A Population-Growth Model for Multiple Generations of Technology Products

Hongmin Li • Dieter Armbruster • Karl G. Kempf

W.P. Carey School of Business, Arizona State University
Tempe, Arizona 85287, USA
School of Mathematical and Statistical Sciences,
Arizona State University, Tempe, Arizona 85287, USA
Intel Architecture Group,
Intel Corporation, Chandler, Arizona 85226, USA
hongmin.li@asu.edu • armbruster@asu.edu • karl.g.kempf@intel.com

In this paper, we consider the demand for multiple successive generations of products and develop a population growth model that allows demand transitions across multiple product generations, and takes into consideration the effect of competition. We propose an iterative descent method for obtaining the parameter estimates and the covariance matrix, and show that the method is theoretically sound and overcomes the difficulty that the units-in-use population of each product is not observable. We test the model on both simulated sales data and Intel's high-end desktop processor sales data. We use two alternative specifications for product strength in this market – performance, and performance/price ratio. The former demonstrates better fit and forecast accuracy, likely due to the low price-sensitivity of this high-end market. In addition, the parameter estimate suggests that, for the *innovators* in the diffusion of product adoption, brand switchings are more strongly influenced by product strength than within-brand product upgrades in this market. Our results indicate that compared to the Bass model, the Norton-Bass model, as well as the Jun-Park choice-based diffusion model, our approach is a better fit for strategic forecasting which occurs many months or years before the actual product launch.

Keywords: *Product Transitions, Forecasting, Multiple-generation Demand Model, Diffusion*

1 Introduction

Marketing, producing, and delivering multiple generations of products is becoming an ever-more challenging task for manufacturers of technology products. This paper originates from a collaborative effort with Intel Corporation to build models to support forecasting when the company periodically introduces newer generations of products in the presence of competition. The pace of new product introduction at Intel is driven by advances in both silicon manufacturing technology and product architecture design (Shenoy and Daniel, 2006). Every new product introduces changes in many dimensions: speed, cache size, power consumption, price, and so on. Not only do a product's characteristics affect its own demand, they also dramatically influence the sales of adjacent generations of products, all of which complicate the task of demand forecasting. To

deliver its technology roadmap to the market, Intel develops and synchronizes plans for investing in factories, equipment, production and distribution, each with a different planning time horizon but all depending critically on a good demand forecast.

We focus on long-range forecasting, for which the company needs to model the demand of multiple successive generations of products. Several elements of the forecast are critical. First, long-range planning, which includes building new factories and procuring expensive equipment, occurs many months or even years before the actual products are released to the market. These decisions require information on the aggregate demand of each product over its life cycle, as well as details such as when the demand begins, how fast it ramps, when it peaks, and the peak-level demand. Next, the model needs to capture interactions among the products and account for the competition that Intel faces. Finally, the model should be able to estimate forecast uncertainty because the primary challenge of long-range planning is to mitigate the risk of future uncertainty (Peng et al. 2012, Kempf et al. 2013). In this paper, we abstract from the situation at Intel and develop a general demand model for multiple product generations and show its usefulness in long-range forecasting.

When products are introduced to a market with multiple previous generations of products, a multitude of dynamics and interactions are in effect. We consider three major market dynamics that contribute to demand: (i) existing customers (i.e., those who own an older-version product) upgrading to newer products, (ii) brand switching by customers, and (iii) market expansion. In this paper, we develop a model that focuses on product upgrades and brand switchings, while incorporating market expansion as a trend correction. In other words, we do not model the macro dynamics driving the total market expansion (such as the state of the economy, the trend of end-customer consumption, and the changing market appetite for technology), but view our model as a tool for forecasting the demand curve of each individual product, given the trend of market expansion.

1.1 Relationship to Prior Research

Bass (1969) characterizes the consumers for durable goods as a combination of innovators, who adopt the product at a constant rate and imitators whose adoption rate depends on the current population of adopters. The resulting demand resembles a diffusion process. Compared to the time-series methods which are primarily data-oriented, the Bass model takes into consideration the underlying market dynamics to predict demand. Researchers have since extended the Bass model to incorporate demand-influencing factors such as advertising, price, and product-specific attributes

(Bass, 1980; Bass et al., 1994; Kamakura and Balasubramanian, 1988; Jain and Rao, 1990; Kalish, 1985), as well as Bayesian updating of the diffusion parameters using early market data (Wu et al., 2010). However, these extensions are limited to a single product diffusion model.

Two recent review articles (Meade and Islam, 2006; Peres et al., 2010) summarize related work on diffusion between technology generations. Fisher and Pry (1971) model the substitution of a new technology for the old technology assuming that the market share of the new technology grows with an exponential rate. Their model is limited to two products and captures the demand during only the transition period instead of each product's *entire* life cycle. Norton and Bass (1987) consider the diffusion of successive generations of products (which we refer to as the Norton-Bass model hereafter). They combine product substitution with diffusion and allow the adoption of the next generation product be composed of two parts: those from the untapped market potential, and those from adopters of the old product upgrading to the newer product. The Norton-Bass model yields the overlapping bell-shaped demand curves commonly observed when multiple generations of products are sold concurrently. However, the complexity of this model increases dramatically with the number of products. Another limitation of this model is that product substitution only occurs between two adjacent generations, not across multiple generations. For semiconductor products, customers often leapfrog as they upgrade and the ability to capture such detail allows a firm to design market strategies that target specific populations (see Gordon, 2009). Moreover, both the Bass model and its extensions usually require data observations that include the demand peak. Therefore, these models are more useful if a substantial number of sales observations are already available for the product to be forecasted. This inevitably limits the prediction window to a much shorter time period than that required by long-range planning decisions.

Our paper is also related to choice-theory-based demand models such as Melnikov (2001), Song and Chintagunta (2003), Gordon (2009) and Gowrisankaran and Rysman (2009), in which consumers' purchasing behavior is modeled as a utility maximization problem. A general drawback of these models is that parametrization is computationally intensive and often product aggregation is necessary (see Gordon (2009)). In comparison, our approach reproduces complicated time series data at the level of individual product with relatively small computational effort.

Jun and Park (1999) combine a choice model with a diffusion model to predict sales of multiple generations of products. They assume an aggregate Bass diffusion for the entire market and let the share of sales for each product be determined by a logit choice probability. In particular, the "type II" model (which we refer to as the Jun-Park model hereafter) described in this paper can be parameterized in the absence of unit-in-use data. They achieve this by mixing product

upgrades with first-time-purchases. In contrast, our model differentiates these different sources of sales, enabling the design of population-specific marketing strategies. The Jun-Park model uses product-specific parameters and thus its application is restricted to two-step-ahead or three-step-ahead forecasts, or to naively copy sales of a previous-generation product as the forecast for a new product. In addition, they model customers' utility as a linear function of time, thus customers' valuation of a product is assumed to change monotonically with time throughout its life time. Consequently, for a new product to replace the older generations, the time coefficient has to always increase from one product to the next, regardless of product strength. This confounds parameter interpretation and makes the model difficult to apply (since one cannot predict what the time coefficient would be for a *new* product). In contrast, our model provides both clear interpretations for the parameters and a clear path for how to forecast sales of future products.

Bayus et al. (2000) review a two-product population growth model and show that several previously studied models, including the Norton-Bass model and the Lotka-Volterra (Murray, 2002) predator-prey model, can all be considered special cases of this model. The population growth model, often used in ecology (Pielou, 1977) and sociology (Tuma and Hannan, 1984), has clear advantages over the Norton-Bass model: It captures product interactions, allows generation leapfrogging, and allows an arbitrary number of products to coexist. However, existing applications are limited to cases where the population sizes are directly observable or can be easily estimated, for example, Mahajan and Muller (1996) on the demand for IBM mainframe computers and Kim et al. (2000) on subscriptions of telecommunication services. In both papers, the population for product-in-use is easily identifiable by the number of service contracts in place. This is not the case for most other products. For example, at Intel, a customer who purchases the newest generation i microprocessor could previously be a user of generation $i-1, i-2$, or a user of the competitor's products, which is not observed by Intel. In addition, the sales of generation i product do not reveal how many customers have left generation i, making it impossible to track the size of population i. Furthermore, sales data of competition are difficult to obtain. Our approach builds upon a population-growth model but overcomes the limitation of non-observable population size and the lack of sales data of the competition.

In this paper, we do not consider supply constraints and use the terms "demand" and "sales" interchangeably. For new product diffusion models under supply constraints, one may refer to Ho et al. (2002) and Kumar and Swaminathan (2003), which extend the single-product Bass model. In addition, we do not consider used or remanufactured products and their impacts on the diffusion dynamics, which is the subject of a related paper by Debo et al. (2006).

1.2 Summary of Contribution and Organization

We present a demand model for multiple generations of products and develop a novel parametrization method that takes advantage of the flexibility afforded by the population growth model even when the population data cannot be obtained. We show that this method performs well on synthetic data, generated by a known demand obscured by noise. We then apply this method to Intel's microprocessor sales data and show that it outperforms other alternatives.

Our model is more appropriate for long-range forecast than existing models because it does not need product-specific parameters to forecast sales. For instance, the Bass model requires sales data for a particular product to first derive the diffusion parameters of this product and then forecast for its remaining life time. With multiple products, the number of parameters grows combinatorially: Not only does each product add its own set of diffusion parameters, but for each pair of products, additional parameters are needed to model product interactions. (See, for example, Mahajan and Muller (1996) and Danaher et al. (2001)). Furthermore, it is not clear how product-specific differences should be taken into account to modify these parameters for future products. In comparison, we parameterize the model based on product strength, which enables forecast for products that are not yet released to the market and even years away from the time of forecast.

To our knowledge, our model is the first to combine brand switchings and within-brand product upgrades among multiple product generations into one model framework. Existing work on diffusion models with competition only considers one product for the focal firm (see, for example, Savin and Terwiesch 2005, Libai et al. 2009).

Finally, we show in this paper how to estimate the parameter variances which characterize the confidence of the forecast, as well as how to adjust the variance estimation when the assumption of independent and identical noise does not hold.

The rest of the paper is organized as follows. Section 2 describes the multi-product demand model in detail. In Section 3, we present the basic idea for overcoming the problem of unobservable population size. We examine the identification condition for this model and prove convergence of the proposed method. In Section 4, we test the model using stochastically generated sales data. We apply the model to the microprocessor data supplied by Intel in Section 5. We then compare the model's fit and forecast performance with the Bass model, the Norton-Bass model, as well as the Jun-Park model. We conclude in Section 6, summarizing the key assumptions and discussing the limitations.

2 Model Description and Assumptions

In this section, we present a discrete-time population growth model for multiple generations of products. Assume that a company is currently selling a total of n generations of product on the market, indexed by the order of each product's market entry. We associate a population x_i with each product $i = 1, \ldots, n$, indicating the current number of units-in-use for this product. We assume that a customer will never purchase a product that is older (in terms of the product's introduction time) than the one he currently owns. In addition, once a customer purchases a new product, he will scrap the old product he previously owned or downgrade it to a secondary usage. Therefore, the state of a customer can be represented by i – the latest product he owns. Similar to the Bass model, we assume that each customer purchases at most one unit of product each time.

We consider H time periods. Let $x_i(t)$ be the population of product i at the beginning of time period t, and $s_i(t)$ be the sales of product i during period t. At the beginning of the focal time horizon, we assume that the market starts with an existing population of products-in-use of some earlier generation(s). These may include product generations that are older than product 1, which are not selling any more but still have a unit-in-use population. Let $K = \{-k, \ldots, -1, 0\}$ be the set of these older products. We assume that $x_i(0),\ i = -k, \ldots, 0, 1, \ldots, n$ are given, with $x_i(0) = 0$ if product i has not been introduced yet. As we show later in applications on both simulated and Intel data, the method is robust to perturbations in the initial population size.

2.1 Product Upgrades

As customers of an older product upgrade to a newer product, sales occur and the population x_i evolves. Specifically, the value of x_i increases if a customer who previously owned an older product purchases product i and decreases if a customer who previously owned product i decides to buy a newer product. Let P_{ij} be the fractional flow rate from population i to j, i.e., the fractional rate at which a customer of product i will buy product j. The population evolution could then be described by the difference equation

$$x_i(t+1) - x_i(t) = \sum_{j<i} x_j(t) P_{ji} - x_i(t) \sum_{j>i} P_{ij},\ i = 1, \ldots, n\ , \tag{2.1}$$

and the sales rate of product i due to upgrades is given by $\sum_{j<i} x_j(t) P_{ji}$, which is the first term of the right side of equation (2.1).

2.2 Brand Switching

The diffusion of a new product is affected not only by adjacent generations of products sold by the same firm but also by products from competitors. In many cases, the competing firms also sell successive generations of products to the same pool of customers. As a result, the population flow could occur across brands and between *any* two products on the market. However, modeling the flow between each product of the focal company and each competing product is not desirable because product-level sales data from competition are not readily available. In this paper, we do not differentiate individual products sold by competitors, but instead treat them as one single population y, which has a time-varying strength $f_y(t)$, reflecting the improvement of competitive products over time. In practice, there may be multiple competing products and one has to determine $f_y(t)$ carefully. For example, one may view $f_y(t)$ as either the average strength of competing products, or the strength of the best competing product at time t. Similar to the assumption of known $x_i(0)$, we assume that $y(0)$ is known.

