

Model Selection Using Database Characteristics: Developing a Classification Tree for Longitudinal Incidence Data

Eric M. Schwartz

Stephen M. Ross School of Business, University of Michigan, Ann Arbor, Michigan 48109, ericmsch@umich.edu

Eric T. Bradlow, Peter S. Fader

The Wharton School, University of Pennsylvania, Philadelphia, Pennsylvania 19104
{ebradlow@wharton.upenn.edu, faderp@wharton.upenn.edu}

When managers and researchers encounter a data set, they typically ask two key questions: (1) Which model (from a candidate set) should I use? And (2) if I use a particular model, when is it going to likely work well for my business goal? This research addresses those two questions and provides a rule, i.e., a decision tree, for data analysts to portend the “winning model” before having to fit any of them for longitudinal incidence data. We characterize data sets based on managerially relevant (and easy-to-compute) summary statistics, and we use classification techniques from machine learning to provide a decision tree that recommends when to use which model. By doing the “legwork” of obtaining this decision tree for model selection, we provide a time-saving tool to analysts. We illustrate this method for a common marketing problem (i.e., forecasting repeat purchasing incidence for a cohort of new customers) and demonstrate the method’s ability to discriminate among an integrated family of a hidden Markov model (HMM) and its constrained variants. We observe a strong ability for data set characteristics to guide the choice of the most appropriate model, and we observe that some model features (e.g., the “back-and-forth” migration between latent states) are more important to accommodate than are others (e.g., the inclusion of an “off” state with no activity). We also demonstrate the method’s broad potential by providing a general “recipe” for researchers to replicate this kind of model classification task in other managerial contexts (outside of repeat purchasing incidence data and the HMM framework).

Keywords: model selection; machine learning; data science; business intelligence; hidden Markov models; classification tree; random forest; posterior predictive model checking; hierarchical Bayesian methods; forecasting

History: Received: August 2, 2011; accepted: September 4, 2013; Preyas Desai served as the editor-in-chief for this article. Published online in *Articles in Advance* January 17, 2014.

1. Introduction

The explosion in technology-enabled data collection has changed the focus of marketing modelers away from aggregated data at the store or market level toward more granular, panel-oriented data structures and associated statistical methodologies. Companies have reduced their reliance on “rolled-up” data provided by syndicated vendors (e.g., IRI, Nielsen) and now build more of their analytics around customer-level longitudinal patterns that they can obtain from their own internal operations. But while this increased reliance on “site-centric” data (Zheng et al. 2011) offers a number of meaningful benefits to the firm, it also comes with some potential costs to the researcher.

First, site-centric data provide a detailed description of each customer’s stream of purchases (and other actions that the firm can measure directly), but such data often lack information about marketing variables, competitive tactics, and other potential

“drivers” of the behavior(s) of interest (Donkers et al. 2007, Schweidel et al. 2008) that are typically provided by a third-party firm and are often difficult to link to purchase data. Thus, many firms are focusing their decision-making efforts around the flow of incidence activities, i.e., the timing and nature of each transaction, which is very rich but also quite different from the inputs used in more traditional marketing-mix models (Hanssens et al. 2005).

Second, this detailed stream of incidence actions can be characterized by an ever-larger swath of mathematical models. That is, increased granularity comes with the potential for increased model complexity and hence a more difficult model selection problem than faced by previous generations of researchers, who often relied on relatively standard model specifications (Cooper and Nakanishi 1988, Wittink et al. 1988) that were sufficient for the relatively standard data structures made available by a small set of third-party data providers.

With this “data evolution” in mind, consider a business intelligence manager for an e-commerce firm who is examining panel data from three recent product launches (see Figure 1). Her goal is to project repeat purchase patterns for each data set because her company’s production, marketing, and customer relationship management activities depend on an accurate forecast. How should she choose which statistical model is most appropriate for each product’s data set? She could run a number of different panel-oriented incidence models and choose the one that fits each data set best, but a series of separate “model bake-offs” would be a highly inefficient process and would offer no assurances that the chosen model(s) will be best suited for forecasting purposes of similar data sets. Instead, are there clues in each data set that might help her make the right choices without having to run an array of models over and over again for each new data set? Can we look at many data sets and model performances to extract general rules about when to use which model? That is the goal of this paper: We want to help managers choose among competing longitudinal incidence models, based only on observed data set-level summary statistics, i.e., *database characteristics*, before they need to run any models.

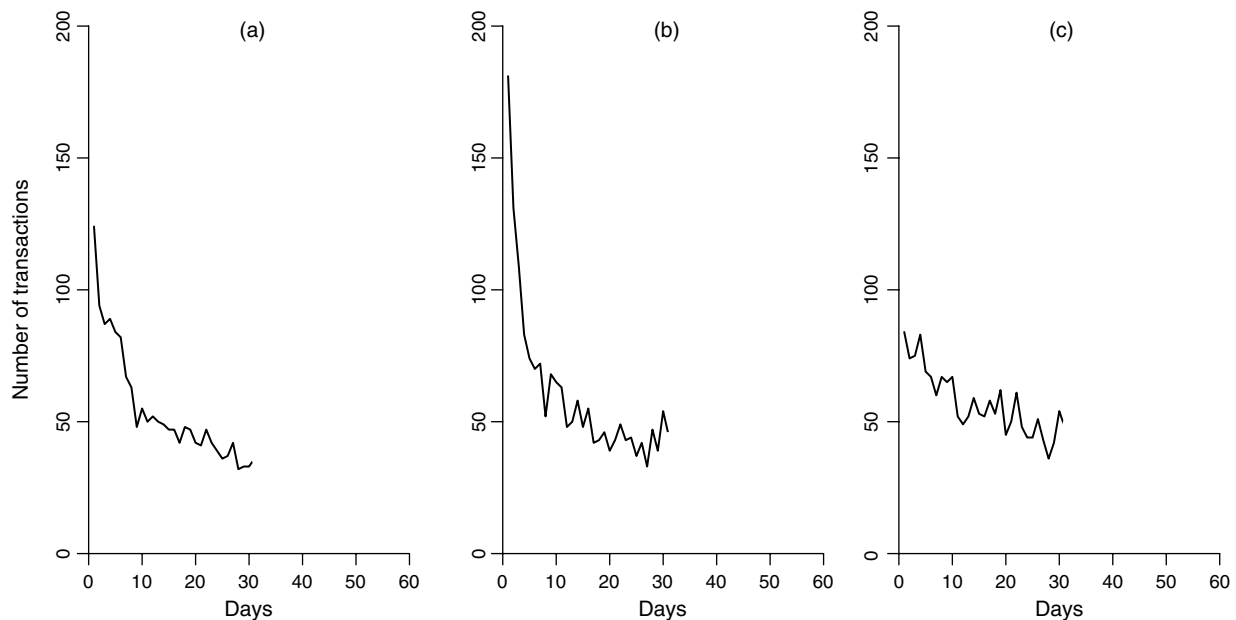
We will create a “decision tree” that can guide the manager toward the most appropriate model specification for a given data set, based only on observable (and easy-to-compute) summary statistics on that data set. In other words, we will do the “up-front work” so that the decision tree is a time-saving tool for other analysts. We recognize that each data set

consists of a mix of heterogeneous customers who may go through different kinds of dynamic purchasing patterns over time, and we want to identify the most suitable model specification to capture these within- and across-customer sources of variation. However, we do not use the database characteristics directly in our models to predict future purchasing (i.e., we do not treat them as X variables in a statistical model), but instead, we use them to help identify the best model (chosen from a class of different latent-state model specifications) that can be used for forecasting and other diagnostic purposes.

For instance, referring back to Figure 1, data set A’s steadily declining sales may indicate that latent customer attrition is prevalent but occurs at different rates for different customers, so a “buy till you die” model such as the Pareto/NBD (Schmittlein et al. 1987) or the beta-geometric/beta-binomial (BG/BB; see Fader et al. 2010) model might be appropriate. In contrast, the sales for data set B seem to show a substantial rise toward the end of the observation period, so a hidden Markov model (HMM), in which customers move back and forth between different states of purchasing propensities (Liechty et al. 2003, Netzer et al. 2008), might be the best model to employ for forecasting purposes. Finally, the sales curve for data set C is harder to classify as a buy till you die or an HMM-type pattern—it seems to reflect elements of both specifications. Perhaps we need a hybrid version of these two models to capture and project it.

Although there are innumerable models that could be viable candidates for this kind of longitudinal incidence data, we choose a particular set that is tightly

Figure 1 Three Different Commonly Observed Patterns of Aggregate Sales over Time (e.g., Arising from a Product Launch or Repeat Purchasing of Cohort of New Customers)



connected to each other but still very flexible. The models we consider are the HMM and three different constrained variants of it (including the BG/BB model). Since they are part of an integrated family, they offer an opportunity to detect when each underlying model component (in this case, the presence of an absorbing state and/or the need for a “no-purchase” state) is worth including or “turning off.” This provides added insight to the analyst about the nature of customer dynamics, above and beyond simplified model implementation and improved model performance.

For this context (i.e., repeat-transaction incidence data, the HMM and its constrained variants), we do all of the “legwork” for the analyst. We run an array of constrained and unconstrained HMM models on dozens of synthetic data sets, generated to broadly represent the kinds of patterns that are likely to occur in real-world settings. Although this is computationally expensive initially (a high up-front cost for us as the researcher), it yields significant savings for the downstream user—the manager simply follows our advice and selects the most appropriate model given the nature of her data set and runs it—the “winner”—and not the entire class of models.

The focal managerial criterion we use to select among models is the forecast error for each cohort’s purchases, so the winning model has the minimum mean absolute error in a holdout period. We use well-established machine-learning methods known as *classification and regression trees* (CART) and *random forests* to derive general rules to suggest which model to use under different circumstances, based entirely on observed (and managerially meaningful) patterns in the customer-base data. The database characteristics that turn out to be most important (in our setting) include the nature of the decline in cohort-level sales over time as well as purchase concentration (e.g., the “80:20 rule”) across customers.

Since the development of the decision tree is our key contribution, the structure of the paper centers around it. There are three “ingredients” for the classification approach, and we devote a section of the paper to each one: the candidate models in §2, the database characteristics in §3, and the performance criterion to determine the “winning” model for each data set in §4. Putting these three ingredients together, we create the decision tree in §5 and focus on its interpretation, validation, and managerial implications.

Although we perform our analysis for a specific data/modeling context (albeit an important one in today’s marketing environment), the same basic “recipe” developed here can be applied to many other settings. Thus, we formalize our approach as a more general methodology in Appendix B, using the same ingredients outlined above: a set of models, database

characteristics, and a selection criterion (i.e., performance or error measure with loss function). We now begin the process of laying out these elements to build our decision tree.

2. Which Models to Consider? The HMM and Its Constrained Variants

The decision tree recommends which model to use for a given data set, but we have to provide a consideration set of models: the HMM and its constrained variants. Why do we consider this class? First, they are appropriate for this popular context of understanding and projecting repeat-purchase patterns of a cohort of customers using longitudinal incidence data (Liechty et al. 2003, Montgomery et al. 2004, Montoya et al. 2010, Netzer et al. 2008, Schweidel et al. 2011). Second, these models cover a wide range of underlying “stories” of customer behavior, leading to different observable data patterns. This helps us achieve the goal of the paper: establish the link between data set-level summaries and model performance.

