to implement an SVM rather than on technical detail. Technical detail is provided in Appendix A for the optimization components of the SVM, including extensions to soft-margin classifiers and multiclass problems. Appendix B illustrates how kernel transformations work. Kernel transformations are a fundamental ingredient of a support vector machine because they allow a linear machine to solve a nonlinear problem.

4.1. Maximum Margin Classifiers

Vapnik (1979) approached the discrete classification problem from the perspectives of function capacity and shattering. He reasoned that although many hyperplanes can fit between linearly separable groups, the optimal separating hyperplane should lie midway between the convex hulls of the two groups and be orthogonal to the shortest line connecting these hulls. The solution has both a primal and a dual form, as shown in (2). (The observations $\mathbf{x}$ and $\mathbf{x}_i$ are vectors in $\Re^n$.) Each form can be solved as a quadratic programming problem (see Appendix A).

$$f(\mathbf{x}) = \mathbf{w}^* \cdot \mathbf{x} + w_0^* = \sum_{i=1}^{n} \alpha_i^* y_i (\mathbf{x} \cdot \mathbf{x}_i) + w_0^*. \qquad (2)$$

Note that the dual representation contains the inner product $(\mathbf{x} \cdot \mathbf{x}_i)$. The dual's dependence on these inner products is key because it facilitates solutions to nonlinear variations of the problem. In particular, Vapnik (1995) noted that a problem in attribute space could be transformed to a problem in a higher-dimensional feature space without altering the solution form. This technique—mapping a problem to a new domain, solving it there (where the solution technique is easier), then mapping the solution back to the original domain—is frequently used in mathematics. A simple example is solving multiplication problems by addition using the mutually inverse log/exp transforms. A second example is solving differential equations using Laplace (and inverse Laplace) transforms to migrate between the time and frequency domains.

4.2. Kernel Transformations

The SVM machine uses kernel transformations that take the general form $\phi(\mathbf{x}\cdot\mathbf{z}) = \langle\phi(\mathbf{x})\cdot\phi(\mathbf{z})\rangle$. That is, the transform of the inner product of the n-vectors $\mathbf{x}$ and $\mathbf{z}$ is the vector inner product of their transformed images. This means that the form of the dual problem is the same whether it is solved in attribute space using $(\mathbf{x}\cdot\mathbf{z})$ or in feature space using $\phi(\mathbf{x})\cdot\phi(\mathbf{z})$. Note further that the inherent size of the problem is the same in both spaces. The degrees of freedom used in estimating the decision boundary depends only on the number of data points, l, not on the product $l\cdot n$ as with maximum likelihood estimators. In fact the boundary depends on a subset of l, the vectors that form the inner edges of the convex hulls of the two classes: the problem's support vectors. Hence, the curse of dimensionality is neutralized.

The optimal boundary is calculated as the solution to the dual quadratic programs (QPs) problem in feature space.

$$\text{Max} \sum_{i=1}^{n}\alpha_i - \tfrac{1}{2}\sum_{i,\,j=1}^{n}\alpha_i\alpha_j\, y_i y_j \phi(\mathbf{x}_i\cdot\mathbf{x}_j) \tag{3a}$$

$$\text{s.t.} \sum_{i=1}^{n} y_i\alpha_i = 0 \quad \alpha_i \geq 0, \quad i = 1, l. \tag{3b}$$

We implement the solution using Platt's (1999) highly efficient *Sequential Minimization Optimization* (SMO) algorithm. SMO breaks the problem into a series of smallest possible QPs that are solved analytically. The algorithm uses less memory and is easier to implement than "standard" algorithms, and can be more than 1,000 times faster (Platt 1999).

Two types of kernels are most often used in practice, the Gaussian (or radial basis) kernel $\phi(\mathbf{x},\mathbf{z}) = \exp(-\|\mathbf{x}-\mathbf{z}\|^2/\sigma^2)$ with hyperparameter σ and the polynomial kernel $\phi(\mathbf{x},\mathbf{z}) = (\mathbf{x}\cdot\mathbf{z}+1)^d$ with hyperparameter d. If the data have dimensionality n, then the feature space mapped by a polynomial kernel has dimension $\binom{n+d}{d}$. The dimensionality of a feature space mapped by a Gaussian kernel can be infinite (Burges 1998).

Appendix B contains numerical details of a complete example using a polynomial kernel. We selected a problem known to be unsolvable with standard GLM methods and with multinomial logit. Neither can the problem be solved by the artificial intelligence technique known as perceptrons (Minsky and Papert 1969). It can be solved using an artificial neural net with hidden layers, but the solution is very slow to converge (Langley and Burgess 2000). The SVM solution using a second-degree polynomial kernel is intuitive and numerically efficient, solving virtually instantly even with more than the two predictors illustrated in Appendix B.

The support vector machine implemented by a kernel family imposes a structure on the set of functions that comprise the learning machine. Currently, there is no metatheory regarding the choice of kernel family. Normally, this choice is based on domain knowledge or researcher preference supplemented by numerical results (Boser et al. 1992). If both standard kernels (Gaussian and polynomial) prove ineffective, more elaborate kernels can be constructed from "building blocks" according to mathematical principles from the theory of integral operators (Cristianini and Shawe-Taylor 2000). Empirical evidence suggests that the choice of kernel family has little influence on the generalization performance of the machine (Vapnik 1995, Schölkopf et al. 1998). In other words, the best kernel in the polynomial family and the best kernel in the Gaussian family typically perform equally well in a given problem context. However, training times to achieve this performance may differ by several orders of magnitude as a function of the selected kernel. The procedure we used to estimate the hyperparameters of the kernels used in the empirical tests reported next is based on well-known resampling techniques (Edgington 1995, Efron and Tibshirani 1993, Good 2005).

5. Empirical Tests of the Support Vector Machine

We turn now to comprehensive tests of the predictive capacity of the support vector machine baselined by direct comparisons to those from multinomial logit.[9] For this purpose, Monte Carlo simulation is appropriate because it allows us to control key aspects of the data-generating process over a wide range of experimental conditions.[10] Ben-Akiva and Lerman (1985) discuss seven variables relevant to the performance of a discrete-choice model: (1) the number of product attributes, (2) the estimation sample size, (3) the magnitude of error in the stochastic component of the random utility model, (4) the number of individual characteristics, (5) the number of choice alternatives, (6) the type of error distribution, and (7) whether or not a correlated error structure is present. We manipulate these seven variables in a $2^5\times 3^2$ factorial design using the levels shown in Table 2. Levels were chosen to cover a broad range of situations encountered in practice, particularly in consumer choice experiments using the logit model.

[9] We use a consumer choice prediction task because of the prominence of discrete-choice modeling in marketing and the widespread understanding of logit as opposed to most of the emerging models mentioned in §2.

[10] Toubia et al. (2004, p. 123) provide a detailed explanation of why Monte Carlo experiments are widely used and appropriate for testing model performance in consumer choice experiments of the type reported here.

Table 2 Independent Variables in Simulation Study

Variable symbol	Variables	Low (L) level	Med (M) level	High (H) level
A	Number of product attributes	2	4	6
B	Estimation sample size	100	400	1,600
C	Error size (stochastic component)	0.5%		10%
D	Number of individual characteristics	1		3
E	Number of choices	2		6
F	Type of error distribution	Normal		Gamma
G	Correlated error structure	No		Yes

Given our focus on predictive accuracy, the dependent variable is the first-choice hit-rate produced by an estimated model in a validation sample (of size 3,000) from the same data-generating process that created the estimation/training sample. A 16-treatment fraction of the full factorial is used.[11] The plan permits clear estimation of all main effects along with six selected two-way interactions. We repeat the experiment 10 times within each treatment.

5.1. Data-Generating Functions

For the deterministic component of utility, we simulate a variety of data-generating functions that mimic known consumer decision structures. We illustrated in §3.1 that for pure noncompensatory structures, such as latitude of acceptance, logit will not perform well. In the tests reported here, we confine attention to cases where logit has no inherent limitations. Thus, we constrain decision rules to be compensatory, but not necessarily linear. Further, we include no effect higher than quadratic. Quadratic effects in utility are common for many product attributes. We include effects in pure form, for an attribute alone, and in moderated form where an effect interacts with an individual characteristic. For example, many consumers exhibit a quadratic response to variations in price. This response is likely to be moderated by an individual's income level. We also include bilinear interactions between two product attributes and allow these to be moderated by individual characteristics.

The systematic components of utility are shown in Table 3. These functions share the following properties. Each component includes two linear terms of product attributes and one linear term of individual characteristic. Each component includes one interaction term between two product attributes. Each includes two interaction terms between a product attribute and an individual characteristic. Each contains a quadratic effect for a product attribute. Finally, each component contains a quadratic by linear interaction between a product attribute (quadratic) and an individual characteristic (linear). This latter type of effect was recovered by Brynjolfsson and Smith (2001) in their study of online consumer decision making. All six deterministic components have the same number of terms.