We assume that the population flow from population j to the competition or from the competition to population j is determined by the gap of product strength. Specifically, if the strength of product j is higher (lower) than $f_y(t)$, then there is a flow from population y to population j (population j to y) but none from j to y (y to j). Let $J(t)(\overline{J}(t))$ be the set of products stronger (weaker) than the competition. Clearly the set J may change with time. Denote the fractional flow rate from $x_i, i \in \overline{J}(t)$ to y as P_{iy} and that from y to $x_i, i \in J(t)$ as P_{yi}. Therefore, sales of product i due to brand switching are given by $[y(t)P_{yi}I(i \in J(t))]$, where $I(\cdot)$ is an indicator function.

2.3 Market Expansion

A third source of sales comes from "new" customers, i.e., customers who have not previously purchased a product in this market, whether from the focal company or from competition. As discussed earlier, this is driven by multi-facet macroeconomic factors such as world economy and overall development of technology. At Intel, forecast for the total market is a separate process from that for individual products. In this paper, we follow the Intel practice and propose a simple approach to correct for the overall market trend. We incorporate this demand source through a known percentage growth rate $\alpha(t),\ t = 1, ..., H$. Let $s(t)$ define the total sales of this market (including both the focal firm and its competition) in period t. We assume that sales due to market expansion in period t is given by $\alpha(t) \cdot s(t-1)$ and that this sales growth is split proportionally among products of both the focal firm and the competition based on each product's most recent

market share. In other words, if the total market grows by $\alpha(t) \cdot s(t-1)$ in period t where $s(t-1)$ is the total sales in the previous period, then product i gains $\alpha(t) \cdot s_i(t-1)$. While this is a simplification, if one can safely assume that market growth is from a population that is similar to the current adopter population, this proportional split assumption is reasonable. In addition, the assumption of exogenous market expansion leads to a flexible model that accommodates essentially any trend of the overall market expansion.

2.4 Resulting Sales

Summarizing the three sources of population flows, we obtain the sales for each product at time t:

$$s_i(t) \;=\; \sum_{j<i} x_j(t) P_{ji} + y(t) P_{yi} I(i \in J(t)) + \alpha(t) s_i(t-1) \;\forall\; t = 1, 2, ..., H \;, \tag{2.2}$$

where the three terms on the right represent sales due to upgrade, brand switching and market expansion respectively.

We assume that, similar to the Bass model, the fractional flow rate from population i to j is given by

$$P_{ij} = p_{ij} + q_{ij} x_j \;, \tag{2.3}$$

where p_{ij} represents an *innovation* effect and q_{ij} the *word-of-mouth* effect. Furthermore, we assume that the parameters p_{ij} and q_{ij} are linearly dependent on product strength:

$$p_{ij} \;=\; \beta_1 + \beta_2 f_{ij} \;, \quad q_{ij} \;=\; \beta_3 + \beta_4 f_{ij} \;, \tag{2.4}$$

where f_{ij} is the difference in product strength between product i and product j measured in percentage improvement. The parameters β_2 and β_4 characterize the importance of product strength whereas β_1 and β_3 incorporate transitions that are independent of product strength. A linear relationship is commonly adopted by researchers for estimating the impact of influencing factors due to its simplicity (e.g., Bass et al. 1994). We take a similar approach for including product strength.

Let f_{iy} (f_{yi}) represent the percentage improvement of product i over product y (product y over product i), we assume that the diffusion parameters p_{iy}, p_{yi}, q_{iy} and q_{yi} satisfy

$$p_{iy} \;=\; \beta_5 + \beta_6 f_{iy} \;, \quad q_{iy} \;=\; \beta_7 + \beta_8 f_{iy} \;, \;\forall i \in \overline{J} \;, \tag{2.5}$$

$$p_{yi} \;=\; \beta_5 + \beta_6 f_{yi} \;, \quad q_{yi} \;=\; \beta_7 + \beta_8 f_{yi} \;, \;\forall i \in J \;. \tag{2.6}$$

Note that the flows from x_i to y and from y to x_i have the same coefficients β_k, $k =$ 5-8. This is based on the observation at Intel that customers who switch brands tend to have similar characteristics.

In the rare case where this is not true, the model could be extended at the cost of additional parameters.

We obtain the sales of product i by substituting equations (2.4), (2.5) and (2.6) into equation (2.2):

$$\begin{aligned} s_i(t) &= \beta_1 \left[\sum_{j<i} x_j(t)\right] + \beta_2 \left[\sum_{j<i} f_{ji}x_j(t)\right] + \beta_3 \left[\sum_{j<i} x_i(t)x_j(t)\right] + \beta_4 \left[\sum_{j<i} f_{ji}x_i(t)x_j(t)\right] \\ &\quad + \beta_5 \left[y(t)I(i \in J)\right] + \beta_6 \left[f_{yi}y(t)I(i \in J)\right] + \beta_7 \left[x_i(t)y(t)I(i \in J)\right] \\ &\quad + \beta_8 \left[f_{yi}x_i(t)y(t)I(i \in J)\right] + \alpha(t)s_i(t-1) \ . \end{aligned} \tag{2.7}$$

Therefore, conditional on $\mathbf{x}(t) = (x_1(t), \ldots, x_n(t))$ and $y(t)$, $s_i(t) - \alpha(t)s_y(t-1)$ is a linear function of the parameter vector $\boldsymbol{\beta}= (\beta_1, \ldots, \beta_8)$. We define a matrix $\mathbf{X}$ with dimension $nH \times 8$ such that

$$\begin{aligned} \mathbf{X}_{t+(i-1)H} &= \Bigg(\sum_{j<i} x_j(t),\ \sum_{j<i} f_{ji}x_j(t),\ \sum_{j<i} x_i(t)x_j(t),\ \sum_{j<i} f_{ji}x_i(t)x_j(t),\ y(t)I(i \in J), \\ &\qquad f_{yi}y(t)I(i \in J),\ x_i(t)y(t)I(i \in J),\ f_{yi}x_i(t)y(t)I(i \in J) \Bigg) , \end{aligned} \tag{2.8}$$

where $\mathbf{X}_{t+(i-1)H}$ is the $[t + (i-1)H]_{th}$ row vector of $\mathbf{X}$.

Then we can rewrite equation (2.7) as

$$\mathbf{s} = \mathbf{X}\boldsymbol{\beta} \ , \tag{2.9}$$

$$\text{where} \qquad \mathbf{s} = \Big(\{s_i(t) - \alpha(t)s_i(t-1)\}_{i=1,2,\ldots,n;\ t=1,2,\ldots,H} \Big) \tag{2.10}$$

is a vector with its $[t + (i-1)H]_{th}$ element equal to $s_i(t) - \alpha(t)s_i(t-1)$, representing the sales of product i during period t "corrected" for the market trend. Throughout the rest of the paper, we assume that the matrix $\mathbf{X}$ has full rank.

To ensure that the discrete time model is well behaved, we assume $\boldsymbol{\beta}$ is small such that sales in and out of each population are small relative to the current population size. Consequently the values of $x_i(t)$ are finite and nonnegative. From an implementation perspective, this is equivalent to keeping the discrete time unit sufficiently small.

Suppose we know the values of $x_i(t)$, $y(t)$ and the sales $s_i(t)\ \forall\ i, t$, then we can solve the linear system of equations given by (2.9) to obtain the parameter $\boldsymbol{\beta}$. In the case with measurement error in sales, one can obtain the estimate for $\boldsymbol{\beta}$ using the linear regression model

$$\mathbf{s} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon} \ . \tag{2.11}$$

where $\boldsymbol{\epsilon}$ is a vector of independent and normally distributed noises. Unfortunately, as discussed earlier, $x_i(t)$ are in most cases not available and therefore we cannot use linear regression methods to estimate $\boldsymbol{\beta}$. Rather, the model we need to estimate is

$$\mathbf{s} = \mathbf{X}(\boldsymbol{\beta})\boldsymbol{\beta} + \boldsymbol{\epsilon} \ , \tag{2.12}$$

where $\mathbf{X}$ is a function of $\boldsymbol{\beta}$.

3 Solving the Nonlinear Regression

The conventional nonlinear regression method for estimating $\boldsymbol{\beta}$ in equation (2.12) involves minimizing the sum of squares

$$v(\boldsymbol{\beta}) \equiv (\mathbf{X}(\boldsymbol{\beta})\boldsymbol{\beta} - \mathbf{s})^T(\mathbf{X}(\boldsymbol{\beta})\boldsymbol{\beta} - \mathbf{s}) \tag{3.1}$$

by optimizing $\boldsymbol{\beta}$. Substituting equation (2.3) into equations (2.1) and (2.2), we obtain the population and sales as quadratic recursive equations (see Online Appendix A.1 for details), which implies that $s_i(t)$, $x_i(t)$ (similarly, $s_y(t)$ and $y(t)$) are polynomial functions of the parameter vector $\boldsymbol{\beta}$ with order 2^t. Hence the problem of minimizing $v(\boldsymbol{\beta})$ over the parameter vector $\boldsymbol{\beta}$ is of polynomial order 2^{2t}, which is practically infeasible to solve for any reasonably large t.

Given the special structure of this problem, we propose an iterative procedure that takes advantage of the linear structure of equation (2.7) to obtain the optimal parameter estimates. Although the population paths $x_i(t)$ and $y(t)$, $t = 1, ..., H$ are not known, we can *construct* them based on the current parameter estimates. Then, from the constructed population paths, we obtain an updated estimate of the parameter vector using the constructed data matrix and the actual sales vector. In particular, we perform linear regression using equation (2.11). We then repeat this process until the parameter estimates converge. In other words, we use the constructed data matrix $\mathbf{X}(\boldsymbol{\beta})$ as the "pseudoregressor." The following is a step-by-step description of the procedure.

- Step 1: Assume that we know the values of x_i and y at time $t = 0$. In the kth iteration, we use the current estimate $\boldsymbol{\beta}^k$ for the parameter vector $\boldsymbol{\beta}$ to construct the population path $\hat{x}_i(t)$ and $\hat{y}(t)$ for $t = 1, \ldots, H$ following equations (A.1) – (A.3) (note that $\hat{s}_y(t)$ is also constructed as an intermediary), and then derive the sales path $\hat{s}_i(t)$ using equation (2.7).

- Step 2: Construct the matrix $\mathbf{X}(\boldsymbol{\beta}^k)$ and the column vector $\mathbf{s}(\boldsymbol{\beta}^k)$ as defined in equations (2.8) and (2.10). Next, run linear regression of $\mathbf{s}(\boldsymbol{\beta}^k)$ against $\mathbf{X}(\boldsymbol{\beta}^k)$ to obtain an updated set

of parameters given by

$$\boldsymbol{\beta}^{k+1} \equiv (\beta_1^{k+1},\ \beta_2^{k+1},\ \ldots\ \beta_8^{k+1})^T = [\mathbf{X}(\boldsymbol{\beta}^k)^T\mathbf{X}(\boldsymbol{\beta}^k)]^{-1}\mathbf{X}(\boldsymbol{\beta}^k)^T\mathbf{s}(\boldsymbol{\beta}^k). \tag{3.2}$$

- Repeat steps 1 and 2 using the updated parameters $\boldsymbol{\beta}^{k+1}$ until convergence, i.e, when the percentage improvement of the residual sum of square (which approximates the scaled norm of the gradient) falls below a very small number.

The iterative method described above is conceptually similar to a fixed-point iteration method for solving a system of nonlinear equations. This can be seen by omitting the error term ϵ and rewriting equation (3.2) as $\boldsymbol{\beta} = [\mathbf{X}(\boldsymbol{\beta})^T\mathbf{X}(\boldsymbol{\beta})]^{-1}\mathbf{X}(\boldsymbol{\beta})^T\mathbf{s}$. For a fixed point method to work, the mapping from $\boldsymbol{\beta}^k$ to $\boldsymbol{\beta}^{k+1}$ needs to be a contraction. However, this is not generally true, even without the error term $\boldsymbol{\epsilon}$. Therefore, the method described above does not always converge. Indeed, we observe both cases of convergence and cases of local divergence where the sequence $\boldsymbol{\beta}^k$ oscillates around the fixed point but never converges. In the next two subsections, we examine the mathematical conditions required for the model in equation (2.12) to be *identifiable* and show that convergence can be achieved by modifying equation (3.2).

Before we proceed, we note that this iterative approach is analogous to the well-known Gauss-Newton nonlinear regression method (Amemiya, 1985), in which the nonlinear model is linearized based on the Taylor series approximation at an initial parameter estimate, and then a new set of parameter estimate obtained from the linear regression is used as the new starting parameter value for subsequent iterations. In our model, we also take advantage of a linear regression step but it is based on the special structure of this multi-product demand model instead of the Taylor series approximation. In addition, our method bears some resemblance to the Expectation-Maximization (EM) method (Dempster et al., 1977) which obtains the maximum likelihood estimate under incomplete data. In the EM method, one uses an initial estimate of the parameter to compute conditional distributions of the missing data and then a new estimate of the parameter is derived by maximizing the expected log likelihood function. In our problem, we derive the expected values of the "missing data", namely, the population sizes, based on the current parameter estimate so as to reduce a complex nonlinear regression to a series of simple linear regressions.

3.1 Model Identification

In the iterative approach described above, we circumvent the problem of unobservable population paths by constructing them using current best estimates of the parameters. Because of the missing information, we encounter the problem of parameter identifiability.

A set of model parameters is *identifiable* if no other set of parameter values leads to the same probability distribution of the dependent variables, in which case the two parameter points are *observationally equivalent* (Rothenberg, 1971). If two parameter points are observationally equivalent, then we cannot statistically distinguish one from the other. In this problem, this would imply that there might be multiple sets of β parameter values that could generate the same sales distribution. In the special case where the measurement error is zero, identifiability is equivalent to the existence of a *unique* fixed point.