Third, these models form an integrated family; that is, each model is a constrained or unconstrained version of another in the set. Some of these are established yet seemingly unrelated models, such as buy till you die and latent-class models, among others. But they are all special cases of the HMM. These connections have only been partially explored and in an ad hoc manner in the previous literature, as we discuss below (Netzer et al. 2008, Schweidel et al. 2011). However, the extra insight that we provide is that the four variants of the HMM that we consider are described by two model components each with two levels (as seen in the 2×2 framework discussed in Table 1). So the decision tree not only recommends a specific model but also emphasizes the presence or absence of more general model components, thereby adding more insight and comparability across data sets.

The unconstrained HMM used here has two states, and the within-state purchase likelihoods are repeated Bernoulli trials by individuals who can begin the calibration period in state 1 or state 2. We allow for unobserved continuous heterogeneity for the within-state purchase propensities as well as the between-state transition probabilities. Formally stated, we let $y_{it} = 1$ for day t if the customer i purchased and $y_{it} = 0$ otherwise.¹ Then,

$$y_{it} \sim \begin{cases} \text{Bernoulli}(p_{1i}) & \text{if in state 1, } Z_{it} = 1, \\ \text{Bernoulli}(p_{2i}) & \text{if in state 2, } Z_{it} = 2, \end{cases} \quad (1)$$

¹Without loss of generality, we use “day” to refer to the unit of discrete time and “purchase” as the observed behavior of interest. It could be, instead, for example, viewing online videos or not in a given week, donating or not in a given quarter, etc.

Table 1 Nested Model Relationships Among the HMM and Its Constrained Variants

	State 2 is cold	State 2 is off
Backward and forward transitions	<p>HMM</p> $\mathbf{p}_i = (p_{1i}, p_{2i}) \text{ and } \Theta_i = \begin{pmatrix} 1 - \theta_{12i} & \theta_{12i} \\ \theta_{21i} & 1 - \theta_{21i} \end{pmatrix}$	<p>On and off</p> $\mathbf{p}_i = (p_{1i}, 0) \text{ and } \Theta_i = \begin{pmatrix} 1 - \theta_{12i} & \theta_{12i} \\ \theta_{21i} & 1 - \theta_{21i} \end{pmatrix}$
Only forward transitions	<p>Hot then cold</p> $\mathbf{p}_i = (p_{1i}, p_{2i}) \text{ and } \Theta_i = \begin{pmatrix} 1 - \theta_{12i} & \theta_{12i} \\ 0 & 1 \end{pmatrix}$	<p>BG/BB</p> $\mathbf{p}_i = (p_{1i}, 0) \text{ and } \Theta_i = \begin{pmatrix} 1 - \theta_{12i} & \theta_{12i} \\ 0 & 1 \end{pmatrix}$

Notes. The rows and columns illustrate how constraints on two model components lead to different models. The \mathbf{p}_i and Θ_i are the individual-level within-state transaction probabilities and between-state transition probability matrix, respectively.

where the latent-state variable, Z_{it} , indicates which state a customer occupies on each day. The individual-level parameters of the HMM are the within-state propensities, \mathbf{p}_i , and the transition probability matrix, Θ_i . That is,

$$\mathbf{p}_i = (p_{1i}, p_{2i}) \text{ and } \Theta_i = \begin{pmatrix} 1 - \theta_{12i} & \theta_{12i} \\ \theta_{21i} & 1 - \theta_{21i} \end{pmatrix}. \quad (2)$$

We let the initial state membership be a population-level parameter, and any individual can start in state 1 with probability π_1 or state 2 with probability $1 - \pi_1$. We assume independent beta distributions to allow for heterogeneity across individuals for the components of \mathbf{p}_i and Θ_i .² Specifically, the prior distributions used are

$$\begin{aligned} p_{1i} &\sim \text{beta}(\mu_{p_1}, \phi_{p_1}), & p_{2i} &\sim \text{beta}(\mu_{p_2}, \phi_{p_2}), \\ \theta_{12i} &\sim \text{beta}(\mu_{\theta_{12}}, \phi_{\theta_{12}}), & \theta_{21i} &\sim \text{beta}(\mu_{\theta_{21}}, \phi_{\theta_{21}}), \end{aligned} \quad (3)$$

where $\mu = a/(a + b)$ is the mean and $\phi = 1/(a + b + 1)$ is the polarization index of the beta distribution with shape parameters a and b (Sabavala and Morrison 1977). In general, for $S \geq 2$ states, each row r of the transition probability matrix is a vector $(\theta_{r1i}, \dots, \theta_{rSi}) \sim \text{Dirichlet}(\alpha_{r1}, \dots, \alpha_{rS})$.³ We distinguish between states by referring to state 1 as having a within-state propensity at least as large as that of state 2 for each individual (i.e., $p_{1i} \geq p_{2i}$ for all i). This prevents the label-switching problem known to exist with latent-state models (Stephens 2000).

We highlight the off-diagonal entries of the transition probability matrix, since θ_{12i} denotes the probability an individual moves “forward” (state 1 to 2) and θ_{21i} represents the probability an individual

moves “backward” (state 2 to 1). Not allowing backward transitions is equivalent to making state 2 absorbing.

Given this formulation of the unconstrained HMM, the three nested models emerge as we constrain either or both model components. To start, when we apply both constraints to all individuals, such that there is an off state ($p_{2i} = 0$) and backward transitions are prohibited ($\theta_{21i} = 0$), the buy till you die BG/BB model emerges.⁴

Then, as we think about how the BG/BB model and HMM differ along these two dimensions, we can consider each of those dimensions separately (i.e., either $p_{2i} = 0$ or $\theta_{21i} = 0$). These constraints determine the two dimensions of Table 1. When applying each of the two constraints separately, different models emerge (the off-diagonal cells of Table 1), and each tells a distinct story of customer behavior.

When only $p_{2i} = 0$, the *on and off* model (OF) emerges. Consumers can make back-and-forth transitions between an “on” state of activity and an “off” state of inactivity. Like the HMM, customers can make backward transitions, yet like the BG/BB model, when in the off state, customers have no chance of activity. This kind of model has been explored in papers on Markov-modulated Poisson processes (Ma and Buschken 2011).

Alternatively, when only $\theta_{21i} = 0$, we get the *hot then cold* model (HC). At any time, customers can be either in a “hot” state (higher propensity to purchase) or a “cold” state (purchasing is less likely but still possible). Like the BG/BB model, once the customer reaches the cold state, she remains there (no backward transitions), and like the HMM, in the cold state, purchasing is possible. The hot-then-cold ordering is informed by the prevalence of customer attrition or, at least, the slowing down of transactions (in aggregate) that is common in most cohort-level

² The initial state membership probability is assumed to be a population-level parameter as a result of the definition of a cohort of customers acquired at the same time. Additionally, the results are robust to using a logit-normal for heterogeneity on all individual parameters and for allowing correlations among them.

³ We use highly uninformative hyperpriors on the population-level parameters of the beta (or Dirichlet) distributions. For more details about the distributions used in the sampling procedure, see Appendix A.

⁴ Unlike the BG/BB as in Fader et al. (2010), which assumes that all individuals start in the “alive” state, in our BG/BB specification we allow individuals to start in either state, according to initial state probability vector π . The model utilized here is more general.

data sets.⁵ Such behavior appears in queuing theory models, such as phase-type distributions (Bladt and Neuts 2003, O’Cinneide 1990), and in marketing models (Fader et al. 2004, Schweidel and Fader 2009).

Past literature has noted how the latent-class model (Kamakura and Russell 1989) is a special case of an HMM ($\theta_{12i} = \theta_{21i} = 0$), and other work often utilizes a nested model with a “death” state (Netzer et al. 2008, Schweidel et al. 2011). However, the other links among the HMM and its constrained models (e.g., BG/BB, HC, OF) that we consider have not been documented in full detail as an integrated framework with the 2×2 structure as described here.

Viewing the HMM and its constrained variants as an integrated family provides an opportunity to detect when (i.e., for which types of data sets) each model component is worth including. One may initially (but erroneously) think that the nested structure would guarantee that the more flexible HMM would perform at least as well as any of its constrained versions (with one or both model components shut off) on all model-performance criteria. But this is not guaranteed in practice. We illustrate that when forecasting repeat transactions out of sample, the more general model does not always beat its nested versions, and hence there is value in the decision tree provided in this paper.

The decision tree answers our key question: For what kinds of database characteristics does each model perform best? To perform this classification, we need a range of different data sets generated from the 2×2 framework. We generate 64 synthetic data sets, each with $T = 30$ weeks of data in calibration (and 30 for holdout) and $N = 500$ customers, with considerable variation by simulating them from unconstrained and constrained versions of the HMM (i.e., to capture each of the submodels as well as the full unconstrained HMM) with a generous range of population-level parameters:

$$\begin{aligned} \mu_{p_1} &\in [0.05, 0.50], & \mu_{p_2} &\in [0.00, 0.10], \\ \mu_{\theta_{12}} &\in [0.10, 0.35], & \mu_{\theta_{21}} &\in [0.00, 0.25], \\ \phi_{p_1}, \phi_{p_2}, \phi_{\theta_1}, \phi_{\theta_2} &\in [0.10, 0.45], & \pi_1 &\in \{0.50, 1.00\}. \end{aligned} \quad (4)$$

We discuss these synthetic data sets and the variability across them in the next section. However, after creating these data sets, we put aside the data-generating process and describe them entirely by easy-to-compute and managerially relevant database characteristics, which we now cover in detail.

⁵ For this reason, we do not consider a separate “cold then hot” model, although the general HMM and OF specifications allow individual-level purchasing to speed up over time.

3. Selecting Database Characteristics

The decision tree is a tool that predicts which specification is likely to be the winning model by only looking at summary statistics of a particular database. Just as we need a set of reasonable models from which to choose, we also need a set of database characteristics to drive the choice process. But which database characteristics should we consider? We illustrate our process of identifying relevant database characteristics by returning to one of the opening examples, repeat purchasing for data set A. Before running any models, analysts frequently examine two typical displays of a cohort’s purchasing behavior: a cross-sectional histogram of customer-level transactions and a longitudinal tracking plot of cohort-level purchases over time. These two graphs appear in Figure 2.

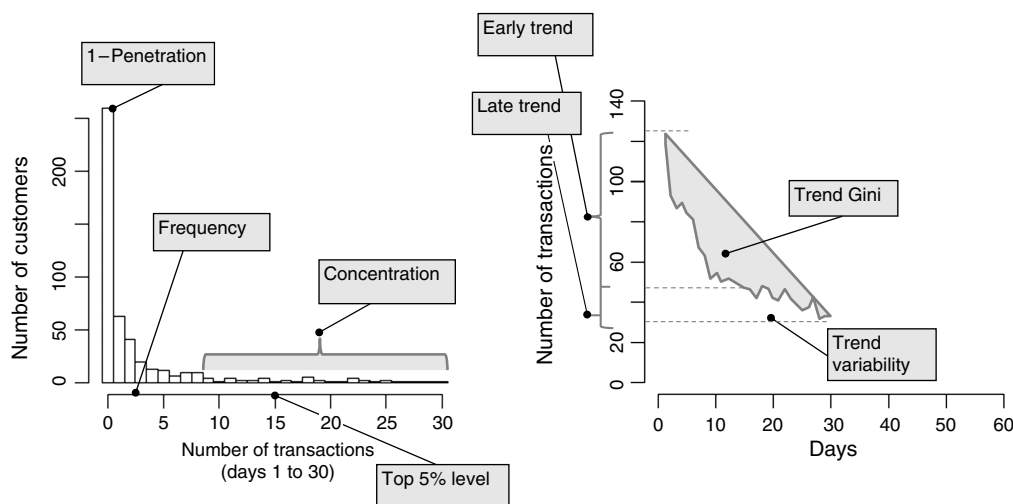
What are the key features of each graph? We want to choose summaries that are both managerially relevant and easy to compute directly from these aggregate plots. We identify four summaries that offer a fairly complete characterization of each plot. For the histogram, we propose summaries to capture the nature of the head and the tail of the distribution as well as its central tendency. For the tracking plot, we focus on the early and late trends in purchasing as the cohort ages and the trend’s overall “bowed” shape, as well as the overall variability over time.