Table 3 Systematic Components of Utility

	Deterministic component of the utility function
Utility 1	$\mathbf{x}'_{ik} = [x_{ik1}\ x_{ik2}\ x_{ik1}x_{ik2}\ x_{ik1}z_i\ x_{ik2}z_i\ x^2_{ik1}\ x^2_{ik2}z_i\ z_i]$ x_{ik1}, x_{ik2} are Person i Choice k's attribute data; z_i is Person i's single individual characteristic.
Utility 2	$\mathbf{x}'_{ik} = [x_{ik1}\ x_{ik2}\ x_{ik1}x_{ik2}\ x_{ik1}z_{i1}\ x_{ik2}z_{i2}\ x^2_{ik1}\ x^2_{ik2}z_{i2}\ z_{i3}]$ x_{ik1}, x_{ik2} are Person i Choice k's attribute data; z_{i1}, z_{i2}, and z_{i3} are Person i's three individual characteristics.
Utility 3	$\mathbf{x}'_{ik} = [x_{ik1}\ x_{ik2}\ x_{ik1}x_{ik2}\ x_{ik2}z_i\ x_{ik3}z_i\ x^2_{ik3}\ x^2_{ik4}z_i\ z_i]$ x_{ik1}, x_{ik2}, x_{ik3}, and x_{ik4} are Person i Choice k's attribute data; z_i is Person i's single individual characteristic.
Utility 4	$\mathbf{x}'_{ik} = [x_{ik1}\ x_{ik2}\ x_{ik1}x_{ik2}\ x_{ik2}z_{i1}\ x_{ik3}z_{i2}\ x^2_{ik3}\ x^2_{ik4}z_{i2}\ z_{i3}]$ x_{ik1}, x_{ik2}, x_{ik3}, and x_{ik4} are Person i Choice k's attribute data; z_{i1}, z_{i2}, and z_{i3} are Person i's three individual characteristics.
Utility 5	$\mathbf{x}'_{ik} = [x_{ik1}\ x_{ik2}\ x_{ik1}x_{ik2}\ x_{ik3}z_i\ x_{ik4}z_i\ x^2_{ik5}\ x^2_{ik6}z_i\ z_i]$ x_{ik1}, x_{ik2}, x_{ik3}, x_{ik4}, x_{ik5}, and x_{ik6} are Person i Choice k's attribute data; and z_i are Person i's single individual characteristics.
Utility 6	$\mathbf{x}'_{ik} = [x_{ik1}\ x_{ik2}\ x_{ik1}x_{ik2}\ x_{ik3}z_{i1}\ x_{ik4}z_{i2}\ x^2_{ik5}\ x^2_{ik6}z_{i2}\ z_{i3}]$ x_{ik1}, x_{ik2}, x_{ik3}, x_{ik4}, x_{ik5}, and x_{ik6} are Person i Choice k's attribute data; z_{i1}, z_{i2}, and z_{i3} are Person i's three individual characteristics.

These six components of the overall data-generating process get progressively more complex as more product attributes and individual characteristics are incorporated. Because data are represented in an "exploded" form when estimating a logit model, the size of the design matrix increases in number of choices, number of product characteristics, and number of individual characteristics. Thus, even though the number of individuals in the validation sample is held constant in each case, the effective estimation/prediction effort varies by treatment.

We implemented the SVM using a polynomial kernel with hyperparameter training using resampling. For logit modeling, we used SAS's multinomial discrete-choice procedure (MDC). MDC supports a wide variety of structural forms, including simple, conditional, and nested logit. We use logit models with alternative specific constants and alternative specific individual characteristics. Whenever error terms are correlated, we fit a nested logit model and assume that the nested structure is known a priori. These assumptions provide an advantage to the logit class of models, giving maximum flexibility to capture the data-generating process within the confines of what we refer to as "standard practice;" e.g., linear-in-parameters forms as described in §§2.3 and 3.3.

[11] See Hahn and Shapiro (1966; Plan Code 65A, Master Plan 5).

Table 4 Predictive Performance of the Support Vector Machine and Logit Model

	Logit model		Support vector machine			Logit model		Support vector machine	
	Training	Testing	Training	Testing		Training	Testing	Training	Testing
1	0.7910 (0.0292)	0.7874 (0.0183)	0.8070 (0.0523)	0.8009 (0.0132)	9	0.7410 (0.0360)	0.6694 (0.0203)	0.9960 (0.0052)	0.9688 (0.0176)
2	0.6178 (0.0150)	0.6162 (0.0086)	0.7087 (0.046)	0.6929 (0.0102)	10	0.8994 (0.0036)	0.8872 (0.0048)	0.9501 (0.0124)	0.9317 (0.0051)
3	0.8166 (0.0034)	0.8145 (0.0023)	0.8292 (0.0071)	0.8359 (0.0025)	11	0.7136 (0.0033)	0.7100 (0.0021)	0.9066 (0.0024)	0.9045 (0.0017)
4	0.6260 (0.0132)	0.5934 (0.0058)	0.6939 (0.0233)	0.6664 (0.0060)	12	0.8690 (0.0095)	0.8490 (0.0047)	0.8834 (0.0174)	0.8797 (0.0051)
5	0.8890 (0.0256)	0.8520 (0.0101)	0.9230 (0.0337)	0.8598 (0.0054)	13	0.3200 (0.0231)	0.3038 (0.0139)	0.8450 (0.0438)	0.8193 (0.0114)
6	0.7715 (0.0068)	0.7806 (0.0055)	0.8234 (0.0180)	0.8521 (0.0057)	14	0.8742 (0.0087)	0.8558 (0.0042)	0.8949 (0.0162)	0.8796 (0.0046)
7	0.9134 (0.0020)	0.9088 (0.0017)	0.9552 (0.0048)	0.9491 (0.0025)	15	0.3347 (0.0081)	0.3316 (0.0056)	0.8917 (0.0074)	0.8930 (0.0023)
8	0.8265 (0.0070)	0.8144 (0.0044)	0.8942 (0.0213)	0.8897 (0.0077)	16	0.9103 (0.0041)	0.9068 (0.0038)	0.9572 (0.0114)	0.9345 (0.0055)

5.2. Results

Table 4 shows the mean first-choice hit-rates for both estimation (training) and validation (prediction) in each of the 16 treatment conditions. Each mean is the average of the 10 values in its corresponding cell. The standard deviation among these 10 values is shown in parentheses. Table 4 indicates that in every treatment condition the SVM outpredicts the corresponding logit model. This is true not only on average, but for every one of the 160 total cases run across all cells. The overall mean prediction rate of the logit is 72.7% while this hit-rate is 85.9% for the support vector machine. Computing the percent dominance within treatment, then averaging these 16 values, the SVM predicts 29.9% better on average than the logit, ranging from a virtual tie in cell 5 to 169.6% and 169.3% better in cells 13 and 15. (For reference, Cell 13 has four product attributes, estimation sample size of 100, high error, one individual characteristic, six choice alternatives, a nonnormal error density, and no correlated error. Cell 15 has six choice alternatives, uses the large sample size, and has low, normally distributed correlated error.)

5.3. Subexperiments on Main Effects

To develop a better understanding of the direct impact of certain factors, we conducted three subexperiments. We focused on three of the four largest sources of variation from the main experiment: (a) the number of choice alternatives, (b) the number of individual characteristics, and (c) the estimation sample size.[12] (Table 5 shows proportion of variance explained by each factor and estimable interactions in the main experiment. Results are significant at $p < 0.05$ or better unless otherwise indicated.)

In the subexperiments, nonmanipulated factors were held fixed while the manipulated factor was varied. Manipulating just one factor removes confounding in the raw marginal means from the main experiment.[13] In the subexperiment for number of choice alternatives, we expanded the number of levels to five (2–6 alternatives) to get a complete picture of the marginal effect of this important variable.

Figure 5 shows results. Because each subexperiment was replicated 10 times, every difference between hit-rate proportions is statistically significant at a minimum of $p < 0.01$, and usually at much smaller p-values. Panel (a) indicates that as the number of choice alternatives increases, the prediction hit-rate of each model falls. This is expected because the "first-choice" prediction task increases in difficulty with more alternatives. However, the decline is much steeper for the MNL, dropping from 85.6% at two alternatives to 74.9% with six, a 12.4% decline. The falloff for the SVM, from 88.0% to 84.2%, is a

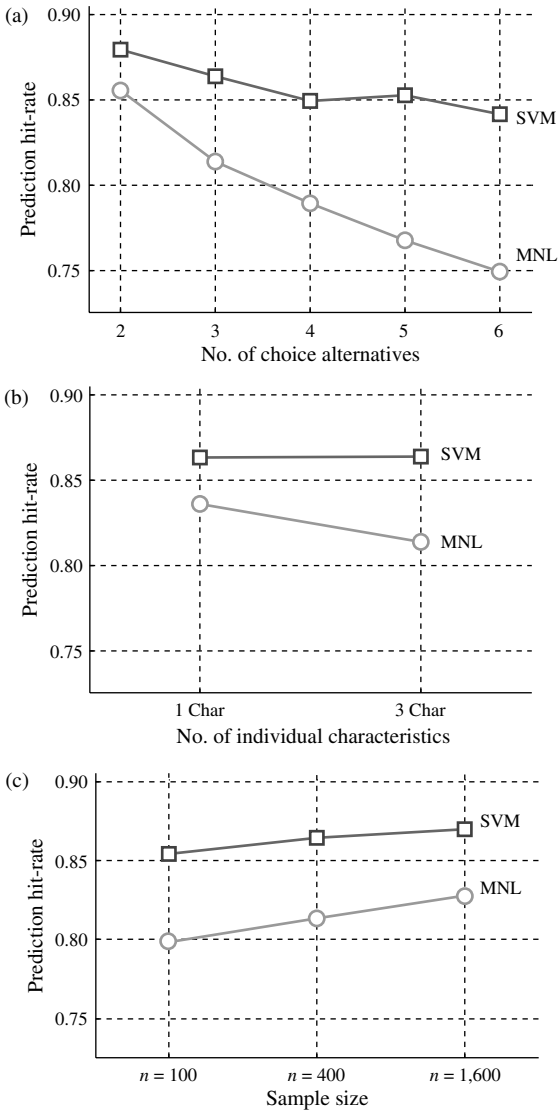
[12] All three subexperiments used the conditions: four attributes, normal, uncorrelated error at 10%. These levels correspond to those in cell 14 of the main experiment, where the SVM and MNL perform nearly the same; i.e., cell 14 is a reasonable starting point for comparisons of marginal effects. When not varied, the estimation sample size was $l = 400$, the number of choice alternatives was three, and the number of individual characteristics was three. Note that manipulating the *number of attributes* factor would alter the "common size" of the utility functions shown in Table 4. Hence, this factor was not included in the subexperiments.