In our problem, if the population paths $x_i(t), t = 1, ..., H$ are observable, then the model described by equation (2.7) is a linear model, $\mathbf{s} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}$, which is identifiable if the error term ϵ_i is zero mean, independent of $\mathbf{X}$, and the matrix $\mathbf{X}$ is of full rank (Greene, 2003).

However, the observation of $\mathbf{X}$ is in general not readily available and the true model is $\mathbf{s} = \mathbf{X}(\boldsymbol{\beta})\boldsymbol{\beta} + \boldsymbol{\epsilon}$, which is non-linear. Rothenberg (1971) shows that a non-linear model is locally identifiable if the information matrix as defined by $\mathbf{R}(\boldsymbol{\beta}) = [r_{ij}(\boldsymbol{\beta})] = E\left[\frac{\partial log f}{\partial \beta_i} \cdot \frac{\partial log f}{\partial \beta_j}\right]$, where f is the probability density of the dependent variable for a given set of parameter values $\boldsymbol{\beta}$, is non-singular at any regular point of the matrix $\mathbf{R}(\boldsymbol{\beta})$. In addition, if f belongs to a special class of the exponential family (e.g., multivariate normal), then the parameter vector $\boldsymbol{\beta}$ is globally identifiable. A straightforward application of the Rothenberg result leads to the following proposition.

Proposition 3.1. *Assume that a noise term $\epsilon_i(t)$ is added to the sales $s_i(t)$ (equation (2.7)), where $\epsilon_i(t), i = 1, ..., n$ are of independent normal distributions. Then* global *identification requires $[\nabla_{\boldsymbol{\beta}}(\mathbf{X}(\boldsymbol{\beta})\boldsymbol{\beta})]^T$ to be of full rank.*

As pointed out by Amemiya (1985), nonlinearity generally helps identification so that the full rank requirement does not imply that the number of variables needs to be greater than or equal to the number of parameters. Therefore, the condition that $[\nabla_{\boldsymbol{\beta}}(\mathbf{X}(\boldsymbol{\beta})\boldsymbol{\beta})]^T$ be of full rank is not restrictive but in fact is easier to satisfy than in an entirely linear system.

3.2 Model Convergence

The identification condition ensures that no two parameter sets generate the same probability distribution of sales. However, it does not guarantee that for any given set of sales data, the procedure converges or that it converges to a stationary point of the optimization problem $\boldsymbol{\beta} = \arg\min_{\boldsymbol{\beta}} v(\boldsymbol{\beta})$ where $v(\boldsymbol{\beta})$ is defined in equation (3.1).

In this section, we present a modification of equation (3.2) that leads to convergence. Specifically, the parameter values used in the next iteration $\boldsymbol{\beta}^{k+1}$ is determined as follows: Let $X(\boldsymbol{\beta}^k)$

be the population matrix constructed using the parameter vector $\boldsymbol{\beta}^k$ and let $\mathbf{b}^k$ be the optimal parameter values obtained through the linear regression using equation (2.11), i.e., $\mathbf{b}^k = [X(\beta^k)^T\mathbf{X}(\boldsymbol{\beta}^k)]^{-1}\mathbf{X}(\boldsymbol{\beta}^k)^T\mathbf{s}$. We define the direction vector $\mathbf{d}^k = \mathbf{b}^k - \boldsymbol{\beta}^k$ and update the parameter estimates by $\boldsymbol{\beta}^{k+1} = \boldsymbol{\beta}^k + \lambda^k\mathbf{d}^k$ where $\lambda^k \in (0, 1]$ represents a scalar step size.

To fix ideas, we follow the definition in Bertsekas (2003) regarding a descent direction in gradient descent algorithms.

Definition 1. Let $f(\boldsymbol{\beta})$ be a continuously differentiable function of the vector $\boldsymbol{\beta}$. A sequence $\{\mathbf{d}^k\}_{k\in\kappa}$ is *gradient related* to $\boldsymbol{\beta}^k$ if $\{\mathbf{d}^k\}_{k\in\kappa}$ is bounded and $\lim_{k\to\infty}\sup_{k\in\kappa}\nabla f(\boldsymbol{\beta}^k)^T\mathbf{d}^k < 0$.

Lemma 3.2. *Define the sequence* $\{\mathbf{d}^k\}$ *and* $\{\boldsymbol{\beta}^k\}$ *such that* $\mathbf{d}^k = \mathbf{b}^k - \boldsymbol{\beta}^k$ *and* $\boldsymbol{\beta}^{k+1} = \boldsymbol{\beta}^k + \lambda^k\mathbf{d}^k$. *Assume that* $\left[\nabla_{\boldsymbol{\beta}}(\mathbf{X}\boldsymbol{\beta}^k)\right]^T\left[\mathbf{X}^T\mathbf{X}\right]^{-1}\mathbf{X}^T$ *is positive definite. Then the sequence* $\{\mathbf{d}^k\}$ *is gradient related to* $\{\boldsymbol{\beta}^k\}$.

The proof of Lemma 3.2 is provided in the Online Appendix. Employing a result in Bertsekas (2003) (Prop. 1.2.1, page 43), we show that the sequence $\{\boldsymbol{\beta}^k\}$ converges to a stationary point of $v(\boldsymbol{\beta})$ if $\boldsymbol{\beta}^k$ is sufficiently small and the step size λ^k is properly chosen.

Corollary 3.3. *Assume that the positive definiteness condition in Lemma 3.2 is satisfied. If the step size* λ^k *is chosen by the Armijo rule or any step size rule that yields a larger cost reduction, i.e., a larger reduction in* $v(\boldsymbol{\beta})$, *at each iteration step than the Armijo rule, the sequence* $\{\boldsymbol{\beta}^k\}$ *converges to a stationary point of* $v(\boldsymbol{\beta})$.

The Armijo rule is a successive reduction rule such that a sufficiently large cost reduction is achieved (see Bertsekas (2003) for more details). The proof of the above corollary is straightforward: The positive definiteness condition leads to a gradient-related sequence $\{\mathbf{d}^k\}$. According to Prop. 1.2.1 in Bertsekas (2003), if the step size is determined by the Armijo rule or one with higher cost reduction than the Armijo rule in each step, then every limit point of $\{\boldsymbol{\beta}^k\}$ is a stationary point. A direct consequence of Corollary 3.3 is that the estimator obtained using this approach is the nonlinear least squares estimator by definition (see Greene (2003)).

Corollary 3.4. *Suppose that the positive definiteness condition in Lemma 3.2 holds. Then the iterative approach described in Corollary 3.3 yields the nonlinear least squares estimator.*

We refer to the iterative approach in Corollary 3.3 as the "iterative descent" approach hereafter. In general, the condition in Lemma 3.2 is difficult to verify even for a given $\boldsymbol{\beta}^k$. However, we show that this condition is always satisfied asymptotically when $\boldsymbol{\beta}^k \to 0$.

Proposition 3.5. *The matrix* $\lim_{\boldsymbol{\beta}\to 0}[\nabla(\mathbf{X}\boldsymbol{\beta})]^T(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T$ *is positive definite and thus the iterative descent approach converges to a stationary point of* $v(\boldsymbol{\beta})$*, i.e., the nonlinear least squares estimator of* $\boldsymbol{\beta}$*, if* $\boldsymbol{\beta}^k \to 0$*.*

The proof of Proposition 3.5 is given in the Online Appendix A.3.

Next, we remark that as $\boldsymbol{\beta} \to 0$, our method not only converges, but also converges fast. Since $\mathbf{X}$ is full rank, $\mathbf{X}^T\mathbf{X}$ is invertible. Thus we can rewrite $\mathbf{d}^k$ as

$$\mathbf{d}^k \;=\; -(\boldsymbol{\beta}^k - \mathbf{b}^k) \;=\; -\left[\boldsymbol{\beta}^k - (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{s}\right] = -(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T(\mathbf{X}\boldsymbol{\beta}^k - \mathbf{s}) \;,$$

where we omit the argument $\boldsymbol{\beta}^k$ of $\mathbf{X}$ for brevity. From the definition of $v(\boldsymbol{\beta})$, we have $\nabla v(\boldsymbol{\beta}^k) = 2[\nabla(\mathbf{X}\boldsymbol{\beta}^k)](\mathbf{X}\boldsymbol{\beta}^k - \mathbf{s})$. Therefore, as $\boldsymbol{\beta}^k \to 0$, the descent direction $d^k = -(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T(\mathbf{X}\boldsymbol{\beta}^k - \mathbf{s}) \to -\frac{1}{2}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T(\mathbf{X}\mathbf{X}^T)^{-1}\mathbf{X}\nabla v(\boldsymbol{\beta}^k)$. Due to the full rank assumption of X, the matrix $(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T(\mathbf{X}\mathbf{X}^T)^{-1}\mathbf{X}$ is positive definite (see Online Appendix A.4). Therefore, our method is a Quasi-Newton method, which typically converges fast (Bertsekas 2003, Page 148) when used in combination with the Armijo rule or the minimization rule (in which a search along the direction of d^k is performed to find the step size that maximizes cost reduction).

3.3 Estimate the Covariance

When the condition in Proposition 3.5 is satisfied, the method converges to the set of parameter values that minimizes the sum of squared errors, i.e., the nonlinear least squares estimator. Consequently, it retains the asymptotic properties of nonlinear least squares estimator, i.e., *consistency* and *asymptotic normality* under very general conditions.

Proposition 3.6. *Suppose the parameter space of* $\boldsymbol{\beta}$ *is compact. Further, suppose* $lim_{N\to\infty} v(\cdot)$ *is a continuous and differentiable function and has a unique minimum at the true parameter* $\boldsymbol{\beta}$*, where* $N \equiv nH$*. In addition, assume that the errors* $\boldsymbol{\epsilon}$ *are independent and homoscedastic, i.e.,* $E[\boldsymbol{\epsilon}\boldsymbol{\epsilon}^T] = \sigma^2\mathbf{I}$ *where* σ^2 *is a scalar and* $\mathbf{I}$ *is the identity matrix. Let* $\hat{\boldsymbol{\beta}}$ *be the estimator obtained from the iterative descent approach. Then* $\hat{\beta}$ *is* consistent*, i.e.,* $lim_{N\to\infty}\hat{\boldsymbol{\beta}} = \boldsymbol{\beta}$*, and* asymptotically normal *, i.e.,* $lim_{N\to\infty}\hat{\boldsymbol{\beta}} \sim Normal\left(\boldsymbol{\beta}, \frac{\sigma^2}{N}\mathbf{Q}^{-1}\right)$ *where* σ^2 *is the variance of* $\boldsymbol{\epsilon}$ *and*

$$\mathbf{Q} = \lim_{N\to\infty}\frac{1}{N}\sum_{i=1}^{N}\left(\frac{\partial[\boldsymbol{\chi}_i(\hat{\boldsymbol{\beta}})\hat{\boldsymbol{\beta}}]}{\partial\hat{\boldsymbol{\beta}}}\right)\left(\frac{\partial[\boldsymbol{\chi}_i(\hat{\boldsymbol{\beta}})\hat{\boldsymbol{\beta}}]}{\partial\hat{\boldsymbol{\beta}}^T}\right)$$

where $\boldsymbol{\chi}_i(\hat{\boldsymbol{\beta}})$ *is the ith row of* $\mathbf{X}(\hat{\boldsymbol{\beta}})$*.*

The result follows from Greene (2003) (Theorems 9.1 and 9.2). A useful consequence of this is that we can estimate the asymptotic variance of the estimator $\hat{\boldsymbol{\beta}}$. It is easy to show that $\frac{\partial \boldsymbol{\chi}_i(\hat{\boldsymbol{\beta}})}{\partial \hat{\boldsymbol{\beta}}}$ is bounded (see the proof of Proposition 3.5 in the Online Appendix). Since $\frac{\partial [\boldsymbol{\chi}_i(\hat{\beta})\hat{\beta}]}{\partial \hat{\beta}} = \boldsymbol{\chi}_i(\hat{\beta}) + \hat{\beta}\frac{\partial \boldsymbol{\chi}_i(\hat{\beta})}{\partial \hat{\beta}}$, we have $lim_{\boldsymbol{\beta}\to 0} Q = lim_{N\to\infty}(\mathbf{X}^T\mathbf{X})$. In addition, σ^2 can be estimated with $\mathbf{e}^T\mathbf{e}/N$ where $\mathbf{e}$ is the residual vector. Therefore, when β is small, we can estimate the covariance matrix of $\boldsymbol{\beta}$ by $\frac{\mathbf{e}^T\mathbf{e}}{N}(\mathbf{X}^T\mathbf{X})^{-1}$.

Correction for Heteroscedasticity and Autocorrelation

While Proposition 3.6 suggests a straightforward method to estimate the covariance matrix of the parameter estimator, it relies on the assumption that the error term $\boldsymbol{\epsilon}$ is spherical, i.e., $E[\boldsymbol{\epsilon}\boldsymbol{\epsilon}^T] = \sigma^2\mathbf{I}$ where $\mathbf{I}$ is the identity matrix. If one could assume that the error term $\boldsymbol{\epsilon}$ is independent of the underlying demand generating process, but rather just a book-keeping error of sales, this would be a reasonable way to estimate the covariance. In general, it is difficult to argue that the error term has constant variance over time. To deal with heteroscedasticity, we can use the White estimator (White, 1980) to estimate the asymptotic variance:

$$\frac{1}{N}\left(\frac{\mathbf{Z}^T\mathbf{Z}}{N}\right)^{-1}\left(\frac{1}{N}\sum_{i=1}^{N} e_i^2\mathbf{z}_i\mathbf{z}_i^T\right)\left(\frac{\mathbf{Z}^T\mathbf{Z}}{N}\right)^{-1},$$

where $\mathbf{Z} = \frac{\partial\left[\mathbf{X}(\hat{\boldsymbol{\beta}})\hat{\boldsymbol{\beta}}\right]}{\partial\hat{\boldsymbol{\beta}}}$ and $\mathbf{z}_i$ is the ith row of $\mathbf{Z}$. As $\boldsymbol{\beta}\to 0$, we estimate the covariance matrix of $\boldsymbol{\beta}$ with $N(\mathbf{X}^T\mathbf{X})^{-1}S_0(\mathbf{X}^T\mathbf{X})^{-1}$ where $S_0 = \frac{1}{N}\sum_{i=1}^{N} e_i^2\boldsymbol{\chi}_i\boldsymbol{\chi}_i^T$.