More specifically, Table 2 contains a listing of these measures, which we will use in our subsequent empirical analysis. Although there is not an exact science to selecting these measures, we choose them here to represent central tendency (e.g., average frequency), higher moments (e.g., top percentile, purchase concentration 80:20-type rule), and trend behavior (e.g., steepness, shape, trend variability). We do not claim this list to be comprehensive, but these values vary widely and in systematic ways across the data sets generated by the HMM and its constrained versions.

The variation in these measures across databases is essential: it allows us to explicitly show the range of empirical patterns we consider here and is required to obtain a meaningful classification tree linking these summaries to the model selection process. We illustrate some of this variation in values of these summary statistics for data sets A, B, and C (see Table 3). It is interesting to see how the data sets are indistinguishable on some dimensions (e.g., frequency), quite distinct from each another on others (e.g., penetration), and occasionally exhibit pairwise similarities (e.g., late trend for data sets B and C).

In most empirical settings, we think about the amount of information as being related to the number of observations within a data set. But in this setting, each data set is reduced to a single observation described along multiple dimensions, i.e., the database characteristics described above. Thus, we

Figure 2 The Observed Database Characteristics Arise Naturally from Plots That Managers Typically Examine When Deciding Which Model(s) to Run



Note. The histogram (left) shows how the number of transactions varies across customers in the observation period, and the tracking plot (right) shows incremental transactions of the cohort over the same period.

construct a “data set of data sets,” a collection of 64 simulated data sets reflecting variation along the summary statistics and representing real-world data sets (Fader et al. 2010, Netzer et al. 2008, Schweidel et al. 2011). Specifically, we generate data sets from all possible combinations of the parameter values noted in §2, which allows us to reflect both the structural variation and the “natural randomness” that arises from simulating the purchases. Once the data sets

are created, the true values of the population-level parameters are no longer taken into consideration.

Figure 3 shows the large variability along the values of the database characteristics across the simulated data sets. For instance, nearly half of the data sets have penetration rates between 40% and 70%. About 40% of them have a very steep declining trend (steeper than a drop in transactions equivalent to 15% of the cohort size), whereas others show some growth in purchases for the cohort over time. Thus, we believe that by selecting and creating data sets in this way, we will have avoided biasing our classification results to favor any particular model specification.

To ensure that our chosen characteristics are explaining most of the meaningful variation across the collection data sets, we ran a principal components analysis and an exploratory factor analysis on an even larger set of summary statistics beyond the ones described earlier. We do not present the detailed results but note a few highlights. The principal components analysis indicates that 99% of the measured variation across the 64 data sets can be captured by six independent components. The loadings of the

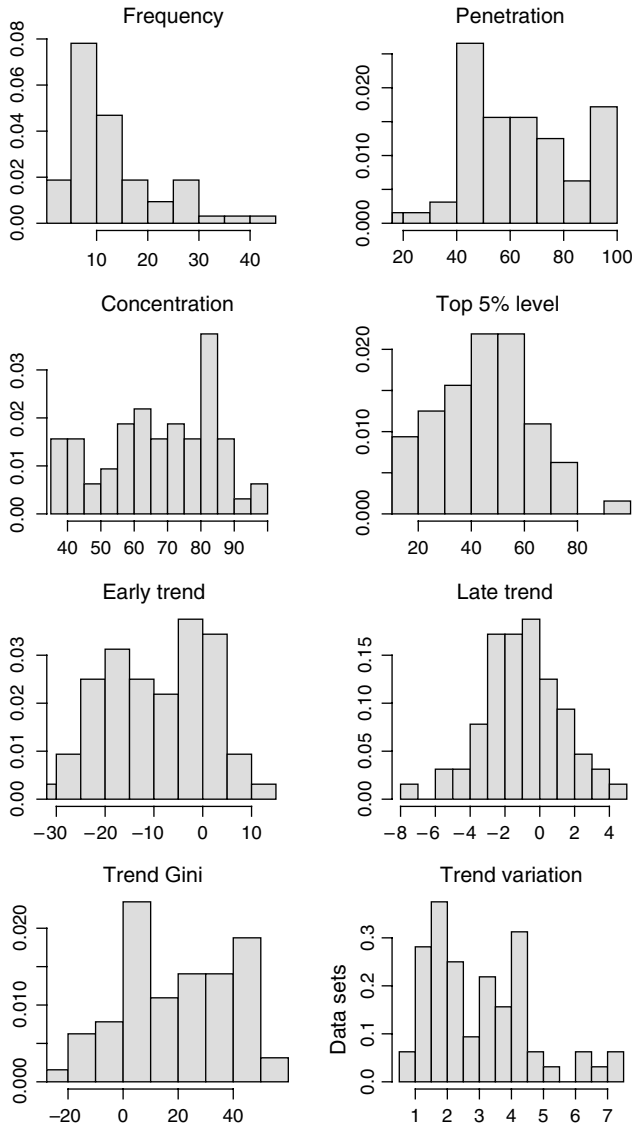
Table 2 Database Characteristics that Capture Features of a Longitudinal Incidence Data Set and Can Be Computed from Summary Plots (e.g., Histogram and Tracking Plot)

Characteristic	Description
Frequency	How many active days of transactions are there per customer?
Penetration	How many unique customers have made at least one transaction?
Concentration	How is activity spread out among customers (i.e., what fraction of all transactions was made by the top 20% of customers)?
Top 5% level	How much are the most active customers purchasing (i.e., what level of transactions is the cutoff for the top 5% of most active customers)?
Early trend	What is the trend in the <i>first half</i> of the calibration period (i.e., drop from first to middle day as a percentage of the size of the customer base)?
Late trend	What is the trend in the <i>second half</i> of the calibration period (i.e., drop from middle to last day as a percentage of the size of the customer base)?
Trend Gini	How much does the actual curve deviate from a line connecting the first and last days (i.e., how much area is there below the trend line and the curve, as percentage of the levels of the line, à la the Gini coefficient)?
Trend variability	How much day-to-day variation is present in the calibration period (i.e., standard deviation of incremental sales)?

Table 3 Database Characteristics for the Three Highlighted Data Sets

Characteristic	%		
	Data set A	Data set B	Data set C
Frequency	11	12	11
Penetration	48	83	68
Concentration	84	55	63
Top 5% level	63	40	43
Early trend	−17	−29	−7
Late trend	−3	1	0
Trend shape	27	46	13
Trend variability	4	6	2

Figure 3 Histograms Summarizing the Variability for Each Database Characteristic Across All 64 Data Sets



principal components analysis and the loadings of the exploratory factor analysis (with five, six, and seven factors) all point to a very similar set of summary statistics, such as central tendency, concentration, and variation over time.

We also recognize that a number of these database characteristics are naturally correlated with each other. Some measures are quite independent (e.g., late trend and penetration, $r = 0.01$), but other pairs have correlations that are large and significant (e.g., average frequency and penetration, $r = 0.86$). Although this kind of multicollinearity could be a serious problem in a typical regression-like model, it does not affect the classification tree and random forest approach since they are nonparametric methods designed specifically for (sequential) variable selection (Breiman 2001a, Breiman et al. 1984).

4. Assessing Model Performance

The final ingredient that goes into the classification tree is a rule for declaring a winning model for a given data set. Here, we select a winner based on each model's ability to predict an important managerial quantity that is widely used for purchasing data because of its link to customer lifetime value and other profit measures: aggregate incremental sales over a holdout period. Specifically, we select an error measure that summarizes the time series of discrepancies between the model and the observed sales for each "Markov chain Monte Carlo (MCMC) world." We will look at the variability of the errors across "worlds" and also average the errors across the worlds to obtain a measure of the model's error for that data set that integrates over the posterior uncertainty. The error measure we use, mean absolute error (MAE), assumes a linear loss function and is frequently used for time-series data. In the more general formulation of this procedure (see Appendix B), one can select any managerial quantity (replacing out-of-sample aggregate sales over time) and error measure with a different loss function (to replace MAE). Although we present results using MAE for our context, our classification tree results are robust to alternative common summary error measures (e.g., mean absolute percent error and root mean squared error).⁶

Formally stated, we quantify performance as the degree to which the model-based posterior predictive distribution of out-of-sample aggregate sales is outlying with respect to the quantity's observed value. We assume the posterior distribution has been obtained using standard MCMC procedures (detailed in Appendix A), yielding posterior draws $g = 1, \dots, G$. For data set k , y_{kt}^{obs} is number of the observed incremental transactions at time period t , and $y_{kmt}^{*(g)}$ is one replicate from model m 's corresponding posterior predictive distribution for that quantity (i.e., incremental transactions). Then for each posterior replicate g , we compute the mean absolute error:

$$d_{km}^{(g)} = \frac{1}{T} \sum_{t=1}^T |y_{kmt}^{*(g)} - y_{kt}^{\text{obs}}|. \quad (5)$$

⁶ We note that our choice of error measure for model selection is in contrast to commonly used likelihood-based summary criteria, such as Bayesian information criterion (BIC) and deviation information criterion (DIC) (Montgomery et al. 2004, Montoya et al. 2010, Netzer et al. 2008, Schweidel et al. 2011). We use an empirical quantity for model selection since many scholars caution against using purely likelihood-based measures (Gelman and Rubin 1995), especially for latent-state models, such as the HMM and its variants, because one must face issues with unstable estimators, computation of the posterior distribution, and correction factors of the log-marginal likelihood (Chib 1995, Lenk 2009, Newton and Raftery 1994, Spiegelhalter et al. 2002).

We will use the values of $d_{km}^{(g)}$ in two ways. On the one hand, we will examine the average posterior MAE for all four models on each data set to determine a single winner per data set. On the other hand, to provide a more nuanced set of findings, we characterize the full posterior uncertainty of the MAE by computing the probability that each model has lowest value (i.e., the proportion of times each model is the winner across the G posterior replicates) because we would not want to overly penalize a model that is a “close second,” for instance. We also use the latter directly in our classification tree.

Now, armed with a set of models (the HMM and its constrained variants), the in-sample database characteristics for each data set (see Table 2), and an error measure (out-of-sample MAE), we have all of the ingredients for the decision tree, which is described next.

5. When to Use Which Model? A Classification Tree

We classify data sets to reveal how we can select the model with the best out-of-sample error by only using in-sample database characteristics. This enables us to answer the paper’s central question: Given a data set’s summary statistics, which model best fits the data?

The winning model, $m_k^{\text{Winner}(g)}$, for data set k and posterior world g is determined by the identifying the model with the minimum error $d_{km}^{(g)}$ among all M models:

$$m_k^{\text{Winner}(g)} = \arg \min_{m=1, \dots, M} d_{km}^{(g)}. \quad (6)$$

We use a classification tree to relate the identity of the winning model, $m_k^{\text{Winner}(g)}$, to the vector of database characteristics, $\tau(\mathbf{Y}_k^{\text{obs}})$. Given the performance of all M models across all K data sets and G posterior replicates, we explain variations in the model performance (i.e., which model wins) as a function of the observed summaries of that data set. Stated formally, we capture this relationship as follows:

$$\hat{m}_k^{\text{Winner}(g)} = \text{Tree}[\tau(\mathbf{Y}_k^{\text{obs}})], \quad (7)$$

where the function “Tree” denotes the classification tree predicting the winning model $\hat{m}_k^{\text{Winner}(g)}$ for each of the data sets $k = 1, \dots, K$ and posterior world $g = 1, \dots, G$.