[13] Although the fractional design in the main experiment is orthogonal, it is unbalanced due to mixing factors with two and three levels. Thus, raw marginal means do not correspond to weighted contrasts.

Table 5 ANOVA Results (Variance Explained Normalized to 100%)

Symbol	Manipulated factor	SVMs	Logit models	SVMs-logit
	Main effects			
A	Number of product attributes	61.12	2.54	7.97
B	Sample size	6.60	10.63	14.94
C	Error size	11.44	0.56	0.76
D	Individual characteristics	0.24	3.45	3.15
E	Number of choices	6.36	52.84	48.05
F	Type of error distribution	0.23	0.25	0.27
G	Correlated error structure	1.65	0.00	0.10
Subtotal		87.64	70.26	75.24
	Two-way interactions			
A∗B	No. of attributes × sample size	9.61	9.74	4.40
A∗C	No. of attributes × error size	0.41	0.08	0.09
A∗E	No. of attributes × number of choices	-ns-	-ns-	-ns-
B∗C	Sample size × error size	-ns-	-ns-	-ns-
B∗E	Sample size × number of choices	-ns-	-ns-	-ns-
C∗E	Error size × number of choices	0.56	19.66	19.47
Subtotal		10.58	29.48	23.97
Error		1.78	0.25	0.79
Total		100	100	100

Figure 5 Three Subexperiments



4.3% decline over the same range. Panel (b) shows that adding individual characteristics—and therefore, complicating the structural form of the relationship between predictors and target—has no effect on the performance of the SVM. The SVM's hit-rate actually improves slightly in the case shown. The MNL, because it performs density estimation prior to the prediction step, is more sensitive to the ratio of parameters to sample size. Since sample size is fixed (at $l = 400$ in this experiment), cases with three individual characteristics require additional parameters be estimated. This use of degrees of freedom negatively impacts prediction ability. Finally, Panel (c) shows that all else being equal, increased sample size helps both models as expected. The MNL is more sensitive to the change, improving 3.62% over the range shown versus 1.84% for the SVM. Of course, the SVM starts from a significantly higher base prediction rate, and hence has less room to improve.

These subexperiments allow us to anticipate the marginal effect on prediction hit-rate in tasks with varying numbers of alternatives, individual characteristics, and estimation sample sizes. However, from the main experiment, we found significant interactions between certain factors. Because the fractional experimental design limited our search for interactions, we suspect higher-order interactions are also at work, particularly given the very low prediction rates for the MNL in cells 13 and 15.

5.4. Discussion

Results suggest that the SVM has considerable promise for accurately predicting consumer choice in the "pure prediction" environments found in automated modeling, mass-produced models, intelligent agents, and data mining. In our main experiment, a single SVM significantly outpredicts the best model from a set of appropriate MNL models even though perfect a priori knowledge is used to correctly select the nesting structure in cases with correlated error. In practice, analyst insight is desirable in controlled experimental settings or with surveyed field data, but in areas where automated modeling is useful, the "one size fits all" aspect of the SVM is a definite advantage.

Results from our subexperiments suggest that the prediction hit-rates of MNL are more severely affected by increases in the choice set size and the number of individual characteristics than are those from the SVM. Although further experimentation is warranted, the addition of more individual characteristics seems not to negatively impact the predictive accuracy of the SVM and may even increase it. Increased

accuracy would follow from the fact that individual differences mediate the main effects of choice dimensions. This effect is likely to be much stronger in real choice tasks when covariates are judiciously selected. In other words, with the SVM, more covariates will drive predictive accuracy up, not down.

6. Limitations of the Present Research

6.1. Weaknesses of the Support Vector Machine

Although the support vector machine shows promise, there are obstacles to overcome before the approach gains acceptance in marketing. The SVM's novel inferential philosophy requires additional theoretical development before it can be routinely used in practical situations. Because there are no probability density assumptions made, the SVM does not yield probability estimates for hypothesis testing (classical view) or predictive/posterior bounds (Bayesian view). Efforts to provide such estimates are available (Platt 2000, Vapnik and Chapelle 2000), but a support vector machine does not generate them naturally. Lack of easy-to-use computer software will also impede the SVM's acceptance in marketing, although better software should be forthcoming in the next few years. More fundamentally, there is no complete, working metatheory to assist with the selection of kernel transformation for an SVM. Depending on the choice of kernel family, parameter estimation can be time consuming.

Although our focus is on prediction, not structural diagnostics, having both is a plus for any model. In the area of interpretability, the SVM faces a catch-22. Its most significant advantages are in nonlinear environments, precisely the environments in which estimated parameters cannot be interpreted directly. True, in these situations numerical analysis can be performed to yield response coefficients for the SVM as outlined in Appendix C. Nevertheless, implementing these analyses requires custom programming.

6.2. Weaknesses of the Experimental Comparisons

Although we took considerable care to be thorough and fair in conducting the experiments reported here, our methods can be criticized on several fronts. The main experiment could be enriched by including more levels on certain factors. For example, using six product attributes and three individual characteristics may be too restrictive in some cases. Countering this, Figure 5 suggests that the SVM's predictive accuracy is not sensitive to the number of individual characteristics. Furthermore, when predictive tasks involve seven-plus attributes, in the kinds of "ad hoc" data collection environments we envision, data preprocessing can filter out redundant information prior to the SVM modeling step (Guyon and Elisseeff 2003). Findings suggest that, comparatively, the SVM would gain rather than lose ground in more complex cases.

Researchers specializing in discrete-choice modeling may be disappointed that more sophisticated RUT models were not used in this research. However, our goal was not to pit MNL against SVM in any direct way, but rather to use reasonably sophisticated MNL models to create a workable baseline to provide perspective on the predictive accuracy of the support vector machine. The MNL models performed well in this research, and using more sophisticated versions could improve MNL predictive hit-rates. We offer our data to specialists developing and testing more flexible models. However, these models require custom programming and considerable human intervention in the identification and fitting stages, steps that defeat the purpose of predictive modeling in the environments on which we focus.

7. Directions for Future Research

Our directions for future research stress two areas where SVM and random utility theory (RUT) models are complementary: structural gap identification and simultaneously improving prediction and diagnosticity when nonlinear information integration rules are in play.

7.1. The SVM as a Structural Gap Identifier

Given a set of predictor variables $\mathbf{x}$, the conditional density $y \mid \mathbf{x}$ may contain irreducible uncertainty about the target variable y. However, when an SVM yields out-of-sample prediction rates that significantly exceed those from a structurally rich RUT model (with both methods using precisely the same input), this is a clear signal that the set $\mathbf{x}$ contains additional predictive information in higher-order effects. Under these circumstances, the support vector machine complements standard modeling in two useful ways. First, it puts the analyst on guard when interpreting estimated coefficients as measures of marginal response. Second, it provides motivation to add additional terms to the systematic component of utility (in RUT models) or, more generally, to the functional relationship between target and predictors in other types of models. For example, if a polynomial kernel is used in the SVM, the degree of the polynomial serves as an upper limit to the nested set of functions that need to be searched to better specify the model. Although the perfect structure is unlikely to be found, adding additional effects should achieve much higher prediction rates and structural accuracy than the linear model under consideration.

7.2. Predictive Power and Structural Diagnostics: Convergent Research for Noncompensatory Information Integration Rules

Because the SVM is predictively robust even with very small samples, future research using the SVM

may blend emerging directions in choice-based conjoint and aggregate-level, discrete-choice modeling. (Technical Appendix B provides an overview of choice-based conjoint modeling and its relation to the present work.) For example, Gilbride and Allenby (2004) have recently published models from the RUT class that stress structural accuracy by identifying and modeling noncompensatory information processing. Meanwhile, Bradlow (2003) has wished for predictively accurate conjoint models for choices from noncompensatory information integration rules. These and other modelers from the "choice-based conjoint" camp rightfully want to be able to estimate models at the individual level that are both structurally accurate and highly predictive. This goal is being actively pursued by Evgeniou et al. (2005), who use an SVM-style kernel transformation for analyzing data from choice-based conjoint experiments. This is a promising blend of research streams. The idea may be extended further. The goals would be, first, to isolate the attributes involved in noncompensatory rules at the individual respondent level (because these may differ by person) and, second, to associate explicit patterns of SVM parameter estimates with specific noncompensatory processing strategies—latitude of acceptance, conjunctive, disjunctive, XOR, etc.—to identify which consumers use these strategies, and with what attributes. Results from such models would yield highly useful insights to marketing strategists in the areas of new product design and market segmentation.

8. Concluding Comments

In this paper, we argue that highly predictive models will play an increasingly important role in 21st century marketing applications, particularly in areas such as automated modeling, mass-produced models, intelligent software agents, and data mining. Politz and Deming (1953) argued forcefully that predictive accuracy is the standard by which model quality should be measured. More than 50 years later the argument still resonates with marketing scientists (Allenby et al. 2002). The support vector machine performs well on predictive tasks where the relationship between predictors and target is complex. Although its modeling philosophy is nonstandard, the SVM and related kernel methods may provide not only accurate "pure prediction," but a unique link between structural diagnostics and predictive accuracy in a wide variety of marketing applications.