For more general cases in which autocorrelation in the data cannot be ignored, one can use the Newey-West (Newey and West, 1988) covariance estimator $N(\mathbf{X}^T\mathbf{X})^{-1}\hat{\mathbf{Q}}(\mathbf{X}^T\mathbf{X})^{-1}$, in which the matrix $\hat{\mathbf{Q}}$, when applied to our problem, is estimated with

$$S_0 + \frac{1}{N}\sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{\ell=1}^{L}\sum_{t=\ell+1}^{H} w_\ell e_{\kappa(i,t)}e_{\kappa(j,t-\ell)}\left(\boldsymbol{\chi}_{\kappa(i,t)}\boldsymbol{\chi}_{\kappa(j,t-\ell)}^T + \boldsymbol{\chi}_{\kappa(j,t-\ell)}\boldsymbol{\chi}_{\kappa(i,t)}^T\right), \tag{3.3}$$

where the subscript $\kappa(i,t) = t + (i-1)H$, the weight $w_\ell = 1 - \frac{\ell}{L+1}$ and L is typically set such that $L \approx H^{1/4}$.

To summarize the results in Section 3, we show that if the true $\boldsymbol{\beta}$ values are sufficiently small, which can be enforced by restricting the discrete time unit to a small interval, the iterative descent approach always converges to a stationary point of $v(\boldsymbol{\beta})$. With the assumption of full rank (Proposition 3.1), the model is identifiable. Moreover, we show that our method yields an estimator that is consistent and asymptotically normal. We also show how the covariance matrix can be estimated

and corrected in the presence of heteroscedasticity and autocorrelation. In the next section, we apply this iterative descent approach to sales data that are stochastically generated from the model given by equation (2.12) and demonstrate its performance. We have applied both the Armijo rule and a "limited" minimization rule which searches along the direction d^k but within a bounded interval between $\boldsymbol{\beta}^k$ and $\mathbf{b}^k$. Both work well for the simulated data while the latter appears to work better for the Intel application. We present the results obtained with the limited minimization rule.

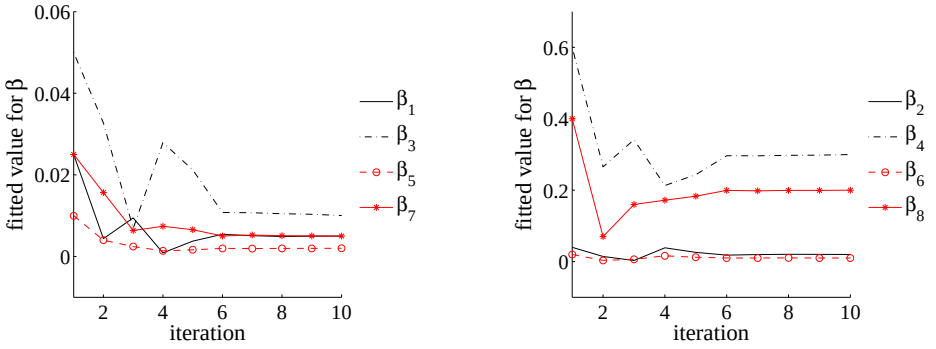
4 Performance on Simulated Demand Data

In this section, we assume that the underlying model is $\mathbf{s} = \mathbf{X}(\boldsymbol{\beta})\boldsymbol{\beta} + \boldsymbol{\epsilon}$ where $\boldsymbol{\epsilon}$ is normally distributed with zero mean. We assume that errors are uncorrelated but allow them to be heteroscedastic. For the ease of reference, denote the variance of observation i with σ_i^2 and define vector $\boldsymbol{\sigma} = (\sigma_1, \sigma_2, \ldots, \sigma_N)$. We apply the iterative descent procedure described in Section 3.2 to solve for the optimal $\boldsymbol{\beta}$ that minimizes the mean squared error.

4.1 Estimation and Fit

We present an example where sales are generated from a model described in equation (2.11) with $\boldsymbol{\beta} = (0.005, 0.02, 0.01, 0.30, 0.002, 0.01, 0.005, 0.2)$. All three sources of demand identified in Section 2 are present and we correct for a market expansion of 0.1% per period. At the start of the time horizon (time 50), there are four existing generations of products (sales of the first product has already dropped to zero) with known population sizes $0.0156, 0.0399, 0.2744, 0.1131$ and the competition population is 1.56. The focal firm introduces new product at a constant pace (one product every 20 time periods). We generate sales data during time $[50, 200]$ which includes sales of ten products, seven of which are new product introductions. The focal firm and competition start at about the same performance level. Product improvement is $10\%, 11\%, ..., 19\%$ (over the immediate predecessor) respectively for the ten products of the focal firm and a constant 5% for the competition (improvement occurs at the same time for both the focal firm and its competition). In this section, we choose the sales data for the time window $[50, 150]$ (which includes sales of three existing products and five new products) to estimate the parameters, assuming that we have an accurate estimate of the population mix at time 50. In Section 4.2, we use the remaining sales data (which includes sales of three existing products and two new products) to test forecast performance; we also vary the initial population sizes and the size of the fit/forecast window to test forecast sensitivity.

To validate the iterative procedure, we first test the case with $\boldsymbol{\sigma} = 0$ to confirm that it converges to the true parameters. See Figure 1. In the presence of noise ($\boldsymbol{\sigma} \neq 0$), the procedure in general does



(a) (b)

Figure 1: Convergence to the True Parameter Values

not converge to the true parameter value that is used to generate the sales, but rather the nonlinear least squares estimator of the model $\mathbf{s} = \mathbf{X}(\boldsymbol{\beta})\boldsymbol{\beta} + \boldsymbol{\epsilon}$, as expected. See Table 1 for three examples generated at different noise levels where we use $\boldsymbol{\sigma} = 5\%, 10\%, 20\%$ to loosely denote the case that the noise term $\boldsymbol{\epsilon}$ has a standard deviation that is $5\%, 10\%, 20\%$ of the mean sales, respectively. Note that "std. err." and "White err." are respectively the standard error of parameters without and with correcting for heteroscedasticity. Also, * and ** denote significance at the 90% and 95% level respectively, computed based on the White error estimate.

	$\boldsymbol{\sigma} = 5\%$			$\boldsymbol{\sigma} = 10\%$			$\boldsymbol{\sigma} = 20\%$		
Parameters	estimate	std. err.	White err.	estimate	std. err.	White err.	estimate	std. err.	White err.
β_1	0.0052**	0.0035	0.0007	0.0048**	0.0058	0.0012	0.0042**	0.0106	0.0024
β_2	0.0209**	0.0141	0.0026	0.0230**	0.0229	0.0043	0.0260**	0.0421	0.0088
β_3	0.0178**	0.0053	0.0030	0.0135**	0.0079	0.0041	0.0192**	0.0152	0.0073
β_4	0.2547**	0.0366	0.0183	0.2650**	0.0553	0.0259	0.2280**	0.1025	0.0439
β_5	0.0017**	0.0009	0.0003	0.0029**	0.0014	0.0006	0.0010	0.0032	0.0014
β_6	0.0110**	0.0047	0.0019	0.0039*	0.0075	0.0032	0.0109	0.0178	0.0090
β_7	0.0064**	0.0026	0.0009	0.0015	0.0042	0.0018	0.0059*	0.0095	0.0040
β_8	0.1889**	0.0163	0.0044	0.2237**	0.0260	0.0089	0.2583**	0.0619	0.0240

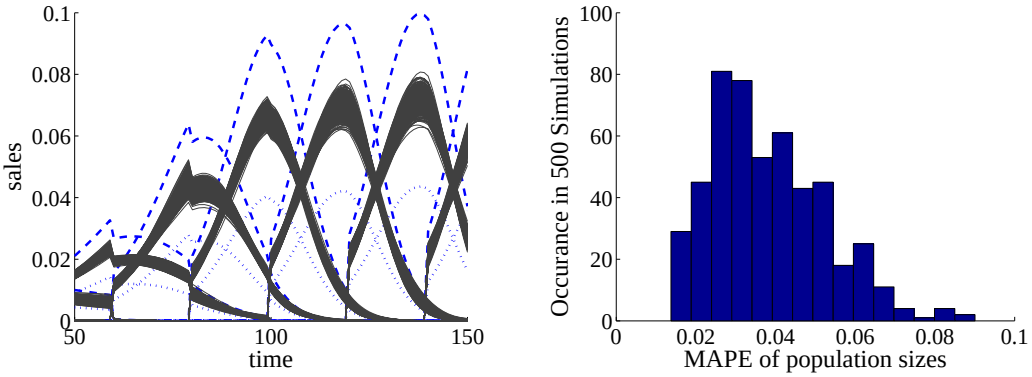
Table 1: Parameter Estimates and Errors for Various Levels of Noise

The statistical significance of the parameters drops as the uncertainty level increases. The smaller parameters tend to lose statistical significance fast (i.e., drop below 90% significance) with increasing uncertain levels. This suggests that, like most other methods, our model is not appropriate for drawing statistical inferences for data sets with high noise. However, as we show next in this section and Section 4.2, it works reasonably well for the purpose of fitting and forecasting

sales even at a high noise level.

To examine the model fit, we generate stochastic sales data for a given set of parameter values and apply the iterative descent procedure to obtain the corresponding parameter estimates. We then generate the fitted sales curve and compare it with the 95% confidence level sales band predicted by the true parameters. Figure 2(a) illustrates such a comparison. The true parameters for the underlying model are the same as in Figure 1. The noise term ($\boldsymbol{\epsilon}$) has a standard deviation that is 20% of the mean sales. (i.e., $\boldsymbol{\sigma}$ = 20% of mean sales). We generate 500 sets of sales data and apply the iterative descent method to each data set. We observe that the majority of the fitted curves (solid curves) centers around the "true" curve (the dashed curve on the top is the upper bound of the 95% confidence interval and the dotted curve on the bottom is the lower bound). More importantly, this observation stays true even as the uncertainty level increases: As the band for the fitted sales curve becomes wider, the confidence intervals of the true sales curve also broadens.

Further, we examine how well the true population variables $x_i(t)$ are recovered. We predict the population path from the parameter estimates for each of the 500 simulated data sets and compute the Mean Absolute Percentage Error (MAPE) of the population for each set of predictions. Figure 2(b) shows the error distribution of the predicted population sizes for $\boldsymbol{\sigma}$ = 20%. We observe that the iterative approach recovers the true population well (the population deviations are well within 10%, in most cases 3% or 4%).



(a) Fitted Sales Curves

(b) Population Deviation

Figure 2: Fitted Sales Curves and Population Deviation

4.2 Forecast Performance and Sensitivity

Next, we illustrate the forecast accuracy of the model using a portion of the data to calibrate the model and the remaining to test forecast performance. In particular, for each randomly generated

data set, we parameterize the model with sales during time period $[50, 150]$ and then forecast the sales for time $[151, 200]$. We compute forecast errors for each data set and Table 2 presents the mean forecast errors averaged over the 500 data sets: RMSE is the root mean squared error; MAE is the mean absolute error; MAPE is the mean absolute percentage error.

Because RMSE is the measure that least square regression optimizes, we use the RMSE (instead of other error measures) to compare the model fit and forecast performance of different methods. We present the MAE and MAPE to provide additional information.

We remark that the observations shown in Table 2 and Figure 3 both demonstrate that our model is well-behaved: in the majority of the simulated cases, the forecast largely replicates the inherent uncertainty of sales data generated at each noise level and we do not observe significant amplifications as noise level increases.

Error Measure	$\boldsymbol{\sigma} = 5\%$	$\boldsymbol{\sigma} = 10\%$	$\boldsymbol{\sigma} = 20\%$
average RMSE	0.0030	0.0054	0.0102
average MAE	0.0022	0.0038	0.0069
average MAPE	9.71%	15.36%	28.01%

Table 2: Forecast Performance Averaged over 500 Data Sets

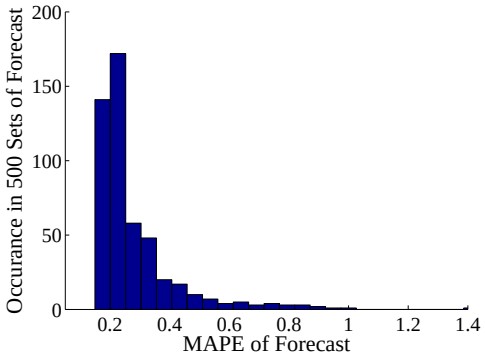


Figure 3: Forecast Error Distribution (s.d. of $\boldsymbol{\epsilon} = 20\%$)

Sensitivity to Initial Population

We have assumed that the initial population sizes are known. In the following, we experiment with cases where the company may over or under estimate these values. Table 3 illustrates forecast performance of the model when the population sizes are overestimated by 10% (case 1), underestimated by 10% (case 2), as well as in cases in which some products are overestimated while others are underestimated ("mixed") – in case 3, the first two products are overestimated by 10% and the next two products are underestimated by 10% whereas in case 4 the opposite is true. The results

are averaged over 100 data sets.