The classification tree provides cutoff values of the data set-level summary statistics to place entire data sets into “buckets.” This classifies data sets in an easy-to-interpret manner. Each bucket of data sets has a similar profile of data set-level summary statistics

and similar patterns of model performance. Therefore, when a new data set is encountered, it can be classified using this decision rule to identify which of the models will likely be most suited for it. This allows us to uncover relationships between observed patterns in the data and model fit that are easy to interpret while avoiding the need to make any additional assumptions about functional form or error distributions common to ordinary regression models.

Additionally, our classification tree approach goes one step further because it also reflects the natural parameter and model uncertainty. We reflect that uncertainty since our Bayesian modeling approach provides the full posterior distribution of performance for each data set–model pair. As a result, each case to be classified is unique to a particular posterior draw from a model run on a data set. This means that the data to be used to construct the classification tree contain $G = 100$ model-based replicates of the $K = 64$ observed data sets. By using G replicates of each set of observed data set summaries (independent variables), we allow for G different values of errors from each model–data set pair; hence, each data set has a distribution of different winning models (dependent variables) and therefore receives an appropriate number of “votes.”

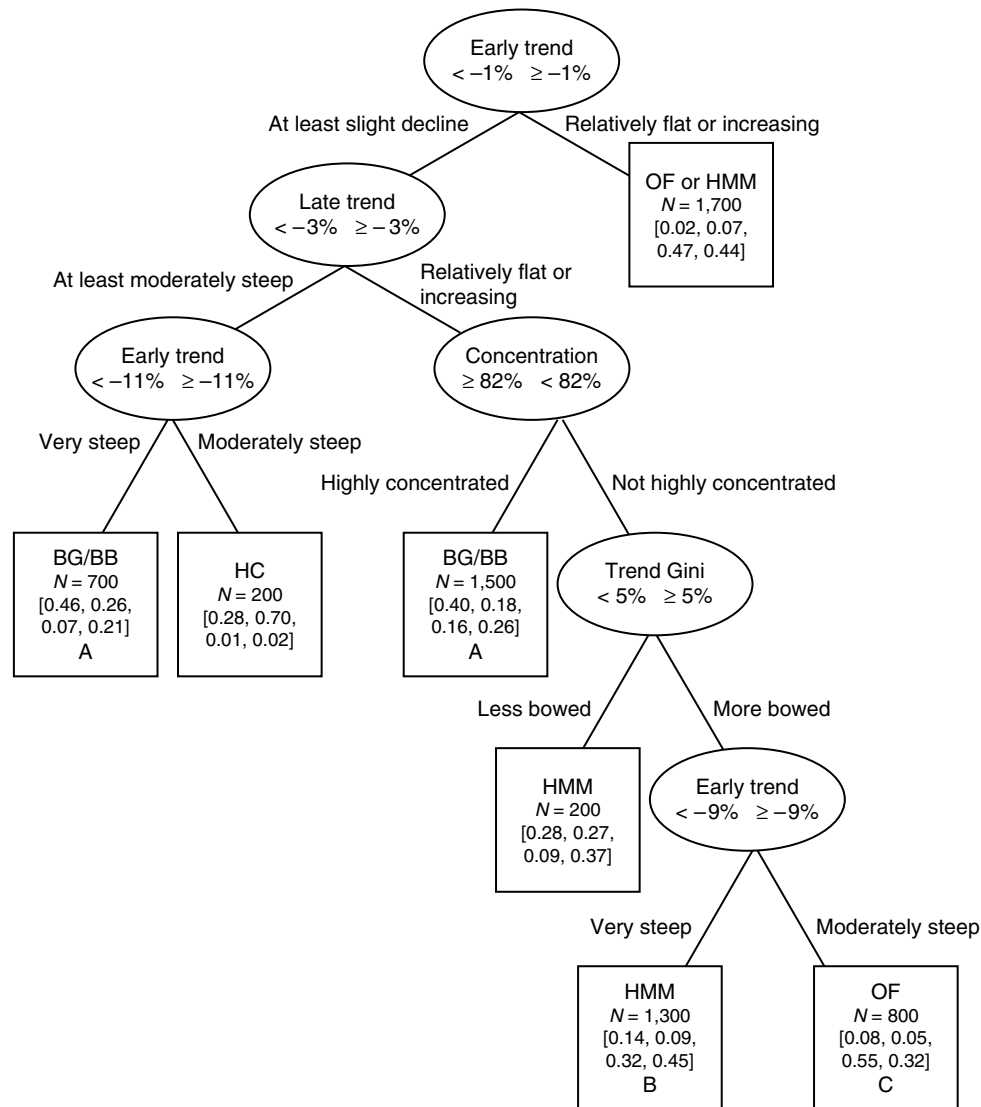
5.1. Classification Tree

The classification tree in Figure 4 can be easily read by starting at the top and following a series of “if...then” decisions down to a terminal node at the bottom of each branch. These terminal nodes represent a group of data sets with the same observed summary statistic branch values (predictor variables). Each node has a recommended winning model but also displays the within-node winning percentages for each model (based on the number of posterior worlds in which each model had the lowest forecast error). Note that the N values in the tree sum to 6,400 cases, reflecting the use of 100 posterior replicates for each of the 64 simulated data sets.

Four database characteristics were selected by the classification tree’s sequential variable selection algorithm as being diagnostic: early trend, late trend, concentration, and trend Gini (trend shape).

The early and late trend statistics reflect the change in transactions over each half of the calibration period (15 days in each half) expressed as a percentage of the total customer base (500 customers). The classification tree partitions early trend into three levels: *very steep* (steeper than a drop in daily transactions equivalent to 11% of the customer base), *moderately steep* (a drop between 1% and 11%), and *relatively flat or positive* (a slope that is more positive than -1%). Late trend is partitioned into two levels, which we label *moderately steep* (a drop steeper than 3% of the total customer

Figure 4 The Classification Tree



Notes. The tree is estimated on 6,400 cases, using 100 posterior samples for each of the 64 data sets. For any data set, the tree should be read from top to bottom: the ovals represent the partitions, the rectangles indicate the terminal nodes, and the listed model is the recommended one for that particular combination of database characteristics. Also listed is a vector summarizing the posterior winning percentage for all four models (left to right: BG/BB, HC, OF, and HMM). The highlighted data sets A, B, and C appear where they are best classified.

base) and *relatively flat or positive* (a slope that is more positive than -3%). Next, the split for concentration has a remarkable resemblance to the 80:20 rule. A data set is either *highly concentrated* (more than 82% of purchases are made by the top 20% of customers) or *not highly concentrated*.

The trend Gini summary statistic reflects the shape of the curve. How much does the actual curve deviate from a line connecting the first and last days of the calibration period (i.e., how much area is there between curve and the trend line)? In other words, this measures the degree to which the curve is “bowed.” The variable is split into a less bowed shape (close to linear with value less than 5%) and a more bowed shape (value greater than 5%). Negative

values indicate that there is more area between the curve and the trend line that sits above the trend line than below the trend line.

We illustrate the use of the tree by returning to our three introductory data sets. Recall their database characteristics were shown in Table 3. We can trace how the tree classifies these data sets to illustrate exactly how a manager can use our decision tree. For instance, data set A exhibits a sales pattern that is downward sloping early on (steeper than -1%) and not strongly downward sloping later on (equal to -3%) and where more than 82% of the purchases are made by the top 20% of customers. Thus, the classification tree recommends that it would be best modeled using the BG/BB model since that model provides

Table 4 For Each Data Set and Each Model's Posterior Draw, the Out-of-Sample Forecast Is Generated from the Posterior Predictive Distribution and MAE Is Computed

Model	Data set		
	A	B	C
BG/BB	4.37	9.71	14.13
HC	4.91	10.16	17.29
OF	6.52	10.20	7.09
HMM	5.13	8.35	9.44

Notes. The posterior mean of the MAE values for each model–data set pair is shown here. The lowest error value (i.e., winning model) for each data set is in bold.

the best out-of-sample forecast for 40% of the posterior replicates associated with the 15 different data sets that have similar values of database characteristics. (And indeed, the BG/BB model does provide the best forecast for data set A, as we show in Tables 4–6.)

It is interesting to note the internal consistency of the tree. In particular, the precise value of the late trend for data set A is exactly the classification tree's cutoff value (–3%). So even if the data set's late trend were just slightly less than that cutoff, the data set would still fall into a node dominated by the BG/BB model (i.e., in the leftmost terminal node of the tree, the BG/BB model is the best-performing model in 46% of posterior replicates).

For data sets with a declining early trend and a flat or increasing late trend, but without a high purchase concentration (fewer than 82% of the purchases made by the top 20% of customers), a different pattern emerges. Data sets B and C are two such examples, so they fall into two terminal nodes in this part of the tree. Data sets represented in this part of the tree show a strong need to allow for back-and-forth transition (HMM and OF). But within the back-and-forth pair, there is less certainty about which one wins.

Further splitting the data sets by trend Gini (trend shape) over the calibration period and by early trend one more time allows the analyst to better discriminate

Table 5 For Each Data Set and Each Model's Posterior Draw, the Out-of-Sample Forecast Is Generated from the Posterior Predictive Distribution

Model	Posterior distribution of MAE (%)								
	Data set A			Data set B			Data set C		
	25	50	75	25	50	75	25	50	75
BG/BB	3.7	4.2	4.8	8.3	9.6	11.0	11.9	14.0	16.2
HC	4.0	4.6	5.5	8.9	10.1	11.4	15.4	17.3	19.1
OF	4.9	6.2	7.8	8.6	10.0	11.6	6.3	7.0	7.8
HMM	4.1	4.7	5.8	7.0	8.1	9.5	7.7	9.0	10.7

Notes. MAE is computed for each replicate data set. The posterior quantiles (25%, 50%, and 75%) of the values across for each data set are shown here. The lowest value (i.e., winning model) for each data set is in bold.

Table 6 Posterior Probabilities of Each Model "Winning" Are Computed as the Proportion of Replicated Data Sets (e.g., "MCMC Worlds") in Which Each Model Has the Lowest MAE

Data set	Posterior probability of model winning (%)			
	BG/BB	HC	OF	HMM
A	45	27	6	22
B	20	11	16	53
C	1	0	81	18

Note. These winning percentages illustrate the uncertainty in declaring a winner.

when each model is likely to perform better. For data sets with a more bowed shape (trend Gini greater than or equal to 5%) and a very steep early trend (steeper than 9%), such as data set B, the HMM wins with 45% of votes versus the OF with 32%. However, for others with a moderately steep early trend (between a 1% and 9% drop) and a more bowed shape, such as data set C, the OF wins with 55% of votes versus the HMM with 32%. Data sets B and C are therefore best classified by the HMM and OF, respectively.

The split on trend shape (trend Gini) and an additional split on early trend should be intuitive because the HMM is a more general model than the OF. As a result, the HMM can generate a wider range of patterns across data sets than the OF can because of the extra model flexibility (e.g., state 2 purchase probability is not necessarily zero). To understand this, keep in mind the patterns common to the data sets in this part of the tree: not highly concentrated purchasing and flat or increasing late trend. On the one hand, for less bow-shaped curves, the OF has difficulty capturing a nearly linear pattern since the off state induces a moderate steep early drop. On the other hand, for markedly bow-shaped curves, the OF also has difficulty capturing both the very steep early declining trends and flat or increasing later trend. Capturing such an interaction among database characteristics is an advantage that CART methods have over traditional linear regression approaches.