Acknowledgments

The computer code and data used in the simulation studies are available upon request. Special thanks to Sharon McFarland for editorial comments on earlier drafts of this paper. The authors gratefully acknowledge the insightful comments of the reviewers and the area editor. The authors are listed in alphabetical order. Contributions were equal and synergistic.

Appendix A. Implementing a Support Vector Machine

This appendix reviews the fundamental derivations required to implement a support vector machine. Results fall into three primary areas, the *optimal maximum margin classifier* (Vapnik 1979), extensions to *nonlinear decision functions using kernel transformations* (Boser et al. 1992), and extensions to *soft-margin classifiers* (Cortes and Vapnik 1995).

The Optimal Margin Classifier

An *optimal margin classifier* finds a particular linear boundary between two perfectly separable classes in an *attribute space* of raw data. The idea is similar in spirit to OLS because the problem is stated and solved as a nonparametric optimization problem, not as a parametric (maximum-likelihood) problem. Suppose we have data sampled from an unknown distribution $P(\mathbf{x}, y)$ where y takes on two values. (Multivalued extensions are discussed subsequently.) A linear decision function that completely separates the observations can be expressed as the inner product between a weight vector $\mathbf{w}$ and an input vector $\mathbf{x}$, plus a constant, w_0.

$$f(\mathbf{x}) = \mathbf{w} \cdot \mathbf{x} + w_0. \tag{A1}$$

In the two-class case, the sign of $f(\mathbf{x})$ determines the membership class of the point $\mathbf{x}$. Thus, for points $\mathbf{x}_i$, $i = 1, \ldots, l$, the problem is to find $(\mathbf{w}, w_0)$ such that

$$(\mathbf{w} \cdot \mathbf{x}_i) + w_0 > 0 \quad \text{if } y_i = 1, \tag{A2a}$$

$$(\mathbf{w} \cdot \mathbf{x}_i) + w_0 < 0 \quad \text{if } y_i = -1. \tag{A2b}$$

These inequalities can be expressed compactly as A3.

$$y_i[(\mathbf{w} \cdot \mathbf{x}_i) + w_0] > 0 \quad \text{for } i = 1, \ldots, l. \tag{A3}$$

The formulation (A3) leads to a direct solution of the classification problem without attempting to estimate the probability density $P(\mathbf{x}, y)$. However, the model is underidentified because, for linearly separable data, there are an infinite number of linear functions that can perform the separation without error. To choose one solution among many, the support vector machine defines the *optimal decision function* as the one that leaves the largest possible margin on both sides of the decision boundary. The *margin of the ith point* $(\mathbf{x}_i, y_i)$ with respect to a particular function f is the quantity $\gamma_i = y_i f(x_i)$, which is positive if f correctly classifies the observation, and nonpositive otherwise. Given a specific sample with linearly separable points, the support vector algorithm finds the separating function with maximum margin of the training set with respect to the class of functions under consideration (Cristianini and Shawe-Taylor 2000). The optimal classifier is called the *maximum margin classifier* or *optimal margin classifier*. Vapnik (1998) shows that this classifier is unique to a given data set.

Derivation—Binary Scenario

To derive the maximum margin classifier, one must minimize the norm of the weight vector, $\mathbf{w}$, under the con-

straints that the margin values are greater than or equal to 1, as shown in (A4).

$$\min \quad \tfrac{1}{2}\|\mathbf{w}\|^2 \tag{A4a}$$

$$\text{s.t.} \quad y_i[(\mathbf{w}\cdot\mathbf{x}_i)+w_0] \geq 1 \quad \text{for } i=1,\ l. \tag{A4b}$$

Using Lagrangian multipliers, (A4) can be transformed into its dual form, where the Kuhn-Tucker conditions guarantee a unique solution.[14] The dual form is

$$\text{Max} \quad \sum_{i=1}^{n}\alpha_i - \tfrac{1}{2}\sum_{i,\,j=1}^{n}\alpha_i\alpha_j y_i y_j(\mathbf{x}_i\cdot\mathbf{x}_j) \tag{A5a}$$

$$\text{s.t.} \quad \sum_{i=1}^{n} y_i\alpha_i = 0 \quad \alpha_i \geq 0, \quad i=1,l, \tag{A5b}$$

where α_i is the Lagrange multiplier for inequality i in (A4b). The dual problem is a quadratic program, efficiently solvable using the recursive procedure developed by Platt (1999). Transitioning from (A4) to (A5) yields

$$\mathbf{w}^* = \sum_{i=1}^{n}\alpha_i^* y_i \mathbf{x}_i, \tag{A6}$$

where $\alpha_1^*, \alpha_2^*, \ldots, \alpha_n^*$ are solutions to the problem (A5). The data points with nonzero α_i are the problem's support vectors. The optimal constant, w_0^*, can be derived using any single support vector, but more often—to achieve numerical stability—is calculated using their mean. The optimal decision function can be represented in terms of either the primal or dual optimal solution as shown in A7.

$$f(\mathbf{x}) = \mathbf{w}^*\cdot\mathbf{x} + w_0^* = \sum_{i=1}^{n}\alpha_i^* y_i(\mathbf{x}\cdot\mathbf{x}_i) + w_0^*. \tag{A7}$$

Two properties of the solution are important. First, the optimal boundary is a function only of the relatively few support vectors, not all data points. Although counterintuitive from a sampling perspective, these "inliers" are not as susceptible to boundary bias as are boundaries based on all data points. Second, the size of the dual problem scales directly with the sample size, not with the dimensionality of the data model (e.g., the function class with its respective parameters). Thus, solutions do not suffer from the curse of dimensionality. This property is illustrated in Appendix B, which provides a complete numerical example of an SVM.

Kernel-Induced Transformations

In practice, observations are rarely linearly separable in the original space, but may be linearly separable in a specially constructed higher-dimensional space. The SVM uses a *kernel-induced transformation* ϕ: $R^n \to \Im$ to map the original input space into a higher-dimensional space, called *feature space*. $\Im$ is chosen so that data points appear in the algorithm uniquely in the form of dot products, i.e., functions where the vector inner product, $(\mathbf{x}\cdot\mathbf{x}_i)$ in (A7), takes the form of the inner product between the images $[\phi(\mathbf{x})\cdot\phi(\mathbf{x}_i)]$ in $\Im$. This property is satisfied by a *kernel function*, K, such that $K(\mathbf{x},\mathbf{x}_i) = \phi(\mathbf{x})\cdot\phi(\mathbf{x}_i)$. Replacing $\mathbf{x}\cdot\mathbf{x}_i$ everywhere with $\phi(\mathbf{x})\cdot\phi(\mathbf{x}_i)$, a support vector machine finds an optimal linear boundary in $\Im$, the feature space, which maps to a nonlinear decision function in the original n-dimensional attribute space (see Appendix B for details). The decision function with kernel (A7) becomes (A8).

$$D(\mathbf{x}) = \sum_{i=1}^{n}\alpha_i^* y_i K(\mathbf{x},\mathbf{x}_i) + w_0^*, \tag{A8}$$

where $K(\mathbf{x}\cdot\mathbf{x}_i) = \phi(\mathbf{x})\cdot\phi(\mathbf{x}_i)$, and α_i^* $(i=1,\ldots,l)$ are solutions to the QP maximization problem (A9).

$$\text{Max} \quad \sum_{i=1}^{l}\alpha_i - \tfrac{1}{2}\sum_{i,\,j=1}^{l}\alpha_i\alpha_j y_i y_j K(\mathbf{x},\mathbf{x}_i) \tag{A9a}$$

$$\text{s.t.} \quad \sum_{i=1}^{n} y_i\alpha_i = 0 \quad \alpha_i \geq 0, \quad i=1,l. \tag{A9b}$$

Soft-Margin Classifiers

In practice, a data set normally contains nonseparable observations due to the nature of the problem domain, random error, theoretical ignorance, variable deficiency, and data mislabeling. In such situations, using the maximum margin classifier will lead to "overfitting" noisy data (Cortes and Vapnik 1995, Cristianini and Shawe-Taylor 2000). *Soft-margin classifiers* seek an optimal decision function with maximum margins for observations that *can* be separated accurately and, simultaneously, a minimum number of errors for nonseparable observations. To accomplish this goal, positive slack variables ξ_i $(i=1,l)$ are included in the decision function; $y_i[(\mathbf{w}\cdot\mathbf{x}_i)+w_0] \geq 1-\xi_i;\ i=1,l$. The weight of the slack variables is controlled by a hyperparameter (or *penalty term*) C. This alters the quadratic programming problem, where (A9c) replaces (A9b).

$$\sum_{i=1}^{n} y_i\alpha_i = 0 \quad 0 \leq \alpha_i \leq C, \quad i=1,\ldots,n. \tag{A9c}$$

The soft-margin classifier (A9c) is identical to the maximum margin classifier (A9) if the penalty term C is infinite.

Multiclass Classification Formulation

The SVM classifier described above is binary. Although direct generalization to multigroup classifiers is possible (e.g., Weston and Watkins 1998, Vapnik 1998), the direct approach is not necessarily efficient. SVM researchers often combine binary classifiers to handle multiclass situations (Krebel 1999, Platt et al. 2000). Suppose we have m classes; a simple and effective procedure is to train m one-versus-rest binary classifiers (say, "one" positive, "rest" negative) and assign a test observation to the class with the largest positive distance (Boser et al. 1992, Vapnik 1995). This procedure has been shown to give excellent results and is the method we use in our multiclass tests, described later.