Case	base case	1(over 10%)	2(under 10%)	3(mixed)	4(mixed)
average RMSE	0.00531	0.00533	0.00535	0.00533	0.00531
average MAE	0.0038	0.0039	0.0038	0.0037	0.0038
average MAPE	15.10%	17.35%	14.48%	14.54%	16.31%

Table 3: Forecast Sensitivity to Initial Population ($\boldsymbol{\sigma} = 10\%$, fit/forecast window=$[50, 150]/[151, 200]$)

Performance resilience to perturbations in the initial population estimation is evident, which appears counterintuitive but is indeed an explainable characteristic of this method. Specifically, the method can self-correct with respect to errors in the initial population estimation: as time elapses and new sales occur, the memory of the initial population and its effect fades away rather quickly.

Sensitivity to Fit/Forecast Time Window

Table 4 demonstrates the sensitivity of forecast errors to the number of time periods (or similarly, the number of products) used to parameterize the model (data are generated with $\boldsymbol{\sigma} = 10\%$; forecasts are averaged over 100 random data sets). As expected, forecast accuracy decreases as the training set becomes smaller; nevertheless, one still obtains reasonable forecasts when the size of the training data is at least comparable to the test data.

fit/forecast window	$[50, 100]/[101, 150]$	$[50, 125]/[126, 175]$	$[50, 150]/[151, 200]$
average RMSE	0.0056	0.0052	0.0053
average MAE	0.0041	0.0038	0.0038
average MAPE	23.85%	16.72%	15.10%

Table 4: Sensitivity of Forecast Errors to the Number of Time Periods ($\boldsymbol{\sigma} = 10\%$)

Constrained versus Unconstrained Regression

We interpret the parameters $\boldsymbol{\beta}$ as nonnegative because the gap of product strength between product i and product j, Δf_{ij}, has a nonnegative effect on the transition parameters p_{ij} and q_{ij}. In the presence of noise, it is possible that an unconstrained linear regression may result in negative value(s) for some component(s) of the estimated parameter vector. Therefore, in our numerical studies, we have constrained the parameters to be nonnegative. Using unconstrained regression may allow one to achieve better fit to the training data; however, it inevitably leads to poorer forecasting performance. Table 5 demonstrates the fit and forecast performance of the constrained and unconstrained models (data are generated with $\boldsymbol{\sigma} = 10\%$; fit and forecast errors are averaged over 100 random data sets). The RMSE for the training data is roughly the same for the constrained and unconstrained cases, however, the forecast RMSE of the unconstrained model is worse than the

constrained model. Similar observations are made in the Intel application in Section 5. Therefore, one is better off taking advantage of the knowledge on the underlying dynamics by constraining the parameters to nonnegative values to avoid a false model which provides better fit but poor forecast.

	Constrained (Fit)	Constrained (Forecast)	Unconstrained (Fit)	Unconstrained (Forecast)
average RMSE	0.0034	0.0053	0.0034	0.0056
average MAE	0.0021	0.0038	0.0022	0.0040
average MAPE	11.57%	15.10%	13.52%	18.36%

Table 5: Fit and Forecast Comparison ($\boldsymbol{\sigma} = 10\%$, fit/forecast window=$[50, 150]/[151, 200]$)

In summary, the iterative descent approach works well for simulated data. In Section 5, we apply the proposed method to Intel data and compare against the Bass, Norton-Bass and Jun-Park models (such a comparison would not be appropriate in Section 4 since the simulated data is generated using our model).

5 Application to the Extreme Edition Microprocessor Market

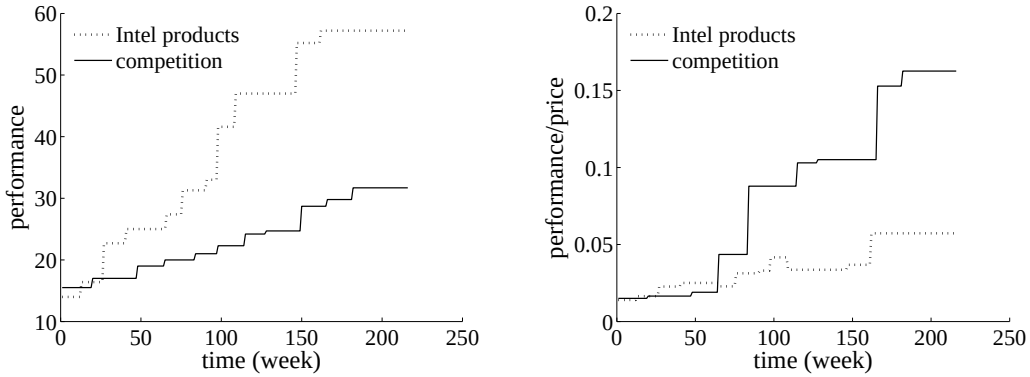
In this section, we apply our model to the sales data of Intel's "extreme edition" microprocessors. These high-end desktop processors are sold primarily to end users who participate in sophisticated computer games - the "extreme gamers." Therefore, this is also referred to as the extreme gamers market.

5.1 Data Description

Included in the data are the introduction date, performance score, and price for each product sold by Intel, as well as those for a major competitor. Weekly sales data cover a 4-year time window with a total of 11 products from Intel. (We include in the Online Appendix the masked data set, as well as a plot of the masked sales for Intel products).

Performance scores for both Intel and competition products are obtained from the Standard Performance Evaluation Corporation (SPEC) based on both integer and floating point benchmarks (SPEC Website, 2010), which are commonly-adopted industry standards. The extreme gamers market is generally perceived as price-insensitive. Nevertheless, we apply the method in two settings: one in which product strength is given by performance score alone, one in which product strength is given by performance/price ratio. The latter is commonly adopted at Intel as a measure of product strength for markets that are price-sensitive. We compare these two alternatives for the extreme gamers market. The data span a time period that is characterized by a performance race between Intel and its competitor with the performance gap between the two companies widening

rapidly over time despite the initial performance lead and steady improvement by the competitor (Figure 4(a)). Figure 4(b) shows the performance/price ratio of the two firms. The competitor priced its product much lower than Intel and was thus leading on performance/price ratio. In addition, the total market during this time was growing over time, reflecting a known market trend. As mentioned earlier, such trend is assumed to be known. Our focus is on how the total market is split between Intel and competition and among the products within Intel. We use 120 weeks of data to parameterize the model (which includes sales of nine products) and the remaining 96 weeks of data (which includes sales of six products, two of which introduced after week 120) to test the forecast accuracy of the proposed model. The forecast is based on a "frozen" fit sample of 120 weeks, which differs from the rolling-horizon forecast typically seen in the literature of diffusion models. The rolling-horizon forecast is unfit for strategic planning decisions, which are often made long before the actual sales.

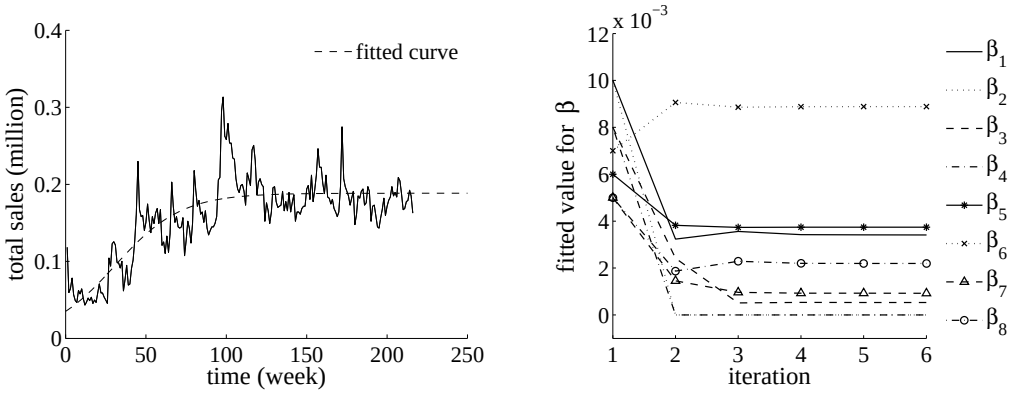


(a) Performance (b) Performance/Price

Figure 4: Performance and Performance/Price

First, we adjust for the aggregate market trend. Note that $\alpha(t)$ is the sales growth due to market expansion as a percentage of the most recent sales. Ideally we would use the original "forecasted" $\alpha(t)$ for this purpose. However, we do not have data of the forecasted sales growth for new expansion. Therefore, to calibrate the model, we use the actual sales (which is available) to estimate the growth trend of total sales, and use it as a proxy for the growth trend of new expansion. This is reasonable if the growth trend due to new market expansion and the growth trend of total sales follow similar pattern. For the extreme gamers market, the total market during this time window was on a logistic growth path. We thus fit a logistic growth curve to the estimated total sales and the growth rate $\alpha(t)$ is obtained through this fitted curve. Figure 5(a) shows the fitted logistic curve.

Before applying the iterative descent approach, we need to estimate the initial population mix. The extreme gamers market is initially dominated by the competition, so we set $x_i(0)$ to zero and $y(0)$ to the size of the initial market. Experts at Intel estimate this to be around 7.5 Million at the start of this time window. We show with additional sensitivity analysis that, the model is robust to fluctuations in this estimate due to the reasons discussed in Section 4. (see Tables 14 to 15 provided in the Online Appendix).



(a) Total Market (b) Convergence of Parameters

Figure 5: Application of Extreme Gamers Market

5.2 Estimation and Forecast

In addition to the two alternatives of product strength measure, namely, performance alone ("perf-only") and performance/price ratio ("perf/price"), we also explore two options for determining product strength of the competition, $f_y(t)$: one defined as the strength of the *strongest* product of the competition ("best-comp"), the other defined as the *mean* product strength of the competition ("mean-comp"). Moreover, as in Section 4, we examine the forecast performance *with* ("constrained") and *without* ("unconstrained") nonnegativity constraint on the parameters.

Figure 5(b) shows the convergence path of the parameters for the "constrained, perf-only, best-comp" case. Similar convergence pattern is observed for other cases.

Table 6 and Table 7 summarize the parameter estimates, standard errors, as well as the White and Newey-West errors obtained using sales data from the first 120 weeks (first 9 products). The model fit and forecast performances are reported in Tables 8 and 9. In addition to the RMSE, MAE, MAPE, we also compute the median absolute percentage error (MdAPE), MAPE of the cumulative sales (cumMAPE), MAPE of the peak sales (peakMAPE), mean absolute error of peak time (timeMAE), as well as the R^2 value. (Note that for nonlinear regression, R^2 value is

not necessarily between 0 and 1.) Again, * and ** denote significance at the 90% and 95% level respectively, computed based on the White error estimate. With the exception of peakMAPE and timeMAE, errors are averaged over all available data points (which includes sales for each product at each time period during its selling window). To compute the error for model fit (Table 8), we use data points within the first 120 weeks; to compute the forecast error (Table 11), we use data points in the 96 weeks starting week 121. Due to the life-cycle effect, predicted and actual sales may differ by a small *absolute* amount but a high *percentage* amount near the tail regions of each product. This is particularly true in a real data application, in which the predicted and the actual sales window may differ significantly near the tails (possibly due to companies enforcing end-of-life for a product instead of letting it follow the course of a diffusion); thus for the Intel application we also report the MdAPE, which partially offsets this problem. Each Intel product has a sales peak in its life cycle and peakMAPE and timeMAE respectively measure error in the peak volume and error in the peak time of the products, which are important for Intel because the company needs to plan capacity appropriately to accommodate peak sales. These values are averaged over the number of products (for the training data, averaged over all products which peaked during the first 120 weeks; for the test data, averaged over all products which peaked after week 120).