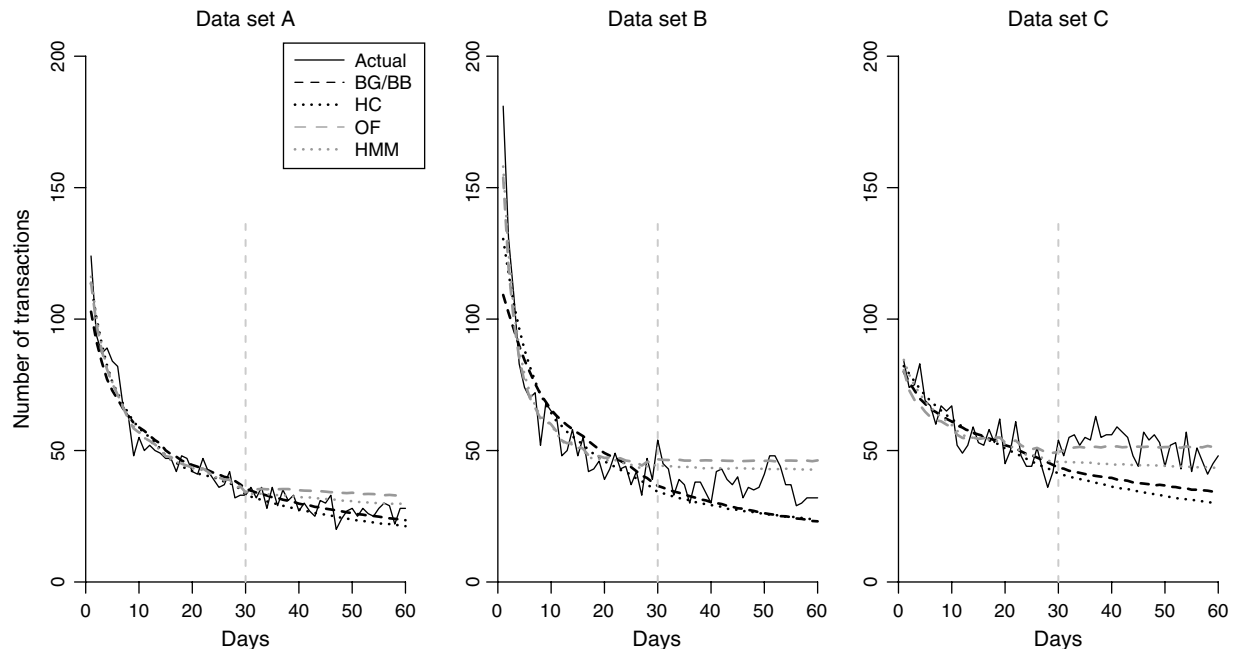
5.2. Uncertainty in Model Performance

As we take a deeper dive into particular branches of the decision tree, we examine the uncertainty in model performance. That is, although the model with the lowest error is declared the winner, we describe how we one can use votes for each winning model by utilizing the full posterior from the Bayesian model output.

As an illustration, we return to data sets A, B, and C to look at the comparative performance of the HMM and its variants from the 2 × 2 framework. The average performance seen in the plots of Figure 5 are quantified in Table 4.

We summarize each model's performance for a data set using MAE averaged over the posterior

Figure 5 Three Example Data Sets with Predictions Arising from Each of the Four Models



Note. The mean of each model's posterior predictive distribution is shown.

uncertainty. For example, for data set A, the BG/BB model has the best average out-of-sample prediction (MAE = 4.37, mean across replicates), closely followed by the HC (4.91). For data set B, the HMM (8.35) clearly outperforms the other three models, and data set C is best modeled by the OF (7.09).

Whereas those are posterior means of model performance, we also convey the degree of uncertainty in these assessments using replicated data sets associated with the full posterior predictive distribution. To illustrate this uncertainty, we plot the predicted incremental sales for each posterior replicate for data set A and the observed daily incremental sales (see Figure 6). By visual inspection of these tracking plots alone, it is difficult to detect whether the BG/BB model truly predicts better than the other three models.

Although we would like to declare a single winning model for each data set, the high level of uncertainty around the model predictions seems to raise a warning flag about making any strong statements about differences among the models. Therefore, we want to quantify the “shades of gray” in model performance by recognizing that when declaring a winning model, the vote need not be unanimous.

Thus, instead of only examining posterior mean of MAE, we characterize its full distribution. For the highlighted data sets A, B, and C, we show the distribution of each model's MAE across all replicates (see Figure 7). Table 5 displays the corresponding distribution summaries (e.g., median and interquartile ranges of MAE across replicates).

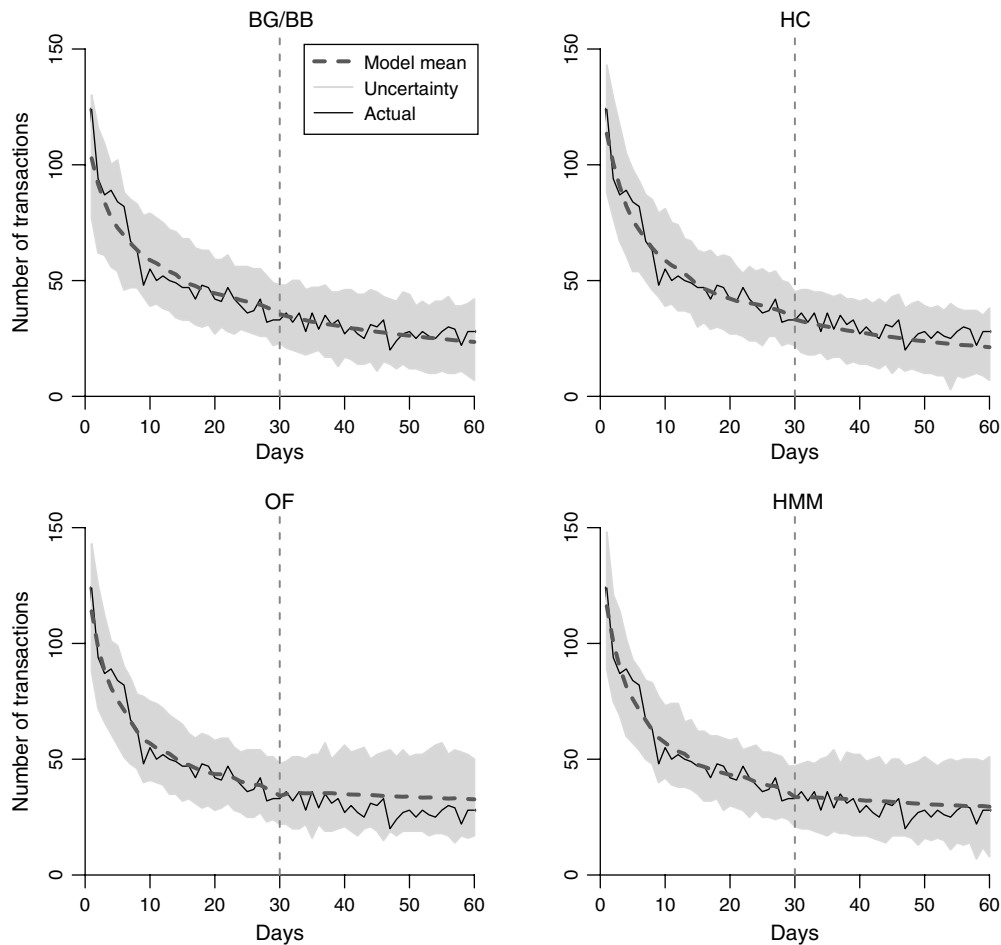
Not surprisingly, the densities of the performance measure of the four models are somewhat overlapping. For example, in data set A, although most of the mass of the BG/BB model density is lower (better) than that of the HMM and HC densities, there is some probability that the HMM or HC has a lower MAE than the BG/BB model. This suggests there is not a unanimous winner. By contrast, in data set C, for instance, there is much less overlap, suggesting that the OF has an even higher chance of a lower MAE than do the others.

But what is the probability that each model is the winning model for a data set? We take advantage of the Bayesian output to make this probability statement. Table 6 shows each model's winning percentage for data sets A, B, and C. That winning percentage, or percentage of votes, is the proportion of posterior worlds in which each model has the lowest error. For instance, the OF is quite clearly the winner for data set C since it wins 81% of the time. For data set A, although the BG/BB model is the winner, the distributions of the error for three of the four models overlap. So it is not surprising that they split the votes, and the BG/BB model wins 45% of the time compared with 27%, 6%, and 22% for HC, OF, and HMM, respectively.

5.3. Assessing the Predictive Value of the Decision Tree: In Sample

How accurate are the resulting recommendations from the tree? We answer this question to assess the tree's predictive value. First, we focus on the simple

Figure 6 Illustrations of the Range of Variability in Model Prediction for Data Set A

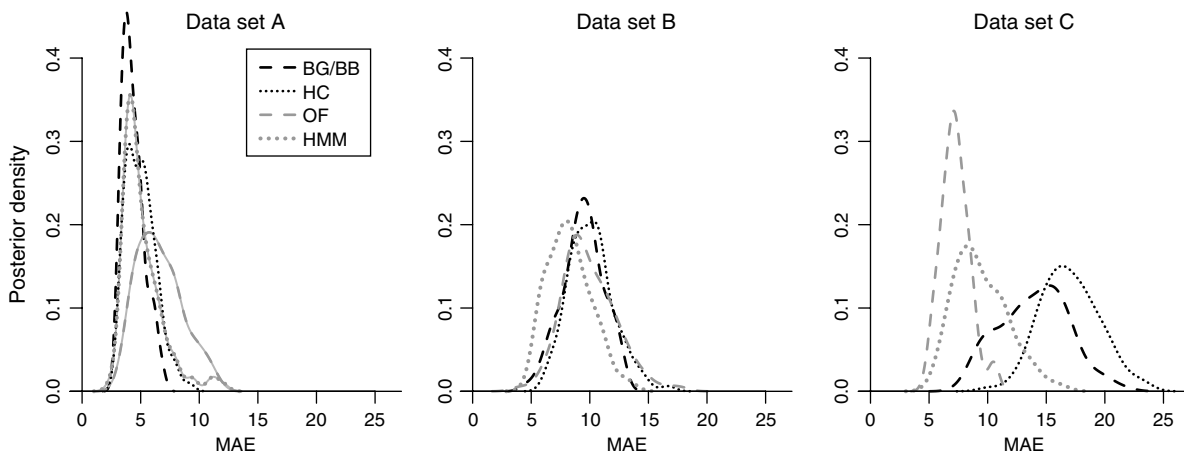


Notes. The observed data (solid line) is fit by each model. Each model's posterior predictive mean (dashed line) and its full distribution from 1,000 posterior draws (light gray shading) are displayed.

measure of the hit rate of the classification tree. The hit rate is the number of times the tree recommends a model that is, in fact, the best model to use on that

data set. Averaged across all models and iterations, the hit rate is 46%. We put this hit rate in context by noting that from a purely operational standpoint,

Figure 7 Posterior Distribution of Out-of-Sample MAE for the Three Highlighted Data Sets



Notes. These are densities of error measures, so smaller values indicate better model fit. When the densities overlap a great deal (e.g., data set A), there is not a clear winning model. When one density that stands out as better than the others (e.g., data set C), there is a clear winning model.

the tree allows the analyst to run one model instead of four. In other words, by reducing the work of an analyst by 75%, the tree makes a recommendation of which model to use that is about twice as good as guessing (a 25% hit rate) randomly among the four models. This hit rate also fares well when compared with tougher comparative yardsticks, such as the proportional chance criterion and maximum chance criterion (Morrison 1969), which yield benchmark hit rates of 28% and 35%, respectively. The latter metric is often hard to beat in a discriminant analysis setting. It assesses how much better our classifications are compared with using the most common actual winner (in this case, the HMM) every time. Thus the decision tree clearly offers some improvements over that simple (but often effective) approach.