SVMs and Capacity Control

A kernel-induced transformation may result in a very high-dimensional feature space. For example, in classification problems the most often used kernels are polynomial kernels (A10) and Gaussian kernels (A11).

$$K(\mathbf{x}_i,\mathbf{x}_j) = (\mathbf{x}_i\cdot\mathbf{x}_j+1)^d, \tag{A10}$$

$$K(\mathbf{x}_i,\mathbf{x}_j) = e^{-|\mathbf{x}_i-\mathbf{x}_j|^2/2\sigma^2}. \tag{A11}$$

[14] Readers interested in a step-by-step derivation of Equations (A4) and (A5) are referred to Cui and Curry (2003).

If the data have dimensionality n, then the feature space mapped by a polynomial kernel of degree d has dimension $\binom{n+d}{d}$. The dimensionality of a feature space mapped by a Gaussian kernel can be infinite (Burges 1998). A linear function in a very high-dimensional feature space will not yield a machine that generalizes well.[15] However, an SVM looks for a linear separating function with maximum margin. By maximizing the margin of the training set, SVMs control the capacity of the optimal linear function in feature space. In fact, it can be shown that the margin of a training set is an effective capacity measure; i.e., the structural risk bound (1) can be expressed as a function of a measure of margin on the training set (Shawe-Taylor and Cristianini 1999a, b, and c; Cristianini and Shawe-Taylor 2000). Thus, *observing a large margin is equivalent to minimizing the VC-dimension, resulting in good generalization from a small sample*. In fact, Vapnik (1998) shows that the capacity of the support vector machine is bounded when the observations are completely separable.

Appendix B. Kernel Transformations: An Example

We engineer a simple but complete example to explain how kernel transformations work. The support vector machine uses such transformations to map the original data in *attribute space* to a higher-dimensional *feature space*. In feature space, a linear decision function is found that corresponds to the nonlinear function in input space. Equation (B1) shows the dual form of this function, which includes the kernel mapping $K(\mathbf{x}\cdot\mathbf{x}_i)=\phi(\mathbf{x})\cdot\phi(\mathbf{x}_i)$ explained in this appendix.

$$D(x)=\sum_{i=1}^{n}\alpha_i y_i K(\mathbf{x},\mathbf{x}_i)+w_0. \tag{B1}$$

The α_i^* are solutions to the dual QP maximization problem, as outlined in Appendix A, Equations (A5) and (A7).

Example

The four data points shown in Table B1 are members of two groups with coordinates on axes x_1 and x_2 and group membership indexed by $y\in\{-1,+1\}$. The points are shown in Figure B1 using the symbols indicated in the table.

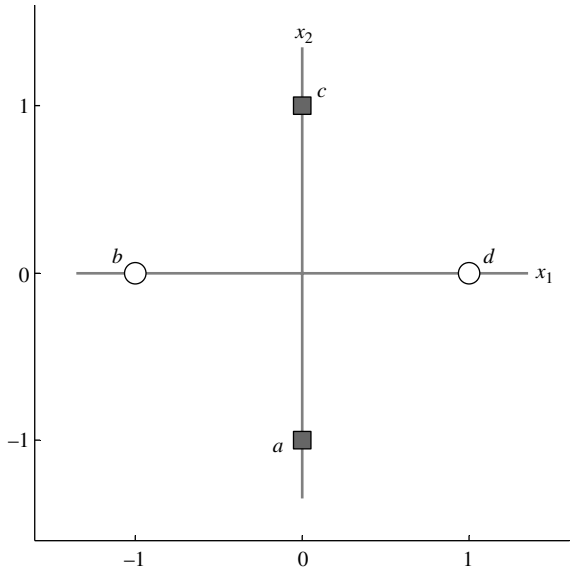
This classification problem is known as the XOR problem in the literature of artificial intelligence. Minsky and Papert (1969) were the first to note that the problem could not be solved by perceptrons. Hinton et al. (1986) rediscovered the problem, and MacKay and Oldfield (1995) presented a solution using a nonlinear, hidden, layered artificial neural net. When implementing this solution, Langley and Burgess (2000) found convergence to be extremely slow.[16]

Figure B1 shows that a linear boundary cannot separate these four points. However, four points can be shattered by a second-degree polynomial. The inner product kernel for a polynomial of degree two is given by the nonlinear function ϕ in (B2).

Table B1 Data Set (Attribute Space)

Data point	x_1	x_2	y	Symbol
a	0	−1	−1	Square
b	−1	0	+1	Circle
c	0	+1	−1	Square
d	+1	0	−1	Circle

Figure B1 Four Points in Two Classes



$$\phi(\mathbf{x},\mathbf{z})=K(\mathbf{x},\mathbf{z})=(\mathbf{x}\cdot\mathbf{z}+1)^2. \tag{B2}$$

Expanding (B2) for our two-dimensional attribute space, we find $\phi(\mathbf{x},\mathbf{z})=\langle\phi(\mathbf{x})\cdot\phi(\mathbf{z})\rangle$ as follows:

$$\begin{aligned}K(\mathbf{x},\mathbf{z})&=\left((x_1\quad x_2)\begin{pmatrix}z_1\\z_2\end{pmatrix}+1\right)^2=(x_1z_1+x_2z_2+1)^2\\&=x_1^2z_1^2+x_2^2z_2^2+2x_1x_2z_1z_2+2x_1z_1+2x_2z_2+1\\&=\begin{pmatrix}x_1^2 & x_2^2 & \sqrt{2}x_1x_2 & \sqrt{2}x_1 & \sqrt{2}x_2 & 1\end{pmatrix}\begin{pmatrix}z_1^2\\z_2^2\\\sqrt{2}z_1z_2\\\sqrt{2}z_1\\\sqrt{2}z_2\\1\end{pmatrix}\\&=\langle\phi(\mathbf{x})\cdot\phi(\mathbf{z})\rangle.\end{aligned} \tag{B3}$$

In words, the kernel function is chosen so that its application to vectors in attribute space will yield the inner product of their representations in feature space. Result (B3) generalizes to (B4), which shows that the transformation

[15] For example, suppose that we have 200 10-dimensional data points. Mapping the 10-dimensional feature space with a polynomial kernel of degree 10 would result in a 184,756-dimensional feature space. A linear function in such a feature space has 184,757 parameters, and can memorize all 200 points.

[16] The data in Table B1 represent a linearly transformed version of an exclusive OR truth table, where class membership is indicated by the binary sum of a point's coordinates. Points belong to class {false = −1} if their scores on each dimension match. Otherwise, they belong to class {true = +1}, and the outcome, $y=x_1+x_2 \pmod 2$.

yields an inner product matrix, albeit between vectors in ϕ-space.

$$\begin{aligned} K(\mathbf{x},\mathbf{z}) &\equiv (\langle \mathbf{x}\cdot\mathbf{z}\rangle + 1)^2 = \left(\sum_{i=1}^{n} x_i z_i + 1\right)\left(\sum_{j=1}^{n} x_j z_j + 1\right) \\ &= \sum_{i,j=(1,1)}^{(n,n)} (x_i x_j)(z_i z_j) + \sum_{i=1}^{n} (\sqrt{2}x_i)(\sqrt{2}z_i) + 1^2 \\ &= \begin{bmatrix} (x_i \cdot x_j)_{i,j=(1,1)}^{(n,n)} & \sqrt{2}x_i & 1 \end{bmatrix} \begin{bmatrix} (z_i \cdot z_j)_{i,j=(1,1)}^{(n,n)} \\ \sqrt{2}z_i \\ 1 \end{bmatrix} \\ &= \langle \phi(\mathbf{x})\cdot\phi(\mathbf{z})\rangle. \end{aligned} \tag{B4}$$

(The algebra goes through for arbitrary d and c. See Cristianini and Shawe-Taylor 2000, Chapter 3). With $d = 2$, the terms involved in this higher-dimensional inner product are quadratic $(x_i x_i) = x_i^2$, bilinear $(x_i x_j)$, linear $(\sqrt{2}x_i)$, and constant (1) terms from the attribute space. The choice of the constant ($c = 1$) determines the relative weights applied to these terms in feature space. The induced structure contains the types of terms—quadratic, bilinear, and higher-order interactions—that may be present in choice data or in other data-generating processes.