	Parameters	β_1	β_2	β_3	β_4	β_5	β_6	β_7	β_8
Perf only	Estimate	0.0034^{*}	0	0.00045	0	0.0037^{**}	0.0089^{*}	0.00091	0.0022
	std. err.	0.0011	0.0019	0.0012	0.0038	0.0008	0.0028	0.0008	0.0041
	White err.	0.0025	0.0032	0.0018	0.0035	0.0015	0.0059	0.0010	0.0055
	Newey-West err.	0.0048	0.0073	0.0038	0.0066	0.0025	0.0092	0.0016	0.0077
Perf Price Ratio	Estimate	0.0150^{**}	0	0.0310^{**}	0.0806^{**}	0.0039^{**}	0.0045	0	0
	std. err.	0.0020	0.0095	0.0061	0.0298	0.0012	0.0035	0.0028	0.0093
	White err.	0.0021	0.0088	0.0052	0.0268	0.0018	0.0062	0.0027	0.0097
	Newey-West err.	0.0033	0.0124	0.0075	0.0432	0.0030	0.0092	0.0043	0.0144

Table 6: Parameter Estimates for the Extreme Gamers Market ("constrained, best-comp")

	Parameters	β_1	β_2	β_3	β_4	β_5	β_6	β_7	β_8
Perf only	Estimate	0.0092^{*}	-0.0135	-0.0059	0.0180^{*}	0.0013	0.0124^{*}	0.0014^{*}	0.0067^{**}
	std. err.	0.0011	0.0018	0.0011	0.0034	0.0008	0.0025	0.0008	0.0037
	White err.	0.0027	0.0035	0.0019	0.0028	0.008	0.0047	0.0008	0.0053
	Newey-West err.	0.0046	0.0063	0.0032	0.0048	0.0011	0.0067	0.0012	0.0076
Perf Price Ratio	Estimate	0.0113^{*}	-0.0091	0.0105^{*}	0.0202^{*}	0.0001	0.0179^{*}	0.0031^{*}	-0.0062
	std. err.	0.0013	0.0039	0.0025	0.0093	0.0010	0.0030	0.0019	0.0068
	White err.	0.0017	0.0034	0.0029	0.0095	0.0010	0.0057	0.0018	0.0092
	Newey-West err.	0.0028	0.0052	0.0052	0.0161	0.0013	0.0075	0.0022	0.0116

Table 7: Parameter Estimates for the Extreme Gamers Market ("unconstrained, best-comp")

As discussed earlier, we use the RMSE to compare model fit and forecast performance because our method is a least squares method. We make several observations from Tables 8 and 9. First,

Method	RMSE	MAE	MAPE	MdAPE	cum APE	peak MAPE	time MAE	R^2
constrained, perf-only, best-comp	0.0176	0.0121	67%	41%	49%	3.44%	5.4	0.40
constrained, perf/price, best-comp	0.0202	0.0153	74%	82%	42%	2.93%	6.3	0.22
unconstrained, perf-only, best-comp	0.0163	0.0115	62%	42%	42%	2.94%	5.3	0.49
unconstrained, perf/price, best-comp	0.0183	0.0138	70%	64%	55%	3.35%	7.1	0.35
constrained, perf-only, mean-comp	0.0182	0.0128	125%	41%	81%	3.51%	5.7	0.36
constrained, perf/price, mean-comp	0.0193	0.0149	122%	69%	75%	3.15%	8.1	0.28

Table 8: Comparison of Model Fit

Method	RMSE	MAE	MAPE	MdAPE	cum MAPE	peak MAPE	time MAE
constrained, perf-only, best-comp	0.0111	0.0089	94%	38%	61.0%	0.69%	13.7
constrained, perf/price, best-comp	0.0154	0.0131	78%	88%	67.0%	1.90%	10.7
unconstrained, perf-only, best-comp	0.0167	0.0132	128%	60%	71.6%	0.24%	31.3
unconstrained, perf/price, best-comp	0.0333	0.0254	187%	94%	150.1%	4.08%	10.0
constrained, perf-only, mean-comp	0.0113	0.0090	95%	39%	62.7%	0.63%	13.7
constrained, perf/price, mean-comp	0.0140	0.0119	84%	85%	66.3%	0.89%	12.0

Table 9: Comparison of Forecast Performance

although the unconstrained regression yields better RMSE in the training data, the forecast RMSE is much worse than the constrained regression, indicating that the constrained regression is more appropriate. This is consistent with the findings from the test on the simulated data. Second, whether one defines $f_y(t)$ as the strength of the strongest competing product or as the mean product strength of competition at time t does not lead to dramatic changes in the forecast performance, with the former performing slightly better (smaller RMSE). This could be easily explained by the fact that the latter is roughly a moving average of the former and, in the absence of drastic trend change in the product strength of the competition, one should not expect major differences. Lastly, we observe that the performance-only alternative generally achieves better results. This is consistent with our earlier expectation that customers in the "extreme" edition processor market are less sensitive to price changes. To further verify that it is safe not to consider price in this particular application, we also test the model by incorporating a separate price term into product strength, i.e., we let f_{ij} be a linear combination of performance improvement and price improvement from product i to product j. The best fit is obtained with zero weight on price (see more details in the Online Appendix). Summarizing the above evaluation, the "constrained, perf-only, best-comp" model specification demonstrates overall better performance and we recommend this model specification for the high-end gamer market. We focus our discussion in the remainder of the paper on this specification.

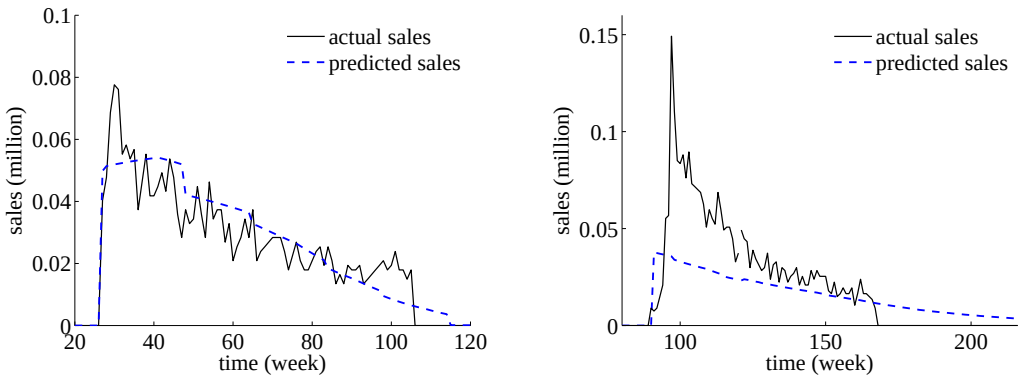
Parameter and Forecast Interpretation

As discussed in Section 4.1, the parameters obtained using our method shows low statistical significance at high noise levels. In the Intel application, for the "constrained, perf-only, best-comp" model, only β_1, β_5, and β_6 are above the 90% significance level, therefore, one cannot make definitive parameter inferences since it is difficult to establish whether a low significance level is due to low influence or high noise in the data. Nonetheless, the higher significance level of β_6 than β_2 seems to suggest that the influence of performance improvement on the *innovators* may be stronger for between-brand switching than for within-brand upgrades. One plausible interpretation is that for within-brand product upgrades, customers are less concerned with the exact performance improvement, as long as Intel passes some expected improvement hurdle (which Intel delivers as shown in Figure 4(a)). However, this is not the case for between-brand switchings. Since processors of different brands are not compatible, switching brand means replacing many other major components in the computer, or buying an entire new system. Therefore, for a customer to switch from one brand to another, he/she needs to be convinced that the performance improvement justifies this move. As a result, we observe stronger influence of performance on brand switchings than on within-brand upgrades.

Figure 6 shows the predicted and actual sales for the product with the best (product 3) and worst (product 7) fit in terms of RMSE. Note that the predicted sales before time 120 are fitted values and sales after time 120 are forecast. The fitted sales curves largely replicate the asymmetric pattern of the recorded sales (i.e., sales usually peak early in a product's life cycle, shown by the left-skewed sales curve), as well as the relative magnitude of sales across products, with the exception of product 7 in Figure 6(b). Close examination of the data reveals that even though product 7 only has a marginal performance improvement over its predecessor, it marks a major silicon technology shift and customers' purchasing decision might be influenced by factors not captured in the performance data, which may help explain the dramatic sales spike in spite of the performance. To maintain simplicity and generality, we did not specifically account for this effect, although adjustment for such technology shift should certainly improve the fit.

5.3 Comparison to Existing Models

Lastly, we compare the forecast performance of our method with the Bass, Norton-Bass and Jun-Park models. As discussed previously, these models require product-specific parameters and are difficult to apply in long-range forecasting. In order to use them and draw a comparison, we let the market potential parameters be linearly-dependent on product strength. The diffusion parameters are assumed to be the same across all the products following the argument made in Norton and



(a) Product 3 (b) Product 7

Figure 6: Actual vs. Predicted Sales

Bass (1987). In addition, in the Jun-Park model, we let the linear coefficient which describes how fast the utility for each product grows with time be dependent on product strength as well. We use generalized versions of these models and in the absence of such linear dependence, they simply reduce to the original version in the literature.

Tables 10 and 11 illustrate the performance of each method. Our method outperforms the Bass, Norton-Bass and Jun-Park models in terms of the RMSE and also along most other dimensions that Intel is interested in. The ability of our model to capture both product upgrades and brand switching has clearly contributed to its improvement over existing models, making it more suitable for long-range forecasting in a market with frequent product introductions and competition. More importantly, we note that our approach accomplishes this without using actual sales data from the competition.

Method	RMSE	MAE	MAPE	MdAPE	cum MAPE	peak MAPE	time MAE	R^2
Iterative Descent (perf only)	0.0176	0.0121	67%	41%	49%	3.44%	5.4	0.40
Bass	0.0243	0.0201	518%	178%	167%	4.14%	20.7	−0.14
Norton-Bass	0.0217	0.0169	282%	137%	129%	3.66%	11.3	0.09
Jun-Park	0.0227	0.0174	383%	133%	120%	4.50%	4.8	0.004

Table 10: Comparison of Model Fit

Method	RMSE	MAE	MAPE	MdAPE	cumMAPE	peakMAPE	timeMAE
Iterative Descent (perf only)	0.0111	0.0089	94%	38%	61%	0.69%	13.7
Bass	0.0208	0.0195	182%	141%	141%	0.58%	13.7
Norton-Bass	0.0645	0.0259	188%	191%	190%	3.00%	9.0
Jun-Park	0.0141	0.0122	129%	95%	92%	0.60%	14.0

Table 11: Comparison of Forecast Performance

Lastly, we remark that the forecast errors in the Intel application are high and the same level

of error might be unacceptable for a one-step ahead or two-step ahead forecast, which is often used in the empirical applications of the Norton-Bass model or the Jun-Park model. However, for long-range forecast of products to be released years later, and given the high uncertainties inherent to this market and the fact that we forecast *weekly* sales instead of quarterly (e.g., Norton and Bass 1987) or yearly (e.g., Mahajan and Muller 1996) sales, high forecast uncertainty is expected. More importantly, high uncertainty does not diminish the value of a forecast. To the contrary, being able to characterize the demand and corresponding uncertainty enables the company to appropriately hedge against future risk. For example, Intel has implemented a method to use the knowledge of future demand uncertainty to optimally design option contracts with its equipment suppliers in order to reduce excess capacity (Peng et al. 2012, Kempf et al. 2013).

5.4 Sensitivity to Fit/Forecast Time Window

We have shown in Section 4.2 that, the forecast accuracy decreases with the amount of historical data used for parametrization. For the Intel data set, we perform a similar sensitivity analysis. We vary the size of the fit sample among $100, 120$ and 140 weeks while fixing the forecast window to the next immediate 76 weeks. We also change the forecast window while fixing the fit window to 120 weeks. Tables 12 and 13 illustrate how the forecast accuracy changes with these variations.

fit/forecast window	$[1, 100]/[101, 176]$	$[1, 120]/[121, 196]$	$[1, 140]/[141, 216]$
RMSE	0.0308	0.0116	0.0098
MAPE	199%	103%	93%
MdAPE	85%	41%	35%

Table 12: Sensitivity of Forecast Errors to the Amount of Historical Data

forecast window	$[121, 140]$	$[121, 160]$	$[121, 180]$	$[121, 200]$	$[121, 216]$
RMSE	0.0114	0.0108	0.0119	0.0115	0.0111
MAPE	59%	71%	106%	101%	94%
MdAPE	31%	30%	37%	41%	38%

Table 13: Sensitivity of Forecast Errors to the Forecast Window (Fit Window = $[1, 120]$)

It appears that increasing the fit window slightly improves the forecast accuracy, while reducing it lowers the accuracy dramatically. As we fix the fit window and vary the forecast window, the forecast accuracy changes significantly but there does not appear to be any clear trend. While it may be tempting to make inferences from the above, any trend (or the lack of it) observed from a single data set could be anecdotal to this particular data set. The trend derived from the simulated data might be more general since it is averaged over 100 data sets.

In summary, the application of the proposed method on the sales of Intel's high-end gamers market demonstrates some appealing features as well as some less satisfying aspects: It requires no data on the unit-in-use population (compared to conventional population-growth models) and accounts for the effect of competition. It converges quickly and compared to alternative methods can be more easily applied to long-range forecasting. However, at a high noise level, low statistical significance of the parameters precludes conclusive inferences of the causal effects and the forecast accuracy is sensitive to the size of the training data set. Since results from a single data set could be anecdotal, more real data applications by researchers and practitioners are required for further evaluation of the method.

6 Conclusion

We have proposed a method for parameterizing and forecasting the demand for multiple successive generations of products. Our model is based on a population growth model. The usual application of a population growth model requires data on the population size. However, for many companies who can not track the "units-in-use" for each product, a direct application of the growth model is not possible. We overcome this difficulty by an iterative approach that constructs the units-in-use population for each product based on current parameter estimates, and then use the constructed population size and sales observations to improve the parameter estimates. We show that this method is theoretically sound so long as we restrict the discrete time interval to a small value. The method performs well for the synthetic data, and outperforms other available methods when applied to the sales data of Intel's high-end microprocessors.

Using synthetic data, we test sensitivity of the method (model fit and forecast performance) against noise levels, sample size (the number of time periods or products), as well as perturbations in the initial population sizes, and show that the model is well-behaved. In the Intel application, we test several alternate model specifications based on how product strength is determined, how the competition strength is determined, and whether or not to restrict the parameters to be non-negative. In particular, we use two alternative specifications for product strength in the extreme gamers market – performance, and performance/price ratio. The former fits the data better and also shows better forecast accuracy, which is most likely due to the low price-sensitivity of this high-end market where processors cost 5 to 10 times time that of a mainstream processor product. In addition, the dependence on product performance appears to be stronger for brand switchings than for the within-brand product upgrades, in terms of the *innovator* effect. This is intriguing,

however, the special characteristics of the extreme gamers market may help explain the observation.