However, this measure is purely an in-sample one: it uses the same 64 data sets for calibration and classification purposes, so it may be subject to overfitting. We next describe a procedure, random forests, that will allow us to reflect the uncertain nature of the tree itself and its application to holdout data.

5.4. Assessing the Predictive Value of the Tree Using Random Forests: Out of Sample

To answer the question about the value of the tree, using another lens, we turn to another machine-learning method closely related to CART known as *random forests* (Breiman 2001a, Liaw and Wiener 2002). Although the single classification tree we have described above takes into account the parameter and model uncertainty, it does not take into account uncertainty in the structure of the classification tree itself. The random forest captures extra variation around the classification. This requires many classification trees, so the random forest algorithm “grows” many trees (hence, the “forest”).

What is special about the random forest is that it has a built-in monitoring system to make sure it produces predictions that are validated on a holdout set and that are utilizing important predictor variables. Both aspects of predictions prevent overfitting (Breiman 2001a, Liaw and Wiener 2002). Fortunately, the random forest algorithm has a built-in cross-validation procedure calculating an n -fold cross-validation, where the holdout sample size, n , is typically about 1/3 of the cases (Breiman 2001a). The holdout misclassification rate, in the language of machine learning, is called the “out-of-bag” error rate, or the “generalized error rate” since it is intuitively similar to cross-validation error, which indicates the ability of the predictive model to generalize to cases outside of the given data set.

The random forest out-of-sample error rates broken down by each model are in Table 7, and the hit rate across all four models is 48%. This closely matches the in-sample hit rate using one classification tree. It is

Table 7 Out-of-Sample Classification for Each of the 6,400 Cases (Data Set–World Pairs) from the Random Forest

Model	BG/BB	HC	HMM	OF	Hit rate (%)
BG/BB	840	151	215	114	64
HC	372	247	145	154	27
HMM	430	40	1,056	677	48
OF	259	7	742	951	49

Notes. Rows indicate which model actually fits the data best. Columns indicate which model was recommended by the random forest’s classification for that data set using a 2/3 sample for calibration (in sample) and 1/3 sample for validation (out of sample). The hit rate is the proportion of each type of data set correctly classified as out of sample (i.e., the diagonal entries divided by row sums).

encouraging to see that, even when a data set is not used for calibration, it can be classified correctly with a high level of accuracy.

Looking more carefully at the classification tree and random forest results, several distinctive patterns arise. It is clear that the BG/BB and HMM models have substantially higher hit rates than do the other two models. It also seems that each of these polar opposite models (at least in terms of parameters and complexity) can serve as effective “representatives” to characterize the entire family of HMM models covered here.

This result raises the question about which of the two constraints/dimensions associated with our 2×2 framework is more important to capture: the presence of an off state or the existence of an absorbing state. A closer inspection of Table 7 clearly reveals the answer: classifying whether or not the data require an absorbing-state model or a back-and-forth model is much more informative than the presence of an off state. There is a high degree of confusion between the BG/BB model and HC, and likewise for HMM and OF, but relatively little confusion between the BG/BB model and OF or between HMM and HC. In Table 8, we aggregate the classifications across this single dimension and see incredibly high hit rates (62% and 89%) when we ignore the presence or absence of the off state.

We have explored the predictive value of the decision tree, so it is natural to ask what is driving its good predictive ability. To better understand the drivers of

Table 8 Combined Cases of Data Sets and Classifications Into Models with Absorbing States (BG/BB and HC) and Back-and-Forth Transitions (OF and HMM)

State	Absorbing	Back-and-forth	Hit rate (%)
Absorbing	1,383	855	62
Back-and-forth	466	3,696	89

Notes. That is, by ignoring the presence or absence of an off/death state, the hit rates are quite high. Like Table 7, these are out-of-sample classifications, so the hit rate is the proportion of cases correctly classified.

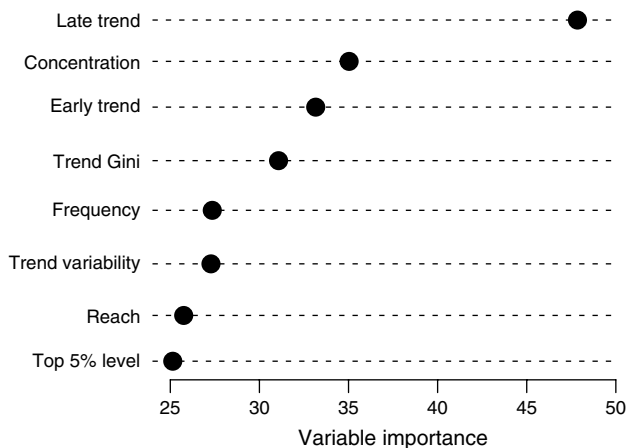
our strong classification capabilities, we now analyze the database characteristics' diagnostic value.

5.5. Which Database Characteristics Are Most Diagnostic?

The output of the random forests uncovers which variables are most important in explaining classification success. This not only validates the decision tree obtained via CART but also quantifies variable importance. Variable importance in random forests is a measure of the average improvement in prediction accuracy of a tree when this variable is included (and its values are intact) compared with when this variable's values are meaningless (arbitrarily permuted across observations).

Figure 8 displays each database characteristic's variable importance. This analysis confirms what we see in the classification tree: early trend, late trend, trend Gini (trend shape), and concentration are the four most important variables, and they are clearly separated out from the others. Among the four, however, late trend is the most important. This makes intuitive sense because the models differ in their ability to generate decreasing or increasing patterns in aggregate sales over time. For instance, for a data set with a strongly increasing late trend, the BG/BB model and HC, because of their absorbing state, would not be able to capture it at all. This provides more evidence that even before running any models, an analyst could use these easy-to-compute database characteristics to refine the decision about which model is likely to perform best.

Figure 8 Relative Importance of Each Database Characteristic (Predictor Variable) Used in the Classification Trees Obtained from the Random Forest



Notes. The most important variables are those that provide the largest increase in out-of-sample (out-of-bag) classification hit rate, averaged across all trees in the forest. The late trend, early trend, and concentration are clearly the three most important and confirmed by appearing in the classification tree obtained via CART methods. The next most important variable is trend Gini (trend shape), which also appears in the classification tree.

5.6. How Much Value Does the Classification Tree Add?

What does the analyst gain by using our decision tree? From the above discussion, we find the decision tree nearly doubles the hit rate compared with uninformed guessing about which of the four models to run. And much of the remaining error rate is associated with the relatively unimportant distinction between the presence or absence of an off state.

But although this information helps ease the task of choosing the right model, it also tells us how well an analyst would do using the decision tree compared with running all four models for every data set. So it is reasonable to ask: How much error would the analyst suffer by using only one model for all data sets? After all, this is the starting point for many analysts. Suppose the analyst only used BG/BB models for all data sets she encountered. How poorly would she have performed? We can compare the error incurred to the average performance if she always used the truly winning model for each data set. Table 9 summarizes this analysis.

Running the BG/BB model on all data sets yields an error 52% worse than using the true winning model (average MAE = 9.52 versus 6.28). An analyst would do better by running only the HMM, which yields an error 21% worse than the using winning model (average MAE = 7.63). By contrast, using the model recommended by decision tree for each data set is the best option because it greatly reduces error to only 12% worse than the winning model (average MAE = 7.05). That is, in terms of relative error to the best model, only using the HMM is 75% worse than using the decision tree. So using the decision tree is a win-win: it requires 75% less effort and helps the analyst to avoid a 75% increase in relative error.

The value of the decision tree is even greater if we look beyond average performance and consider the worst case scenario of model performance. When examining the variability in performance, the 95% level of error for using any single model can be quite

Table 9 Absolute and Relative Benefits of Using the Classification Tree over Running Each Model for All Data Sets

	BG/BB	HC	OF	HMM	Tree	Winner
Mean MAE	9.52	9.19	8.14	7.63	7.05	6.28
% worse than winner	52%	46%	30%	21%	12%	—
95% MAE	22.34	18.97	16.70	15.63	13.00	11.57
% worse than winner	93%	64%	44%	35%	12%	—

Notes. The MAE values reflect performance of running each model for all data sets compared with following the tree's recommendation ("Tree") and always selecting the model with best out-of-sample fit for each data set ("Winner"). The percentages illustrate the loss compared with the best-fitting model (Winner).

high. However, the decision tree greatly controls that upper tail of error. In particular, the high end of possible error for the HMM is 35% worse than the winner's error, but the decision tree is only 12% (see Table 9).

In short, our tree shows that an analyst should not use the same model for all occasions, and it clearly quantifies the cost of doing so.

6. General Discussion and Future Directions

When researchers and managers regularly encounter a particular kind of data structure and regularly choose among a standard set of models, they often develop good intuition about when to use which model. Our approach rigorously quantifies and validates this kind of intuition through a well-structured decision tree.

For the case of a database of repeat purchases over time for a cohort of customers, we make specific recommendations about when to use the HMM and its constrained variants and which data set-level summaries are important for that decision. We find that for data sets exhibiting an early decreasing trend in aggregate sales, the BG/BB model provides the best forecast when the trend continues to decrease even later in the calibration period. But when it looks like the trend has leveled off, the BG/BB model frequently underpredicts, and more complexity is often warranted. An interesting exception to this rule is the case of high purchase concentration, which suggests that the buy till you die framework is still likely to provide the best forecast. This may be reflective of the customers exhibiting high heterogeneity in purchase and churn rates rather than a more complex back-and-forth state-switching process over time.

In the case of the 2×2 framework, the models are classified with strong evidence along one dimension (the presence of an absorbing state versus back-and-forth movement across states), but the data offer weaker evidence to help discriminate data sets and models along the other dimension (the presence or absence of an off state). This may be surprising in light of many papers that add a death state to an HMM-like model. But it may be the case that such models work well mainly because of the constraint making that state absorbing and not necessarily because the behavior is "turned off" within it. This finding could have important implications for model builders and should be investigated more carefully in settings beyond this framework.

Beyond our HMM-based example, our proposed approach for empirical identification is more broadly relevant. We explicitly test the characteristics of data sets that distinguish one model from a related one. Although this differs from a formal theoretical identification (e.g., using economic principles), it is aligned

with the calls for such activities that have been arising more frequently in marketing (Hartmann et al. 2008).

The procedure that we propose is quite general: given the appropriate inputs (i.e., database characteristics), it can generate a decision tree prescribing which model should be used for any given data set and any given outcome/goal of interest. Understanding the interplay between database characteristics and the relative performance of models (and model components) is a useful contribution beyond the illustrative (yet common) context presented here. Although we illustrate it here with the HMM on forecasting incidence data (e.g., repeat purchasing of a cohort), it is agnostic to these choices. In general, the recipe for this method requires the following elements: (1) a consideration set of candidate models, (2) a set of predictor variables consisting of observed summary statistics from each data set, and (3) the outcome variable, which is a choice of how to "pick the winner"; this requires a key managerial quantity and a loss function for computing the error measure.

Classification and regression trees and random forests, although popular in machine learning and statistics, are still relatively new to the field of marketing, so we hope our work will call more attention to this powerful and versatile tool. Furthermore, our application of it to the problem of model selection (as opposed to variable selection) is relatively uncommon even in the statistics literature, but it is clearly a natural and important issue in many marketing contexts.