Inner product matrices are always square and symmetric, and their size is simply the number of data points. With four points, we expect to see the 4×4 matrix $\mathbf{K}$ shown in (B5). For example, the entry $\mathbf{K}_{2,1} = 1$ is found from the following inner product:

$$\begin{aligned} \mathbf{K}_{2,1} &= \langle -1^2 \quad 0^2 \quad \sqrt{2}\cdot -1\cdot 0 \quad \sqrt{2}\cdot -1 \quad \sqrt{2}\cdot 0 \quad 1\rangle \\ &\quad \cdot \langle 0^2 \quad -1^2 \quad \sqrt{2}\cdot 0\cdot -1 \quad \sqrt{2}\cdot 0 \quad \sqrt{2}\cdot -1 \quad 1\rangle^T = 1, \\ \mathbf{K} &= \begin{bmatrix} 4 & 1 & 0 & 1 \\ 1 & 4 & 1 & 0 \\ 0 & 1 & 4 & 1 \\ 1 & 0 & 1 & 4 \end{bmatrix}. \end{aligned} \tag{B5}$$

$\mathbf{K}$ is known as the *kernel* or *gram matrix* corresponding to the kernel function K. Because $\mathbf{K}$ is an inner product matrix in $\Re^n$, it is a positive semidefinite, symmetric matrix that can be factored as $\mathbf{K} = \mathbf{P}\mathbf{\Delta}\mathbf{P}^T$, where $\mathbf{\Delta}$ is a diagonal matrix of the eigenvalues $\lambda_t \geq 0$ of $\mathbf{K}$ with corresponding eigenvectors, $\mathbf{p}_t = [p_{t1}, \ldots, p_{tn}]^T$ as the columns of $\mathbf{P}$. (See Young and Householder 1940; Bronson 1991, Chapter 9.) Using the mapping ϕ: $\mathbf{x}_i \to (\sqrt{\lambda_t}p_{ti})_{t=1}^{n} \in \Re^n$ $(i = 1, \ldots, n)$, it follows directly that ϕ is a kernel function corresponding to the feature mapping ϕ; e.g.,

$$\langle \phi(\mathbf{x}_i)\cdot\phi(\mathbf{x}_j)\rangle = \sum_{t=1}^{n} \lambda_t p_{ti} p_{tj} = (\mathbf{P}\mathbf{\Delta}\mathbf{P}^T)_{ij} = \mathbf{K}_{ij} = K(\mathbf{x}_i, \mathbf{x}_j). \tag{B6}$$

With a particular choice of weightings in the kernel function (*Mercer kernels*), the analyst can even insure that the features will be orthogonal. Burges (1998) derives this property, which is a deeper result from continuous mathematics in Hilbert spaces (Courant and Hilbert 1953, Vapnik 1995).

In summary, SVM kernels map vectors in a low-dimensional *attribute space* into their inner products in a high-dimensional *feature space*. These functions directly provide the inner product that appears in the dual form of the separating hyperplane quadratic program, avoiding the need to perform calculations in feature space. Because estimation involves only inner products of data points, the problem size scales with sample size, not the dimensionality of the data. This neutralizes the "curse of dimensionality" that afflicts parametric models (see Ben-Akiva et al. 1997, Friedman 1997).

Optimal Decision Function

The optimal decision function is obtained by substituting the four data points into the decision function (B1) and expanding. This yields Expression (B7).

$$\begin{aligned} D(\mathbf{x}) &= \alpha_1 K(\mathbf{x},\mathbf{x}_1) - \alpha_2 K(\mathbf{x},\mathbf{x}_2) + \alpha_3 K(\mathbf{x},\mathbf{x}_3) - \alpha_4 K(\mathbf{x},\mathbf{x}_4) + w_0 \\ &= \alpha_1\left((x_1 \quad x_2)\begin{pmatrix}1\\0\end{pmatrix} + 1\right)^2 - \alpha_2\left((x_1 \quad x_2)\begin{pmatrix}0\\1\end{pmatrix} + 1\right)^2 \\ &\quad + \alpha_3\left((x_1 \quad x_2)\begin{pmatrix}-1\\0\end{pmatrix} + 1\right)^2 \\ &\quad - \alpha_4\left((x_1 \quad x_2)\begin{pmatrix}0\\-1\end{pmatrix} + 1\right)^2 + w_0 \\ &= \alpha_1(x_1+1)^2 - \alpha_2(x_2+1)^2 + \alpha_3(-x_1+1)^2 \\ &\quad - \alpha_4(-x_2+1)^2 + w_0. \end{aligned} \tag{B7}$$

The optimal decision boundary—which is linear in feature space—is found by solving the optimization problem (B8), a constrained quadratic program.

$$\begin{aligned} &\text{Max} \quad Q(a) = \alpha_1 + \alpha_2 + \alpha_3 + \alpha_4 - \tfrac{1}{2}\sum_{i,j=1}^{4} \alpha_i \alpha_j y_i y_j K_{ij} \\ &\text{s.t.} \quad \sum_{i=1}^{4} y_i \alpha_i = \alpha_1 - \alpha_2 + \alpha_3 - \alpha_4 = 0, \\ &\qquad 0 \leq \alpha_i, \quad i = 1, 4. \end{aligned} \tag{B8}$$

Solution

The solution to the QP problem is $\alpha_1^* = \alpha_2^* = \alpha_3^* = \alpha_4^* = 0.5$, indicating that all four points are support vectors. The functional Q reaches its maximum of 1.0 at this point. The decision function in the dual form—which is nonlinear in attribute space—is given by (B9).

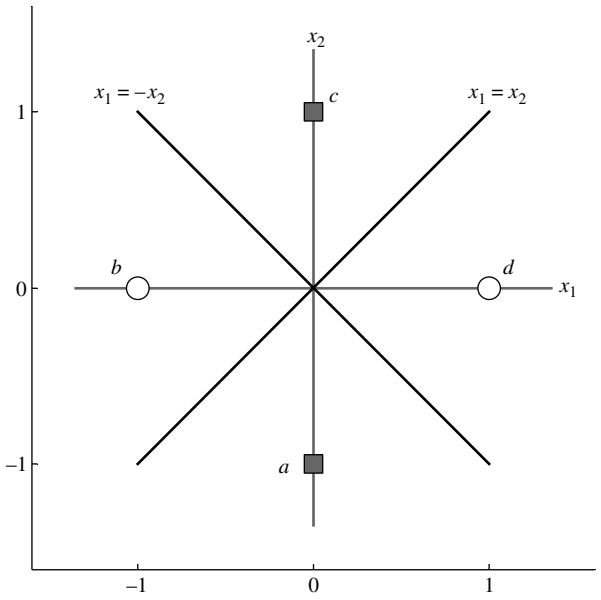
$$\begin{aligned} D(x) &= \sum_{i=1}^{4} \alpha_i^* y_i K(x, x_i) + w_0 \\ &= \tfrac{1}{2}[(x_1+1)^2 - (x_2+1)^2 + (-x_1+1)^2 - (-x_2+1)^2] + w_0 \\ &= x_1^2 - x_2^2 + w_0. \end{aligned} \tag{B9}$$

We can use any one of the support vectors to solve for the constant w_0; i.e., $y_i \cdot D(x_i) = 1$ or solving, we find $w_0^* = 0$. Therefore, the optimal decision boundary has the form $x_1^2 = x_2^2$ or $(x_1 + x_2)(x_1 - x_2) = 0$, which yields the nonlinear boundary function shown in Figure B2.

Appendix C. Structural Diagnostics with the SVM

Suppose we have a data set where each observation consists of an n-dimensional vector $\mathbf{x}_i \in R^n$, $i = 1, \ldots, l$ and an associated output y_i. We use x^k to represent the kth predictor variable, $k \in \{1, \ldots, n\}$. The sensitivity of the output $D(\mathbf{x})$ with respect to the kth input x^k can be determined from the partial derivative $\partial D(\mathbf{x})/\partial x^k$ of the decision function with

Figure B2 The SVM Decision Boundary



respect to x^k. This derivative is evaluated at the observed value of x^k holding other variables x^m, $m \neq k$ at their observed values. The net effect of x^k is then determined by averaging over respondents in each response category for each variable.

In a support vector machine, partial derivatives are easy to determine because the decision function is linear; e.g., $D(\mathbf{x}) = \sum_{j=1}^{s} \alpha_j^* y_j K(\mathbf{x}, \mathbf{x}_j) + w_0^*$. Thus, a given kernel, K, yields Equation (C1).

$$\frac{\partial D(\mathbf{x})}{\partial x^k} = \sum_{j=1}^{s} a_j^* y_j \frac{\partial K(\mathbf{x}, \mathbf{x}_j)}{\partial x^k}. \tag{C1}$$

However, results vary by kernel type. We illustrate for three kernel families—identity, Gaussian, and polynomial—using two-dimensional data $\mathbf{x} = (x^1, x^2)$. If K is the identity transformation, results are carried out in attribute space and

$$\frac{\partial D(\mathbf{x})}{\partial x^1} = \sum_{j=1}^{s} a_j y_j \frac{\partial\left(x_j^1 \cdot x^1 + x_j^2 \cdot x^2\right)}{\partial x^1} = \sum_{j=1}^{s} a_j y_j x_j^1, \tag{C2}$$

which is a constant as expected. If a Gaussian kernel is used, we have the chain of reasoning shown as (C3).

$$\begin{aligned} \frac{\partial D(\mathbf{x})}{\partial x^1} &= \sum_{j=1}^{s} a_j y_j \frac{\partial \exp\left(-(x_j^1 \cdot x^1 + x_j^2 \cdot x^2)/2\pi\sigma^2\right)}{\partial x^1} \\ &= \sum_{j=1}^{s} a_j y_j \frac{-x_j^1}{2\pi\sigma^2} \exp\left(-\frac{\left(x_j^1 \cdot x^1 + x_j^2 \cdot x^2\right)}{2\pi\sigma^2}\right). \end{aligned} \tag{C3}$$

This expression can be easily evaluated. If a polynomial kernel is used, we have Equation (C4). Similarly, we can derive sensitivity formulae for other types of kernels.

$$\begin{aligned} \frac{\partial D(\mathbf{x})}{\partial x^1} &= \sum_{j=1}^{s} a_j y_j \frac{\partial\left(x_j^1 \cdot x^1 + x_j^2 \cdot x^2 + 1\right)^d}{\partial x^1} \\ &* = \sum_{j=1}^{s} a_j y_j x_j^1 d\left(x_j^1 \cdot x^1 + x_j^2 \cdot x^2 + 1\right)^{d-1}. \end{aligned} \tag{C4}$$

Posttraining when s, a_j, and the *support vectors* are known, the average sensitivity of output $D(\mathbf{x})$ with respect to output x^k evaluated at a given input response category is calculated by integrating over respondents who choose category C, denoted $l_{(y_i=C)}$.

$$S_{x^k} = \frac{1}{l_{(y_i=C)}} \sum_{i}^{l_{(y_i=C)}} \frac{\partial D(\mathbf{x})}{\partial x^k}. \tag{C5}$$

References

Abraham, Magid, Len Lodish. 1987. Promoter: An automated promotion evaluation system. *Marketing Sci.* **6**(1) 1–25.