The parametrization method we propose is a nonlinear least squares method and therefore it suffers from some common limitations of nonlinear regression: the optimization may yield a local optimum (as the objective function is a high-order polynomial which may have multiple roots); if the noise level in the data is high, convergence may be slow and errors in the parameter estimates will be high. A limitation unique to our approach is the restriction to small coefficients. This is driven by (i) the need to ensure nonnegative and finite population size in the discrete time model, and (ii) exploitation of the asymptotic result for convergence in Proposition 3.5. In both (i) and (ii), it is difficult to define *a priori* an analytical bound for $\boldsymbol{\beta}$. However, when applying the method, it is easy to spot cases when $\boldsymbol{\beta}$ is chosen to be too large (one either encounters negative or extremely high population sizes, or observes oscillation instead of convergence). Therefore, lack of an analytical bound does not become an impediment for implementation. In practice, small $\boldsymbol{\beta}$ can be enforced by restricting the time unit to a small interval. This, however, may place a higher requirement on the temporal granularity of the data. With smaller time unit, the noise in the data may become larger. Hence, there is a tradeoff and identifying the optimal data collection time interval may be a trial-and-error process specific to each application.

Moreover, we have made several assumptions in this multi-generation diffusion model, some of which are similar to the Bass model; others pertain to the inter-generational effect: (i) the products are consumer durables and each customer purchases only one unit, i.e., there is no repeat purchase of the same product; (ii) we assume that customers only buy a product that is newer than the one that is currently owned; (iii) customers switch from and to competitors' products only if doing so results in a performance increase; (iv) the initial population mix at the start of the forecast window and the trend of the *total* market growth and/or decline can be estimated with high confidence. Clearly these assumptions do not hold in every industry and for every company. Lastly, as mentioned earlier, our approach targets multiple product generations with incremental positive improvement between successive generations, and does not apply to the diffusion of a single radically innovative product such as the first-ever electric refrigerator.

We show in this paper how this demand model can be parameterized and employed at Intel as an input to long-range planning in the extreme gamers market. To expand to other markets, the model may need to be adjusted. For example, the market for servers is also known to be relatively insensitive to price, like the extreme gamers market. But the market for mainstream consumer laptops is very sensitive to price fluctuations. Nonetheless, it does not require fundamental model change and one may simply extend the interpretation of "product strength" to a measure appro-

priate for the specific market. In the case of multi-dimensional product strength, for example, with price being one of the dimensions in addition to performance, there may be additional data and parameter-related requirements: with one more dimension in the product strength measure, the number of coefficients increases from eight to twelve, thus additional data may be needed to ensure identification. Also, convergence could become more difficult since our method requires all coefficient to be small and this is harder to ensure with more parameters. This problem could, however, be remedied by more careful scaling of the performance and price data. Although the model is motivated and developed based on a particular company, coexistence of multiple generations of products is common in the technology industry and we hope that researchers and practitioners may find it useful in other contexts as well.

Acknowledgement

We are grateful to Dr. Jay Schwarz from Intel Corporation for sharing his expertise, and to Professor Stephen Graves for his helpful comments. We also thank EIC Alan Scheller-Wolf, the associate editor, and three anonymous reviewers for their constructive suggestions. Dieter Armbruster's research was partially supported by a grant from the Volkswagen Foundation under the program on complex networks.

References

Amemiya, T. 1985. *Advanced econometrics, page 256.* Cambridge, Massachusetts: Harvard University Press.

Bass, F. 1969. A new product growth model for consumer durables. *Management Science* 15:215–227.

Bass, F. 1980. The relationship between diffusion rates, experience curves, and demand elasticities for consumer durable technological innovations. *The Journal of Business* 53 (3): S51–S67.

Bass, F., T. Krishnan, and D. Jain. 1994. Why the Bass model fits without decision variables. *Marketing Science* 13 (3): 203–223.

Bayus, B., N. Kim, and A. Shocker. 2000. Growth models for multiproduct interactions: current status and new directions. In *New-Product Diffusion Models*, ed. V. Mahajan, E. Muller, and Y. Wind, 141–163. Kluwer Academic Publishers.

Bertsekas, D. P. 2003. *Nonlinear programming.* Belmont, MA: Athena Scientific.

Danaher, P. J., B. G. S. Hardie, and W. P. J. Putsis. 2001. Marketing-mix variables and the diffusion of successive generations of a technological innovation. *Journal of Marketing Research* 38 (4): 501–514.

Debo, L., L. Toktay, and L. Van Wassenhove. 2006. Joint life-cycle dynamics of new and remanufactured products. *Production and Operations Management* 15 (4): 498–513.

Dempster, A., N. Laird, and D. Rubin. 1977. Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society, Series B Methodological* 39 (1): 1–38.

Fisher, J., and R. Pry. 1971. A simple substitution model of technological change. *Technological Forecasting and Social Change* 3:75–88.

Gordon, B. R. 2009. A dynamic model of consumer replacement cycles in the PC processor industry. *Marketing Science* 28 (5): 846–867.

Gowrisankaran, G., and M. Rysman. 2009. Dynamics of Consumer Demand for New Durable Consumer Goods. *Working paper, Boston University*.

Greene, W. 2003. *Econometric analysis*. New Jersey: Prentice Hall.

Ho, T.-H., S. Savin, and C. Terwiesch. 2002. Managing demand and sales dynamics in new product diffusion under supply constraint. *Management Science* 48 (2): 187–206.

Jain, D. C., and R. C. Rao. 1990. Effect of price on the demand for durables:modeling, estimation, and findings. *Journal of Business Economics and Statistics* 8 (2): 163–170.

Jun, D. B., and Y. S. Park. 1999. A choice-based diffusion model for multiple generations of products. *Technological Forecasting and Social Change* 61:45–58.

Kalish, S. 1985. A new-product adoption model with price, advertising, and uncertainty. *Management Science* 31 (12): 1569–1585.

Kamakura, W., and S. Balasubramanian. 1988. Long-term view of the diffusion of durables: a study of the role of price and adoption influence process via tests of nested models. *International Journal of Research in Marketing* 5:1–13.

Kempf, K., F. Erhun, E. Hertzler, T. Rosenberg, and C. Peng.. 2013. Optimizing Capital Investment Decisions at Intel Corporation. *Interfaces* 43 (1).

Kim, N., D. Chang, and A. Shocker. 2000. Modeling inter-category dynamics for a growing information technology industry. *Management Science* 46 (4): 496–512.

Kumar, S., and J. Swaminathan. 2003. Diffusion of innovations under supply constraints. *Operations Research* 51 (6): 866–879.

Libai, B., E. Muller, and R. Peres. 2009. The influence of within-brand and cross-brand word of mouth on the growth of competitive markets. *Journal of Marketing, 73(2), 19-34.* 73 (2): 19–34.

Mahajan, V., and E. Muller. 1996. Timing, diffusion, and substitution of successive generations of technological innovations: the IBM mainframe case. *Technological Forecasting and Social Change* 51:109–132.

Meade, N., and T. Islam. 2006. Modelling and forecasting the diffusion of innovation A 25-year review. *International Journal of Forecasting* 22:519–545.

Melnikov, O. 2001. Demand for Differentiated Durable Products: The Case of the U.S. Computer Printer Industry. *Mimeo, Yale University*.

Murray, J. D. 2002. *Mathematical biology : An introduction*. New York: Springer-Verlag.

Newey, W., and K. West. 1988. A simple positive semi-definite, heteroscedasticity and correlation consistent covariance matrix. *Econometrica* 55:703–708.

Norton, J., and F. Bass. 1987. A diffusion theory model of adoption and substitution for successive generations of high- technology products. *Management Science* 33 (9): 1069–1086.

Peng, C., F. Erhun, E. Hertzler, and K. Kempf. 2012. Capacity Planning in the Semiconductor Industry: Dual-Mode Procurement with Options. *Manufacturing and Service Operations Management* 14 (2): 170–185.

Peres, R., E. Muller, and V. Mahajan. 2010. Innovation diffusion and new product growth models: A critical review and research directions. *International Journal of Research in Marketing* 29:91–106.

Pielou, E. 1977. *Mathematical ecology*. New York: Wiley.

Rothenberg, T. J. 1971. Identification in parametric models. *Econometrica* 39 (3): 577–591.

Savin, S., and C. Terwiesch. 2005. Optimal product launch times in a duopoly: Balancing life-cycle revenues

with product cost. *Operations Research* 53 (1): 26–47.

Shenoy, S. R., and A. Daniel. 2006. Intel architecture and silicon cadence: the catalyst for industry innovation. *ftp://download.intel.com/software/pdf/IAandSiliconCadence.pdf* .

Song, I., and P. Chintagunta. 2003. A Micromodel of New Production Adoption with Heterogeneous and Forward-Looking Consumers: Application to the Digital Camera Category. *Quantitative Marketing and Economics* 1:371–407.

SPEC Website 2010. Published Online http://www.spec.org/benchmarks.html.

Tuma, N., and M. Hannan. 1984. *Social dynamics: Models and methods*. New York: Academic Press.

White, H. 1980. A heteroscedasticity-consistent covariance matrix estimator and a direct test for heteroscedasticity. *Econometrica* 48:817–838.

Wu, S. D., K. Kempf, M. Atan, B. Aytac, S. Shirodkar, and A. Mishra.. 2010. Improving New-Product Forecasting at Intel Corporation. *Interfaces* 40 (5): 385–396.

A Population-Growth Model for Multiple Generations of Technology Products

Hongmin Li • Dieter Armbruster • Karl G. Kempf

A Online Appendix

A.1 Equations of Population Evolution and Competition Sales

$$\begin{aligned} x_i(t+1) &= x_i(t) + \beta_1 \left[\sum_{j<i} x_j(t) - \sum_{j>i} x_i(t)\right] + \beta_2 \left[\sum_{j<i} f_{ji}x_j(t) - \sum_{j>i} f_{ij}x_i(t)\right] \\ &\quad + \beta_3 \left[\sum_{j<i} x_i(t)x_j(t) - \sum_{j>i} x_i(t)x_j(t)\right] + \beta_4 \left[\sum_{j<i} f_{ji}x_i(t)x_j(t) - \sum_{j>i} f_{ij}x_i(t)x_j(t)\right] \\ &\quad + \beta_5 \left[y(t)I(i\in J) - x_i(t)I(i\in \bar{J})\right] + \beta_6 \left[f_{yi}y(t)I(i\in J) - f_{iy}x_i(t)I(i\in \bar{J})\right] \\ &\quad + \beta_7 \left[x_i(t)y(t)(I(i\in J) - I(i\in \bar{J}))\right] + \beta_8 \left[x_i(t)y(t)(f_{yi}I(i\in J) - f_{iy}I(i\in \bar{J}))\right] \\ &\quad + \alpha(t)s_i(t-1)\ , \end{aligned} \tag{A.1}$$

$$\begin{aligned} y(t+1) &= y(t) + \beta_5 \left[-\sum_{i\in J} y(t) + \sum_{i\in \bar{J}} x_i(t)\right] + \beta_6 \left[-\sum_{i\in J} f_{yi}y(t) + \sum_{i\in \bar{J}} f_{iy}x_i(t)\right] \\ &\quad + \beta_7 \left[-\sum_{i\in J} x_i(t)y(t) + \sum_{i\in \bar{J}} x_i(t)y(t)\right] + \beta_8 \left[-\sum_{i\in J} f_{yi}x_i(t)y(t) + \sum_{i\in \bar{J}} f_{iy}x_i(t)y(t)\right] \\ &\quad + \alpha(t)s_y(t-1)\ , \end{aligned} \tag{A.2}$$

$$\begin{aligned} s_y(t) &= \beta_5 \left[\sum_{i\in \bar{J}} x_i(t)\right] + \beta_6 \left[\sum_{i\in \bar{J}} f_{iy}x_i(t)\right] + \beta_7 \left[\sum_{i\in \bar{J}} x_i(t)y(t)\right] + \beta_8 \left[\sum_{i\in \bar{J}} f_{iy}x_i(t)y(t)\right] \\ &\quad + \alpha(t)s_y(t-1)\ . \end{aligned} \tag{A.3}$$

A.2 Proof of Lemma 3.2

Proof. For brevity we omit the argument $\boldsymbol{\beta}^k$ of $\mathbf{X}$ in the proof. Since $\mathbf{X}$ is full rank, $\mathbf{X}^T\mathbf{X}$ is invertible. Thus we can rewrite $\mathbf{d}^k$ as

$$\begin{aligned} \mathbf{d}^k &= (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{X}[(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{s} - \boldsymbol{\beta}^k] \\ &= (\mathbf{X}^T\mathbf{X})^{-1}(\mathbf{X}^T\mathbf{s} - \mathbf{X}^T\mathbf{X}\boldsymbol{\beta}^k) = -(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T(\mathbf{X}\boldsymbol{\beta}^k - \mathbf{s})\ . \end{aligned}$$

From the definition of $v(\boldsymbol{\beta})$, we have

$$\nabla v(\boldsymbol{\beta}^k) = 2[\nabla(\mathbf{X}\boldsymbol{\beta}^k)](\mathbf{X}\boldsymbol{\beta}^k - \mathbf{s})\ .$$

Thus

$$\begin{aligned} \frac{1}{2}[\nabla v(\boldsymbol{\beta}^k)]^T\mathbf{d}^k &= -\left\{[\nabla(\mathbf{X}\boldsymbol{\beta}^k)](\mathbf{X}\boldsymbol{\beta}^k - \mathbf{s})\right\}^T \left[(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T(\mathbf{X}\boldsymbol{\beta}^k - \mathbf{s})\right] \\ &= -(\mathbf{X}\boldsymbol{\beta}^k - \mathbf{s})^T[\nabla(\mathbf{X}\boldsymbol{\beta}^k)]^T(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T(\mathbf{X}\boldsymbol{\beta}^k - \mathbf{s}) < 0\ , \end{aligned}$$

where the last inequality holds because $[\nabla(\mathbf{X}\boldsymbol{\beta}^k)]^T(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T$ is positive definite.