Unlike traditional uses of classification methods, we add an extra twist by employing them in a fully Bayesian framework, allowing us to leverage the full posterior distribution. This differs from previous mixtures of Bayesian and classification methods, e.g., Bayesian CART (Chipman et al. 1998), since we construct a decision tree from information that already incorporates a joint posterior distribution. Our mix of Bayesian approaches with classification methods is a promising area of research for the interface of marketing, statistics, and machine learning. The combination of the two approaches represents an exciting blurring of methodological boundaries, and marketing problems such as the one examined here have a great deal to offer in the debate between the "two cultures" of data modeling (statistics) and algorithmic modeling (machine learning) put forward by Breiman (2001b).

As computational costs decrease and access to grid/cloud computing increases, the procedure we propose here will be even easier to do in a variety of contexts. Of course, one could argue that with greater computing power, there is less need to worry about selecting the single best model a priori—just run a bunch of models and pick the best one. But this logic is flawed for several reasons. First, our analysis focuses on performance in a holdout period, not in-sample fit. Second, and related, there is great

danger in choosing models that are overly complex and excessively customized to every different data set. And third, we believe strongly in exploring and learning from the underlying patterns that are driving the observed data patterns. This kind of “data science” not only will help analysts create and choose better models but also will help managers make better tactical decisions to create and extract more value from their customer relationships.

Acknowledgments

This research was supported by a grant from Amazon Web Services, which funded the use of the Elastic Computing Cloud. The first author thanks the Wharton Risk Management and Decision Processes Center for its support through the Russell Ackoff Doctoral Student Fellowships, as well as Eva Ascarza, Michael Braun, Oded Netzer, and Kenneth Shirley for their useful comments. The authors all thank Yao Zhang, who contributed to a related working paper, “Children of the HMM: Tracking Longitudinal Behavior at Hulu.com,” which was presented at the 2010 Marketing Science Conference.

Appendix A. Hierarchical Bayes Sampler Details

We provide the computational details for the models that we ran. We provide the details of the sampler for the general HMM with S states. It can be constrained for the two-state HMM and each of its nested models, as described in §2.

The MCMC procedure generates draws from the joint posterior:

$$\begin{aligned}
 & [Z_{it}, \mathbf{p}_i, \Theta_i, \mathbf{a}_p, \mathbf{b}_p, \alpha, \pi | \mathbf{Y}] \\
 &= \prod_{i=1}^I \prod_{t=1}^T [Y_{it} | Z_{it}, p_i][Z_{it} | Z_{i,t-1}, \Theta_i, \pi] \\
 & \cdot [\mathbf{p}_i | \mathbf{a}_p, \mathbf{b}_p, \mathbf{Z}_i, \mathbf{Y}_i][\Theta_i | \alpha, \mathbf{Z}_i][\mathbf{a}_p, \mathbf{b}_p][\alpha][\pi] \quad (\text{A1})
 \end{aligned}$$

with constants (I individuals and T time periods), individual-level parameters (\mathbf{p}_i and Θ_i), and population-level parameters ($\mathbf{a}_p, \mathbf{b}_p, \alpha$, and π).

The procedure obtains these draws by alternating between the following conditional distributions:

$$\begin{aligned}
 & [Z_i | \mathbf{Y}_i, \Theta_i, \mathbf{p}_i, \pi], \\
 & [\mathbf{p}_i | \mathbf{Y}_i, \mathbf{Z}_i, \mathbf{a}_p, \mathbf{b}_p], \\
 & [\Theta_i | \mathbf{Z}_i, \alpha], \\
 & [\mathbf{a}_p, \mathbf{b}_p | \mathbf{p}], \\
 & [\alpha | \Theta], \\
 & [\pi | \mathbf{Z}].
 \end{aligned} \quad (\text{A2})$$

For each entry in the “data set of data sets” described in §3, we estimate all four models from the 2×2 framework. We use 64 data sets, 4 models per data set, 2 chains per model, resulting in 512 independent MCMC chains. We run each chain for at least 50,000 iterations depending on the convergence criterion for that given model and chain. This requires more than 1,000 days of computing time on a single core. Instead of using only one core, we distributed the computational task to take advantage of the parallel structure of the task. On Amazon’s Elastic Computing Cloud, we used 64 nodes with

eight cores per node for 48 hours (24,576 “core-hours”). We ran each MCMC chain on each of the 512 cores, so we finished running all the models in two days. This was funded by a grant from Amazon Web Services.

Each model is estimated using a version of the MCMC sampler for HMM with certain components shut off or not. The code is available from the authors upon request. For each chain and for each pair of chains for each model, we perform a set of within-chain diagnostics for convergence and computation of effective sample size, as well as across-chain diagnostics for post-convergence mixing—all recommended now as standard practice (Gelman and Rubin 1992, Gelman et al. 2004, Geweke 1992, Plummer et al. 2006, Raftery and Lewis 1992).

The draws of model parameters, Ω , and latent states, \mathbf{Z}^* , have been obtained using a data-augmented Gibbs sampler (Tanner and Wong 1987) with an embedded Metropolis-Hastings step. Below we describe how each subset of parameters was drawn from its corresponding conditional distribution in the MCMC procedures.

Step 1. Generate $\mathbf{Z}_i = (Z_{i1}, \dots, Z_{iT})$. The customer’s latent-state sequence is drawn via the forward-backward algorithm. The latent states are sampled starting at $t = T$ moving backward based on the probabilities defined recursively starting at $t = 1$ and moving forward using dynamic programming. For the case of $S = 2$, given the observed outcome at t and the probability of being in either state at $t - 1$, each element $\delta_{i,t,k}$ is a sum of the two elements from $t - 1$ weighted by the probability of the corresponding transition probabilities. Then the probability of drawing state k is

$$\begin{aligned}
 & [Z_{it} | \mathbf{Y}_i, \Theta_i, \mathbf{p}_i, \pi] = \frac{\delta_{i,t,k}}{\delta_{i,t,1} + \dots + \delta_{i,t,S}}, \\
 & \delta_{i,t,1} = p_{1i}^{y_{it}} (1 - p_{1i})^{1 - y_{it}} (\delta_{i,t-1,1} \cdot \theta_{11i} + \dots + \delta_{i,t-1,S} \cdot \theta_{S1i}), \quad (\text{A3}) \\
 & \dots \\
 & \delta_{i,t,S} = p_{Si}^{y_{it}} (1 - p_{Si})^{1 - y_{it}} (\delta_{i,t-1,1} \cdot \theta_{1Si} + \dots + \delta_{i,t-1,S} \cdot \theta_{SSi}).
 \end{aligned}$$

Once the sequences from $1, \dots, T$ are drawn for all individuals, then conditioning on those sampled latent states as if they were data (i.e., data augmentation) simplifies the subsequent conditional distributions. Hence, we define a vector, \mathbf{N}_i , where each element counts the number of times an individual spent a day in each latent state, $N_{ij} = \sum_{t=1}^T 1\{Z_{i,t-1} = j\}$. We also define a matrix, \mathbf{N}_{it} , where each entry j, k counts the number of transitions made between each pair of latent states, $N_{ijk} = \sum_{t=1}^T 1\{Z_{i,t-1} = j\}1\{Z_{it} = k\}$.

Step 2. Generate $\mathbf{p}_i = (p_{1i}, \dots, p_{Si})$. The customer’s purchase probability vector is sampled directly from a beta distribution. The use of independent beta priors for each probability yields a beta posterior distribution (since the likelihoods have no covariates). For state k , the prior and posterior are

$$\begin{aligned}
 & [p_{ki}] = \text{beta}(a_{pk}, b_{pk}), \\
 & [p_{ki} | \mathbf{Y}_i, \mathbf{Z}_i, \mu_p, \phi_p] \\
 &= \text{beta}\left(a_{pk} + \sum_{t=1}^T y_{it} 1\{Z_{it} = k\}, b_{pk} + N_{ij} - \sum_{t=1}^T y_{it} 1\{Z_{it} = k\}\right), \quad (\text{A4})
 \end{aligned}$$

where a_{pk} and b_{pk} are the shape parameters of the beta distribution and μ_{pk} and ϕ_{pk} are the mean and polarization index, as defined in §2. To ensure $p_{1i} \geq p_{2i}$, we use rejection sampling of the whole vector.

Step 3. Generate Θ_j . The customer’s transition probability matrix must have its rows sum to 1, so it is a multinomial vector. Using independent Dirichlet priors on each row yields Dirichlet posteriors. For the probability of moving from j to any other state $1, \dots, S$, the prior and posterior are

$$[\theta_{j,1:S,i}] = \text{Dirichlet}(\alpha_{j,1:S}),$$

$$[\theta_{j,1:S,i} | \mathbf{Z}_i, \boldsymbol{\mu}_{\theta_j}, \phi_{\theta_j}] = \text{Dirichlet}(\alpha_{j,1} + N_{ij,1}, \dots, \alpha_{j,S} + N_{ij,S}),$$
(A5)

where the 1:S indexes a vector of parameters and where \mathbf{r}_j is the vector of Dirichlet shape parameters, which can be summarized by mean probability vector $\boldsymbol{\mu}_{\theta_j} = \boldsymbol{\alpha}_j / [\sum_{k=1}^S \alpha_{jk}]$ and polarization index $\phi_{\theta_j} = 1 / [1 + \sum_{k=1}^S \alpha_{jk}]$.

Step 4. Generate π . The initial latent-state membership probability vector depends on the latent states across all individuals at $t = 1$. Defining $N_{1k} = \sum_{i=1}^I 1\{Z_{k1} = k\}$, the uniform hyperprior and the posterior are

$$[\pi] = \text{Dirichlet}(1, \dots, 1),$$

$$[\pi | \mathbf{Z}] = \text{Dirichlet}(1 + N_{11}, \dots, 1 + N_{1S}).$$
(A6)

Step 5. Generate (a_p, b_p) . There is a highly uninformative hyperprior for each shape parameter of each beta distribution characterizing the heterogeneity of state-specific purchase propensities. For state k , the hyperprior and posterior are

$$[a_{pk}, b_{pk}] \propto (a_{pk} + b_{pk})^{-5/2},$$

$$[a_{pk}, b_{pk} | p_k] \propto L_{\text{beta}}(p_{k1}, \dots, p_{kI})(a_{pk} + b_{pk})^{-5/2},$$
(A7)

where L_{beta} is the beta density function, and the prior distribution proportional to $(a + b)^{-5/2}$ is recommended by Gelman et al. (2004). That prior is uniform on the beta distribution $a/(a + b)$ and considered weakly informative on the polarization index $(1 + a + b)^{-1}$. Since the posterior has no closed-form expression, we use a Metropolis-Hastings step with a log Normal proposal density. Its tuning parameter, or variance, is set to 0.05 to obtain an appropriate acceptance probability.

Step 6. Generate $\boldsymbol{\alpha}$. The Dirichlet distribution shape parameters $\boldsymbol{\alpha}_j = (\alpha_{j1}, \dots, \alpha_{jS})$ are generated by a generalization of the procedure used to generate the shape parameters of the beta distributions. For state j (i.e., row j of the transition probability matrix), the hyperprior and posterior are

$$[\boldsymbol{\alpha}_j] = \left(\sum_{k=1}^S \alpha_{jk} \right)^{-(2S+1)/2},$$
(A8)

$$[\boldsymbol{\alpha}_j | \theta_{j,1:S}] \propto L_{\text{Dirichlet}}(\theta_{j,1:S,1}, \dots, \theta_{j,1:S,I}) L_{\text{gamma}}(\boldsymbol{\alpha}_j),$$

where $L_{\text{Dirichlet}}$ is the Dirichlet density function. Again, the prior distribution proportional to $(\alpha_{j1} + \dots + \alpha_{jS})^{-(2S+1)/2}$ is a generalization for the Dirichlet shape parameters of the prior used for the beta shape parameters (Everson and Bradlow 2002). Since the posterior has no closed-form expression, we use a Metropolis-Hastings step with a log Normal proposal density. Its tuning parameter, or variance, is set to 0.05 to obtain an appropriate acceptance probability.