Abraham, Magid, Len Lodish. 1993. An implemented system for improving promotion productivity using store scanner data. *Marketing Sci.* **12**(3) 248–269.

Allenby, Greg M., Neeraj Arora, James L. Ginter. 1995. Incorporating prior knowledge into the analysis of conjoint studies. *J. Marketing Res.* **35**(May) 152–162.

Allenby, Greg, Neeraj Arora, Chris Diener, Jaehwan Kim, Mike Lotti, Paul Markowitz. 2002. Distinguishing likelihoods, loss functions and heterogeneity in the evaluation of marketing models. *Canadian J. Marketing Res.* **20**(1) 44–59.

Andrews, Rick L., Asim Ansari, Imran S. Currim. 2002. Hierarchical Bayes versus finite mixture conjoint analysis models: A comparison of fit, prediction, and partworth recovery. *J. Marketing Res.* **39**(February) 87–98.

Ariely, Dan, John G. Lynch Jr., Manuel Aparicio, IV. 2004. Learning by collaborative and individual-based recommendation agents. *J. Consumer Psych.* **14**(1 & 2) 81–95.

Avery, Christopher, Paul Resnick, Richard Zeckhauser. 1999. The market for evaluations. *Amer. Econom. Rev.* **89**(June) 564–584.

Bellman, Richard E. 1961. *Adaptive Control Processes: A Guided Tour*. Princeton University Press, Princeton, NJ.

Ben-Akiva, Moshe, Steven R. Lerman. 1985. *Discrete Choice Analysis: Theory and Application to Travel Demand*. MIT Press, Cambridge, MA.

Ben-Akiva, Moshe, Daniel McFadden, Abe Makoto, Ulf Bockenholt, Denis Bolduc, Dinesh Gopinath, Takayuki Morikawa, Venkatram Ramaswamy, Vithala Rao, David Revelt, Dan Steinberg. 1997. Modeling methods for discrete choice analysis. *Marketing Lett.* **8**(3) 273–286.

Berry, Michael, Gordon Linoff. 1997. *Data Mining Techniques: for Marketing, Sales, and Customer Support*. John Wiley and Sons, New York.

Blattberg, Robert C., Byung-Do Kim, Jianming Ye. 1994. Large-scale databases: The new marketing challenge. Robert C. Blattberg, Rashi Glazer, John D. C. Little, eds. *The Marketing Information Revolution*. Harvard Business School Press, Boston, MA, 173–203.

Boser, Bernhard E., Isabelle M. Guyon, Vladimir Vapnik. 1992. A trained algorithm for optimal margin classifier. *Fifth Annual Workshop on Computational Learning Theory*. ACM, Pittsburgh, PA, 144–151.

Bradlow, Eric T. 2003. Current issues and a "wish list" for conjoint analysis. Working paper, The Wharton School of the University of Pennsylvania, Philadelphia, PA, 1–10.

Brazerman, Max H. 1994. *Judgment in Managerial Decision Making*, 3rd ed. J. Wiley, New York.

Bronson, Richard. 1991. *Matrix Methods: An Introduction*, 2nd ed. Academic Press, Inc., Boston, MA.

Bucklin, Randolph E., Donald R. Lehmann, John D. C. Little. 1998. From decision support to decision automation: A 2020 vision. *Marketing Lett.* **9**(3) 235–246.

Bucklin, Randolph E., James M. Lattin, Asim Ansari, Sunil Gupta, David Bell, Eloise Coupey, John D. C. Little,Carl Mela, Alan

Montgomery, Joel Steckel. 2002. Choice and the Internet: From clickstream to research stream. *Marketing Lett.* **13**(3) 245–258.

Burges, Christopher J. C. 1998. A tutorial on support vector machines for pattern recognition. *Data Mining Knowledge Discovery* **2** 121–167.

Chapelle, Olivier, Patrick Hallner, Vladimir Vapnik. 1999. SVM for histogram-based image classification. *IEEE Trans. Neural Networks* **10**(5) 1055–1064.

Cooper, Lee G., Giovanni Giuffrida. 2000. Turning datamining into a management science tool. *Management Sci.* **46**(2) 249–264.

Cortes, Corinna, Vladimir Vapnik. 1995. Support vector networks. *Machine Learning* **20** 273–297.

Courant, Rourant, David Hilbert. 1953. *Methods of Mathematical Physics*, Vol. 1. Interscience Publishers, Inc., New York.

Cristianini, Nello, John Shawe-Taylor. 2000. *An Introduction to Support Vector Machines—and Other Kernel-Based Learning Methods*. Cambridge University Press, Cambridge, UK.

Cui, Dapeng, David Curry. 2003. Applications of support vector machines in marketing: An exposition. Working paper.

Dawes, Robyn M., B. Corrigan. 1974. Linear models in decision-making. *Psych. Bull.* **81** 95–106.

Diehl, Kristin R., Laura J. Kornish, John G. Lynch, Jr. 2003. Smart agents: When lower search costs for quality information increase price sensitivity. *J. Consumer Res.* **30**(June) 56–71.

Domencich, Thomas A., Daniel McFadden. 1975. *Urban Travel Demand: A Behavioral Analysis*. North-Holland Publishing Co., Amsterdam, The Netherlands.

Edgington, Eugene S. 1995. *Randomization Tests*, 3rd ed. Marcel Dekker, Inc., New York.

Efron, Bradley, Robert J. Tibshirani. 1993. *An Introduction to the Bootstrap*. Chapman and Hall, New York.

Einhorn, Hillel J. 1970. The use of nonlinear, non-compensatory models in decision making. *Psych. Bull.* **73** 221–230.

Evgeniou, Theodoros, Constantinos Boussios, Giorgos Zacharia. 2005. Generalized robust conjoint estimation. *Marketing Sci.* **24**(3) 415–429.

Fisher, Ronald A. 1950. *Contributions to Mathematical Statistics*. Wiley, New York.

Friedman, Jerome H. 1997. On bias, variance, 0/1—Loss, and the curse of dimensionality. *Data Mining Knowledge Discovery* **1** 55–77.

Gershoff, Andrew, Patricia M. West. 1998. Using a community of knowledge to build intelligent agents. *Marketing Lett.* **9**(January) 79–91.

Gilbride, Timothy J., Greg M. Allenby. 2004. A choice model with conjunctive, disjunctive, and compensatory screening rules. *Marketing Sci.* **23**(3) 391–406.

Good, Phillip I. 2005. *Resampling Methods: A Practical Guide to Data Analysis*, 3rd ed., Vol. XX. Birkhäuser Book (Springer), Boston, MA.

Guadagni, Peter M., John D. C. Little. 1983. A logit model of brand choice calibrated on scanner data. *Marketing Sci.* **2**(3) 203–238.

Guyon, Isabelle, André Elisseeff. 2003. An introduction to variable and feature selection. *J. Machine Learning Res.* **3**(March) 1157–1182.

Hahn, G. J., S. S. Shapiro. 1966. A catalog and computer program for the design and analysis of orthogonal symmetric and asymmetric fractional factorial experiments. Technical report no. 66-C-165, General Electric Research and Development Center, Schenectady, NY.

Häubl, Gerald, Valerie Trifts. 2000. Consumer decision making in online shopping environments: The effects of interactive decision aids. *Marketing Sci.* **19**(1) 4–21.

Hempel, Carl G. 1965. Aspects of scientific explanation. *Aspects of Scientific Explanation and Other Essays in the Philosophy of Science*. The Free Press, New York, 331–496.

Hinton, G. E., J. L. McClelland, D. E. Rumelhart. 1986. Distributed representations. D. E. Rumelhart, J. L. McClelland, the PDP Research Group, eds. *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*. MIT Press, Cambridge, MA, 77–109.

Hofstede, Frenkel Ter, Youngchan Kim, Michel Wedel. 2002. Bayesian prediction in hybrid conjoint analysis. *J. Marketing Res.* **39**(May) 253–261.

Hunt, Shelby D. 1983. *Marketing Theory: The Philosophy of Marketing Science*. Richard D. Irwin, Inc., Homewoods, IL.

Iacobucci, Dawn, Phipps Arabie, Anand Bodapati. 2000. Recommendation agents on the Internet. *J. Interactive Marketing* **14**(3) 2–11.

Kahneman, Daniel. 2002. Maps of bounded rationality: A perspective on intuitive judgment and choice. *Nobel Prize Lecture* (December 8) 449–489. http://nobelprize.org/ economics/laureates/2002.

Kamakura, Wagner A., Michel Wedel, Fernando de Rosa, Jose Afonso Mazzon. 2003. Cross-selling through database marketing: A mixed data factor analyzer for data augmentation and prediction. *Internat. J. Res. Marketing* **20**(1) 45–65.

Kardes, Frank R. 1999. *Consumer Behavior and Managerial Decision Making*. Addison-Wesley, Reading, MA.

Krebel, Ulrich. 1999. Pairwise classification and support vector machines. Bernhard Schölkopf, Christopher J. C. Burges, Alexander J. Smola, eds. *Advances in Kernel Methods—Support Vector Learning*. MIT Press, Cambridge, MA, 255–268.

Langley, Keith, Neil Burgess. 2000. Linear and nonlinear hidden units in artificial neural networks. Under review.

Lilien, Gary L., Arvind Rangaswamy, Gerrit H. van Bruggen, Berend Wierenga. 2002. Bridging the marketing theory-practice gap with marketing engineering. *J. Bus. Res.* **55** 111–121.

Little, John D. C. 2001. Marketing automation on the Internet. *5th Invitational Choice Symposium*, U. C. Berkeley/Asilomar, June 1–5, Berkeley, CA.

Luce, Duncan R. 1959. *Individual Choice Behavior: A Theoretical Analysis*. Wiley, New York.

MacKay, David J. C., Martin J. Oldfield. 1995. Generalization error and the number of hidden units in a multilayer perceptron. Working paper, Cambridge University, Cambridge, UK.

Mattern, David, Simon Haykin. 1999. Support vector machines for dynamic reconstruction of a chaotic system. Bernhard Schölkopf, Christopher J. C. Burges, Alexander J. Smola, eds. *Advances in Kernel Methods—Support Vector Learning*. MIT Press, Cambridge, MA, 211–242.

Minsky, Marvin, Seymour Papert. 1969. *Perceptrons: An Introduction to Computational Geometry*. MIT Press, Cambridge, MA.

Moe, Wendy W., Peter S. Fader. 2004. Dynamic conversion behavior at e-commerce sites. *Management Sci.* **50**(3) 326–335.

Müller, Klaus-Robert, Alexander J. Smola, Gunnar Rätsch, Bernhard Schökopf, Jens Kohlmorgen, Vladimir Vapnik. 1999. Using support vector machines for time series prediction. Bernhard Schölkopf, Christopher J. C. Burges, Alexander J. Smola, eds. *Advances in Kernel Methods—Support Vector Learning*. MIT Press, Cambridge, MA, 243–253.

Osuna, Edgar, Robert Freund, Federico Girosi. 1997. Training support vector machines: An application to face detection. *Proc. Comput. Vision Pattern Recognition*, 130–136.

Platt, John C. 1999. Fast training of support vector machines using sequential minimal optimization. Bernhard Schölkopf, Christopher J. C. Burges, Alexander J. Smola, eds. *Advances in Kernel Methods—Support Vector Learning*. MIT Press, Cambridge, MA, 185–208.

Platt, John C. 2000. Probabilities for SV machines. Alexander J. Smola, Peter L. Bartlett, Bernhard Schölkopf, Dale Schuurmans, eds. *Advances in Large Margin Classifiers*. MIT Press, Cambridge, MA, 61–74.

Platt, John C., Nello Cristianni, John Shawe-Taylor. 2000. Large margin DAGs for multiclass classification. S. A. Solla, T. K. Leen, K.-R. Müller, eds. *Advances in Neural Information Processing Systems*. MIT Press, Cambridge, MA, 547–553.

Politz, Alfred, W. Edwards Deming. 1953. On the necessity to present consumer preferences as predictions. *J. Marketing* (July). [Reprinted in *Marketing Res.* (June 1990) 50–55.]

Ratner, Bruce. 2003. *Statistical Modeling and Analysis for Database Marketing: Effective Techniques for Mining Big Data*. CRC Press. http://dmstat1.com.

Rossi, Peter E., Greg M. Allenby. 2003. Bayesian statistics and marketing. *Marketing Sci.* **22** 304–328.

Russo, J. Edward, Paul J. H. Shoemaker. 1989. *Decision Traps*. Doubleday/Currency, New York.

Sándor, Zsolt, Kenneth, Train. 2004. Quasi-random simulation of discrete choice models. *Transportation Res. Part B* **38** 313–327.

Schmitz, John, Gordon O. Armstrong, John D. C. Little. 1990. CoverStory—Automated news finding in marketing. *Interfaces* **20**(6) 29–38.

Schölkopf, Bernhard, Alex J. Smola, Klaus-Robert Müller. 1999. Kernel principle component analysis. Bernhard Schölkopf, Christopher J. C. Burges, Alexander J. Smola, eds. *Advances in Kernel Methods—Support Vector Learning, Advances in Kernel Methods*. MIT Press, Cambridge, MA, 327–352.

Schölkopf, Bernhard, Peter Barlett, Alex Smola, Robert Williamson. 1999. Shrinking the tube: A new support vector regression algorithm. Michael S. Kearns, Sara A. Solla, David A. Cohn, eds. *Advances in Neural Information Processing Systems*. MIT Press, Cambridge, MA, 330–336.

Schölkopf, Bernhard, P. Simard, Alex J. Smola, Vladimir Vapnik. 1998. Prior knowledge in support vector kernels. Michael I. Jordan, Michael S. Kearns, Sara A. Solla, eds. *Advances in Neural Information Processing Systems*, Vol. 10. MIT Press, Cambridge, MA, 640–646.

Shawe-Taylor, John, Nello Cristianini. 1999a. Margin distribution and soft margin. A. J. Smola, P. Bartlett, B. Scholkopf, C. Schuurmans, eds. *Advances in Large Margin Classifiers*. MIT Press, Cambridge, MA, 349–357.

Shawe-Taylor, John, Nello Cristianini. 1999b. Margin distribution bounds on generalization. *Proc. Eur. Conf. Comput. Learning Theory, EuroColt'99*, 263–273.

Shawe-Taylor, John, Nello Cristianini. 1999c. Further results on the margin distribution. *Proc. Conf. Comput. Learning Theory*, COLT 99, 278–285.

Sismeiro, Catarina, Randolph E. Bucklin. 2004. Modeling purchase behavior at an e-commerce web site: A task completion approach. *J. Marketing Res.* **XLI** 306–323.

Smith, Michael D., Erik Brynjolfsson. 2001. Consumer decision making at an Internet Shopbots brand still matters. *J. Indust. Econom.* **XLIX**(4) 541–558.

Stitson, Mark O., Alex Gammerman, Vladimir Vapnik, Volodya Vovk, Chris Watkins, Jason Weston. 1999. Support vector regression with ANOVA decomposition kernels. Bernhard Schölkopf, Christopher J. C. Burges, Alexander J. Smola, eds. *Advances in Kernel Methods—Support Vector Learning*. MIT Press, Cambridge, MA, 285–292.

Thurstone, L. L. 1927. A law of comparative judgment. *Psych. Rev.* **34** 273–286.

Toubia, Oliver, John R. Hauser, Duncan I. Simester. 2004. Polyhedral methods for adaptive choice-based conjoint analysis. *J. Marketing Res.* **41**(February) 116–131.

Tversky, Amos. 1972. Elimination by aspects: A theory of choice. *Psych. Rev.* **79**(July) 281–299.

Vapnik, Vladimir. 1979. *Estimation of Dependences Based on Empirical Data*. Springer Verlag, New York.

Vapnik, Vladimir. 1995. *The Nature of Statistical Learning Theory*. Springer Verlag, New York.

Vapnik, Vladimir. 1998. *Statistical Learning Theory*. Wiley, New York.

Vapnik, Vladimir, A. Chervonenkis. 1964. A note on one class of perceptrons. *Automation Remote Control* **25**(1).

Vapnik, Vladimir, A. Chervonenkis. 1971. On the uniform convergence of relative frequencies of events to their probabilities. *Theory Probab. Its Appl.* **16**(2) 264–280.

Vapnik, Vladimir, Olivier Chapelle. 2000. Bounds on error expectation for SVM. Bernhard Schölkopf, Christopher J. C. Burges, Alexander J. Smola, eds. *Advances in Large Margin Classifiers*. MIT Press, Cambridge, MA, 261–280.

Viaene, Stijn, Richard A. Derrig, Bart Baesens, Guido Dedene. 2002. A comparison of state-of-the-art classification techniques for expert automobile insurance claim fraud detection. *J. Risk Insurance* **69**(3) 373–421.

West, Patricia M., Patrick L. Brockett, Linda L. Golden. 1997. A comparative analysis of neural networks and statistical methods for predicting consumer choice. *Marketing Sci.* **16**(4) 370–391.

West, Patricia M., Dan Ariely, Steve Bellman, Eric Bradlow, Joel Huber, Eric Johnson, Barbara Kahn, John D. C. Little, David Schkade. 1999. Agents to the rescue? *Marketing Lett.* **10**(3) 285–300.

Weston, Jason, Chris Watkins. 1998. Support vector machines for multi-class pattern recognition. Michel Verleysen, ed. *Proc. 6th Eur. Sympos. Artificial Neural Networks* (*ESANN*), Bruges, Belgium, 276–288.

Weston, Jason, Alex Gammerman, Mark O. Stitson, Vladimir Vapnik, Volodya Vovk, Chris Watkins. 1999. Support vector density estimation. Bernhard Schölkopf, Christopher J. C. Burges, Alexander J. Smola, eds. *Advances in Kernel Methods—Support Vector Learning*. MIT Press, Cambridge, MA, 293–306.

Wierenga, B., G. H. van Bruggen. 2000. *Marketing Management Support Systems: Principles, Tools and Implementation*. Kluwer Academic Publishing, Boston, MA.

Wierenga, Beremd, Gerrit H. van Bruggen, Richard Staelin. 1999. The success of marketing management support systems. *Marketing Sci.* **18**(3) 196–207.

Young, Gale, A. S. Householder. 1940. Factorial invariance and significance. *Psychometrika* **5** 47–56.