Appendix B. General Recipe: Developing a Decision Tree for Model Selection Using Database Characteristics

Although we perform our analysis for a specific data/modeling context, the same basic recipe developed in this

paper can be applied to many other settings. We formalize this recipe as a general method for model evaluation and selection involving the three basic ingredients: the set of candidate models, the database characteristics, and the performance criterion. This enables the analyst to answer the questions “Which model should I use for this this data set?” and “Given a data set, how well will a given model perform?”

Step 1. Selecting models. Our procedure supposes that the analyst has a consideration set of models $1, \dots, M$. The models can all be run on data sets with the same structure. We also suppose that model-based simulation can be done via Monte Carlo, or Markov chain Monte Carlo, if needed. Our procedure assumes that an MCMC sampler has been run to obtain draws $g = 1, \dots, G$ from each model’s the joint posterior distribution, $[\boldsymbol{\Omega}_m | \mathbf{Y}^{\text{obs}}]$.

Step 2. Choosing database characteristics. We characterize the database with a set of summary statistics. These should be (1) easy-to-compute characteristics, (2) managerially relevant, and (3) largely comprehensive and mutually exclusive. Formally stated, we denote these the data set-level summary statistics as a covariate vector, $\boldsymbol{\tau}(\mathbf{Y}_k^{\text{obs}})$, for data set k . These are to be computed before running any models on the k th data set, which itself is denoted by $\mathbf{Y}_k^{\text{obs}}$. These are the independent variables of interest in the eventual classification.

Step 3. Determining performance criterion. Assessing model performance for model selection is an important step that should be driven by the business goal. We use an empirical validation approach via posterior predictive distributions. We generate data, $\mathbf{Y}_m^{*(g)}$, from the model-based predictive distribution, $[\mathbf{Y}_m^* | \mathbf{Y}^{\text{obs}}]$, where g indexes replications $1, \dots, G$.

For the performance measure feature s , we summarize the generated data by $T_s(\mathbf{Y}_m^{*(g)})$. We quantify performance as model errors: the degree to which the model-based posterior predictive distribution of feature s is outlying with respect to the feature’s observed value, $T_s(\mathbf{Y}_k^{\text{obs}})$.

Let D denote a loss function along a single dimension—that is, the distance between the draws from the posterior predictive distribution and the single observed value of feature of a data set. This distance, d_{mks} , summarizes model m ’s performance on data set k in terms of feature s , utilizing all replicates $g = 1, \dots, G$:

$$d_{mks} = D(T_s(\mathbf{Y}_{mk}^{*(1)}), \dots, T_s(\mathbf{Y}_{mk}^{*(G)}); T_s(\mathbf{Y}_k^{\text{obs}})).$$
(B1)

The choice of the function D should depend on the desired feature.

Regardless of which performance metric and error measure is chosen, a single metric is obtained for each posterior replicate. For each replicate, we select the model with the lowest error and consider it the winning model, which is the nominal categorical outcome variable to be classified.

Step 4. Classifying data sets by relating model performance to observed database characteristics. Putting those three ingredients together, we create the decision tree to infer the relationship between which model is best (outcome) and database characteristics (predictors). We formalize the classification as its own predictive tool. The independent variables of interest are the data set-level summary statistics, $\boldsymbol{\tau}(\mathbf{Y}_k^{\text{obs}})$, computed before running any models on

the k th data set, denoted by $\mathbf{Y}_k^{\text{obs}}$. These values are the predictors of model performance. The error measure, $d_{ksm}^{(g)}$, as defined above, is on a continuous scale. For classification purposes, however, the dependent variable should be an indicator of the winning model $m_{ks}^{\text{Winner}(g)}$, a nominal categorical variable with M levels. For each data set k , feature s , and posterior replicate g ,

$$m_{ks}^{\text{Winner}(g)} = \arg \min_{m=\{1,\dots,M\}} d_{ksm}^{(g)} \quad (\text{B2})$$

We use a classification tree to relate the identity of the winning model, m_k^{Winner} , to the database characteristics, $\tau(\mathbf{Y}_k^{\text{obs}})$. Given the performance of all M models across all K data sets for feature s , we explain variations in the model performance (i.e., which model wins) as a function of the observed summaries of that data set. Stated formally, we capture this relationship as follows:

$$\hat{m}_k^{\text{Winner}(g)} = \text{Tree}[\tau(\mathbf{Y}_k^{\text{obs}})], \quad (\text{B3})$$

where “Tree” denotes the classification tree predicting the winning model $\hat{m}_k^{\text{Winner}(g)}$ for each of the data sets $k = 1, \dots, K$ and replicates $g = 1, \dots, G$. The results will show which data set-level summaries are associated with differences in performance across the models for the feature of interest. The exact same setup used for CART methods can be used for implementing random forests. The same basic relationships are uncovered, but different methods are used.

References

- Bladt M, Neuts MF (2003) Matrix-exponential distributions: Calculus and interpretations via flows. *Stochastic Models* 19(1): 113–124.
- Breiman L (2001a) Random forests. *Machine Learn.* 45(1):5–32.
- Breiman L (2001b) Statistical modeling: The two cultures. *Statist. Sci.* 16(3):199–231.
- Breiman L, Friedman JH, Olshen RA, Stone CJ (1984) *Classification and Regression Trees* (Chapman & Hall, New York).
- Chib S (1995) Marginal likelihood from the Gibbs output. *J. Amer. Statist. Assoc.* 90(432):1313–1321.
- Chipman HA, George EI, McCulloch RE (1998) Bayesian CART model search. *J. Amer. Statist. Assoc.* 93(443):935–948.
- Cooper LG, Nakanishi M (1988) *Market-Share Analysis: Evaluating Competitive Marketing Effectiveness* (Kluwer Academic Publishers, Boston).
- Donkers B, Verhoef PC, de Jong M (2007) Modeling CLV: A test of competing models in the insurance industry. *Quant. Marketing Econom.* 5(2):163–190.
- Everson PJ, Bradlow ET (2002) Bayesian inference for the beta-binomial distribution via polynomial expansions. *J. Comput. Graphic. Statist.* 11(1):202–207.
- Fader PS, Hardie BGS, Huang CY (2004) A dynamic change-point model for new product sales forecasting. *Marketing Sci.* 23(1):50–65.
- Fader PS, Hardie BGS, Shang J (2010) Customer-base analysis in a discrete-time noncontractual setting. *Marketing Sci.* 29(6):1086–1108.
- Gelman A, Rubin DB (1992) Inferences from iterative simulation using multiple sequences. *Statist. Sci.* 7(4):457–472.
- Gelman A, Rubin DB (1995) Avoiding model selection in Bayesian social research. *Sociol. Methodol.* 25:165–173.
- Gelman A, Carlin JB, Stern HS, Rubin DB (2004) *Bayesian Data Analysis*, 2nd ed. (Chapman & Hall, New York).
- Geweke J (1992) Evaluating the accuracy of sampling-based approaches to calculating posterior moments. Bernardo JM, Berger JO, Dawid AP, Smith AFM, eds. *Bayesian Statistics*, Vol. 4 (Oxford University Press, Oxford, UK), 169–193.
- Hanssens DM, Leeflang PSH, Wittink DR (2005) Market response models and marketing practice. *Appl. Stochastic Models Bus. Indust.* 21(4/5):423–434.
- Hartmann W, Manchanda P, Nair H, Bothner M, Dodds P, Godes D, Hosanagar K, Tucker C (2008) Modeling social interactions: Identification, empirical methods and policy implications. *Marketing Lett.* 19(4):287–304.
- Kamakura WA, Russell GJ (1989) A probabilistic choice model for market segmentation and elasticity structuring. *J. Marketing Res.* 26(4):379–90.
- Lenk PJ (2009) Simulation pseudo-bias correction to the harmonic mean estimator of integrated likelihoods. *J. Comput. Graphic. Statist.* 18(4):941–960.
- Liaw A, Wiener M (2002) Classification and regression by random forest. *R News* 2(3):18–22.
- Liechty JC, Pieters R, Wedel M (2003) Global and local covert visual attention: Evidence from a Bayesian hidden Markov model. *Psychometrika* 68(4):519–541.
- Ma S, Buschken J (2011) Counting your customers from an “always a share” perspective. *Marketing Lett.* 22(3):243–257.
- Montgomery AL, Li S, Srinivasan K, Liechty JC (2004) Modeling online browsing and path analysis using clickstream data. *Marketing Sci.* 23(4):579–595.
- Montoya R, Netzer O, Jedidi K (2010) Dynamic allocation of pharmaceutical detailing and sampling. *Marketing Sci.* 29(5):909–924.
- Morrison DG (1969) On the interpretation of discriminant analysis. *J. Marketing Res.* 6(2):156–163.
- Netzer O, Lattin JM, Srinivasan V (2008) A hidden Markov model of customer relationship dynamics. *Marketing Sci.* 27(2):185–204.
- Newton MA, Raftery AE (1994) Approximate Bayesian inference with the weighted likelihood bootstrap. *J. Roy. Statist. Soc. Ser. B* 56(1):3–48.
- O’Cinneide CA (1990) Characterization of phase-type distributions. *Comm. Statist. Stochastic Models* 6(1):1–57.
- Plummer M, Best N, Cowles K, Vines K (2006) CODA: Convergence diagnosis and output analysis for MCMC. *R News* 6(1):7–11.
- Raftery AE, Lewis SM (1992) How many iterations in the Gibbs sampler? Bernardo JM, Berger JO, Dawid AP, Smith AFM, eds. *Bayesian Statistics*, Vol. 4 (Oxford University Press, Oxford, UK), 763–773.
- Sabavala DJ, Morrison DG (1977) A model of TV show loyalty. *J. Advertising Res.* 17(6):35–43.
- Schmittlein DC, Morrison DG, Colombo RA (1987) Counting your customers: Who are they and what will they do next? *Management Sci.* 33(1):1–24.
- Schweidel DA, Fader PS (2009) Dynamic changepoints revisited: An evolving process model of new product sales. *Internat. J. Res. Marketing* 26(2):119–124.
- Schweidel DA, Bradlow ET, Fader PS (2011) Portfolio dynamics for customers of a multi-service provider. *Management Sci.* 57(3):471–486.
- Schweidel DA, Fader PS, Bradlow ET (2008) Understanding subscriber retention within and across cohorts using limited information. *J. Marketing* 72(1):82–94.
- Spiegelhalter DJ, Best NG, Carlin BP, Linde AVD (2002) Bayesian measures of model complexity and fit. *J. Roy. Statist. Soc. Ser. B* 64(4):583–639.
- Stephens M (2000) Dealing with label-switching in mixture models. *J. Roy. Statist. Soc. Ser. B* 62(Part 4):795–809.
- Tanner MA, Wong WH (1987) The calculation of posterior distributions by data augmentation. *J. Amer. Statist. Assoc.* 82(398):528–540.
- Wittink DR, Addona MJ, Hawkes WJ, Porter JC (1988) SCAN * PRO: The estimation, validation and use of promotional effects based on scanner data. Internal paper, Cornell University, Ithaca, NY.
- Zheng Z, Fader P, Padmanabhan B (2011) From business intelligence to competitive intelligence: Inferring competitive measures using augmented site-centric data. *Inform. Systems Res.* 23(3):698–720.