We have made the assumption earlier that $x_i(t)$ is always nonnegative and bounded from above. Therefore, as long as we start with a $\boldsymbol{\beta}^k$ that is bounded, $\mathbf{b}^k = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{s}$ is bounded; and thus β^{k+1} is bounded. As a result, the sequence $\{\mathbf{d}^k\}$ is bounded and $\{\mathbf{d}^k\}$ is gradient related to $\{\boldsymbol{\beta}^k\}$. □

A.3 Proof of Proposition 3.5

Proof. From equation (A.1), we have:

$$\begin{aligned}
\frac{\partial x_i(t+1)}{\partial \beta_1} &= \frac{\partial x_i(t)}{\partial \beta_1} + \sum_{j<i} x_j(t) - \sum_{j>i} x_j(t) + \beta_1 \frac{\partial}{\partial \beta_1}\left[\sum_{j<i} x_j(t) - \sum_{j>i} x_j(t)\right] \\
&+ \beta_2 \frac{\partial}{\partial \beta_1}\left[\sum_{j<i} f_{ji}x_j(t) - \sum_{j>i} f_{ij}x_i(t)\right] + \beta_3 \frac{\partial}{\partial \beta_1}\left[\sum_{j<i} x_i(t)x_j(t) - \sum_{j>i} x_i(t)x_j(t)\right] \\
&+ \beta_4 \frac{\partial}{\partial \beta_1}\left[\sum_{j<i} f_{ji}x_i(t)x_j(t) - \sum_{j>i} f_{ij}x_i(t)x_j(t)\right] + \beta_5\left[\frac{\partial y(t)}{\partial \beta_1} I(i \in J) - \frac{\partial x_i(t)}{\partial \beta_1} I(i \in \bar{J})\right] \\
&+ \beta_6\left[\frac{\partial y(t)}{\partial \beta_1} f_{yi} I(i \in J) - \frac{\partial x_i(t)}{\partial \beta_1} f_{iy} I(i \in \bar{J})\right] + \beta_7 \frac{\partial x_i(t)y(t)}{\partial \beta_1}(I(i \in J) - I(i \in \bar{J}))] \\
&+ \beta_8 \frac{\partial x_i(t)y(t)}{\partial \beta_1}[f_{yi} I(i \in J) - f_{iy} I(i \in \bar{J})] + \alpha(t)\frac{\partial s_i(t-1)}{\partial \beta_1} .
\end{aligned}$$

Since $x_i(t)$ $y(t)$, and $s_i(t)$ are bounded $\forall i, t$, it is easy to show by induction that $\frac{\partial x_i(t)}{\partial \beta_1}$, $\frac{\partial y(t)}{\partial \beta_1}$ and $\frac{\partial s_i(t)}{\partial \beta_1}$ are bounded $\forall i, t$. Since $\frac{\partial x_i(t)x_j(t)}{\partial \beta_1} = x_i(t)\frac{\partial x_j(t)}{\partial \beta_1} + x_j(t)\frac{\partial x_i(t)}{\partial \beta_1}$ and $\frac{\partial x_i(t)y(t)}{\partial \beta_1} = x_i(t)\frac{\partial y(t)}{\partial \beta_1} + y(t)\frac{\partial x_i(t)}{\partial \beta_1}$, it follows that $\frac{\partial x_i(t)x_j(t)}{\partial \beta_1}$ and $\frac{\partial x_i(t)y(t)}{\partial \beta_1}$ are also bounded. We can show similarly that $\frac{\partial x_i(t)}{\partial \beta_m}$, $\frac{\partial y(t)}{\partial \beta_m}$, $\frac{\partial s_i(t)}{\partial \beta_m}$, $\frac{\partial x_i(t)x_j(t)}{\partial \beta_m}$, and $\frac{\partial x_i(t)y(t)}{\partial \beta_m}$ where $m = 2, \ldots, 8$ are bounded. Consequently, $\lim_{\beta \to 0}(\nabla_\beta X)\beta \to 0$. Therefore,

$$\begin{aligned}
\lim_{\beta \to 0}[\nabla(\mathbf{X}\beta)]^T(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T &= \lim_{\boldsymbol{\beta} \to 0}[\mathbf{X} + (\nabla\mathbf{X})\boldsymbol{\beta}](\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T \\
= \lim_{\boldsymbol{\beta} \to 0}\mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T + \lim_{\boldsymbol{\beta} \to 0}(\nabla\mathbf{X})\boldsymbol{\beta}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T &= \lim_{\boldsymbol{\beta} \to 0}\mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T ,
\end{aligned}$$

where the last equality follows from $\lim_{\boldsymbol{\beta} \to 0}(\nabla_{\boldsymbol{\beta}}\mathbf{X})\boldsymbol{\beta} \to 0$. Since $\mathbf{X}$ is full rank, the term $\lim_{\boldsymbol{\beta} \to 0}\mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T$ is positive definite. (To see that $\mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T$ is positive definite, consider any $\mathbf{a} \neq 0$. Define $\mathbf{b} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{a}$; thus $\mathbf{a} = \mathbf{X}\mathbf{b}$. We then have $\mathbf{a}^T\mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{a} = \mathbf{a}^T\mathbf{X}\mathbf{b} = \mathbf{a}^T\mathbf{a} > 0$.) Hence the matrix $\lim_{\beta \to 0}[\nabla(\mathbf{X}\boldsymbol{\beta})]^T(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T$ is positive definite. From Corollary 3.3, the augmented iterative approach converges to a stationary point of $v(\boldsymbol{\beta})$. □

A.4 Proof of Positive Definiteness of $(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T(\mathbf{X}\mathbf{X}^T)^{-1}\mathbf{X}$

Proof. To see this, consider any $\mathbf{y} \neq 0$. Define $\mathbf{b} = (\mathbf{X}\mathbf{X}^T)^{-1}\mathbf{X}\mathbf{y}$; thus $\mathbf{y} = \mathbf{X}^T\mathbf{b}$.
Therefore, $\mathbf{y}^T(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T(\mathbf{X}\mathbf{X}^T)^{-1}\mathbf{X}\mathbf{y} = \mathbf{y}^T(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{b} = \mathbf{b}^T\mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{b} > 0$, where the first equality holds by the definition of $\mathbf{b}$, the second equality holds because $\mathbf{y}^T = \mathbf{b}^T\mathbf{X}$, and the last inequality holds due to positive definiteness of $\mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T$ as shown in the proof of Proposition 3.5. □

A.5 Plot of Intel's Sales

Figure 7 shows Intel's sales (data are masked).

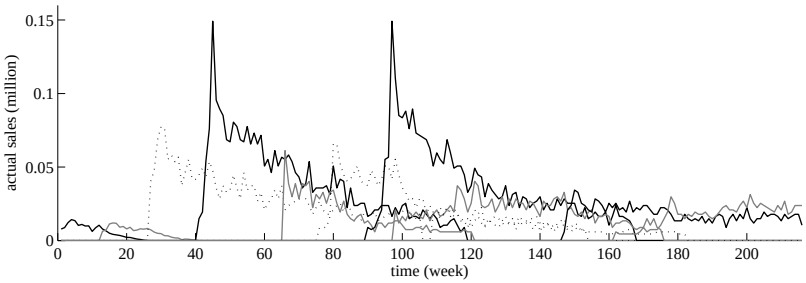


Figure 7: Intel Sales

A.6 Additional Implementation Details of the Intel Application

In this part of the appendix, we provide additional details on the implementation of our method. The masked data set is made available with this online appendix. To ensure small $\boldsymbol{\beta}$ values, we scale down the sales by 10^6. The trend curve estimated for total market sales (including sales by Intel and estimated sales for competition), denoted by $S(t)$, has the form $S(t) = \frac{a}{1+e^{-kt+b}}$ where $a = 0.885$, $b = 1.474$ and $k = 0.04818$, and we estimate the percentage expansion $\alpha(t)$ using $\alpha(t+1) = \frac{S(t+1)-S(t)}{S(t)}$. The iterative descent method is implemented in Matlab, using the procedure described in Section 3 and Corollary 3.3. We use the limited maximization rule (as described at the end of Section 3.2) for determining the step size. The initial parameter estimates are obtained using the cumulative sales as approximates for population path and running a linear regression as in equation (2.11). We then apply the iterative descent method to obtain the parameter estimate using the training data. This parameter estimate is then used to compute various fit and forecast errors according to the following equations. We use $\underline{t}(i)$ and $\overline{t}(i)$ to denote the introduction and ending time periods for product i. Let F be the set of products that has sales during the training data window (the first 120 weeks), and let F^p be the set of products that peaked during the training window. Let G be the set of products that have sales during the test window (the 96 weeks starting from week 121) and G^p be the set of products that peaked during the test window. Note that the set F and set G may overlap. The errors for model fit are computed as

$$
\begin{aligned}
\text{RMSE} &= \sqrt{\frac{\sum_{i\in F}\sum_{t=\underline{t}(i)}^{\overline{t}(i)}(\hat{s}_i(t)-s_i(t))^2}{\sum_{i\in F}\left(\overline{t}(i)-\underline{t}(i)+1\right)}}\ , \\
\text{MAE} &= \frac{\sum_{i\in F}\sum_{t=\underline{t}(i)}^{\overline{t}(i)}|\hat{s}_i(t)-s_i(t)|}{\sum_{i\in F}(\overline{t}(i)-\underline{t}(i)+1)}\ , \\
\text{MAPE} &= \frac{\sum_{i\in F}\sum_{t=\underline{t}(i)}^{\overline{t}(i)}(\hat{s}_i(t)-s_i(t))/s_i(t)}{\sum_{i\in F}(\overline{t}(i)-\underline{t}(i)+1)}\ , \\
\text{MdMAPE} &= \text{median}\{(\hat{s}_i(t)-s_i(t))/s_i(t)\}_{i\in F, t=\underline{t}(i),\ldots,\overline{t}(i)}\ , \\
\text{cumAPE} &= \frac{\sum_{i\in F}\sum_{t=\underline{t}(i)}^{\overline{t}(i)}(\hat{cs}_i(t)-cs_i(t))/cs_i(t)}{\sum_{i\in F}(\overline{t}(i)-\underline{t}(i)+1)}\ , \\
\text{peakMAPE} &= \sum_{i\in F^p}(\hat{s}_i^p(t)-s_i^p(t))/s_i^p(t))/|F^p|\ , \\
\text{timeMAE} &= \sum_{i\in F^p}(\hat{t}_i^p-t_i^p))/|F^p|\ ,
\end{aligned}
$$

where $s_i(t)$ and $\hat{s}_i(t)$ $(cs_i(t), \hat{cs}_i(t))$ represent the actual and predicted sales (cumulative sales), and $|F^p|$ denotes the size of set F^p. Note that if the sales of product $i \in F$ started before week 1 or ended after week 120, we revise the values of $\underline{i}(t)$ and $\overline{i}(t)$ accordingly (i.e., set $\underline{t}(i) = 1$ or $\overline{t}(i)$ to 120) when computing the fit error. Using the parameter obtained from the training data set, we generate sales forecast based on

equation (2.7), as well as equations in the online appendix A.1. We do not update the parameter estimates when making forecast, so the forecasts are not based on a rolling horizon. The forecast errors are similarly computed as for the fit errors by replacing sets F and F^p with sets G and G^p respectively.

For the alternative methods, Bass, Norton-Bass and Jun-Park methods, data fitting is performed in SAS. The parameter estimates are then imported to Matlab to generate forecast and compute fit and forecast errors, following the same error measure equations shown above.

A.7 Sensitivity to Initial Population Size

Tables 14 to 15 illustrate how the model fit, and forecast are affected when the estimate of the initial population size fluctuates. We illustrate with the "constrained, perf-only, best-comp" specification and vary the initial population of the competition, $y(0)$, and the initial population of product 1, $x_1(0)$.

$y(0)$	$x_1(0)$	RMSE	MAE	MAPE	MdAPE	cum MAPE	peak MAPE	time MAE	R^2
$6M$	$0M$	0.0175	0.0122	67%	42%	49%	3.45%	5.6	0.41
$7M$	$0M$	0.0176	0.0121	67%	41%	49%	3.43%	5.6	0.41
$7.5M$	$0M$	0.0176	0.0121	67%	41%	50%	3.44%	5.4	0.40
$8M$	$0M$	0.0177	0.0121	67%	41%	50%	3.45%	5.7	0.40
$9M$	$0M$	0.0178	0.0121	67%	41%	50%	3.47%	5.7	0.39
$6M$	$1M$	0.0175	0.0120	66%	39%	53%	3.43%	5.3	0.41
$7M$	$1M$	0.0177	0.0120	66%	39%	53%	3.45%	5.7	0.40
$7.5M$	$1M$	0.0177	0.0120	66%	40%	53%	3.46%	5.7	0.39
$8M$	$1M$	0.0178	0.0121	66%	40%	53%	3.47%	5.7	0.39
$9M$	$1M$	0.0179	0.0121	66%	38%	53%	3.49%	4.6	0.38

Table 14: Sensitivity of Model Fit to Initial Population Size

$y(0)$	$x_1(0)$	RMSE	MAE	MAPE	MdAPE	cumMAPE	peakMAPE	timeMAE
$6M$	$0M$	0.0134	0.0110	112%	46%	75%	0.44%	5.0
$7M$	$0M$	0.0118	0.0095	99%	39%	66%	0.60%	13.3
$7.5M$	$0M$	0.0111	0.0089	94%	38%	61%	0.69%	13.7
$8M$	$0M$	0.0105	0.0083	88%	35%	57%	0.77%	13.7
$9M$	$0M$	0.0095	0.0074	78%	31%	48%	0.92%	13.7
$6M$	$1M$	0.0118	0.0095	100%	39%	66%	0.61%	7.0
$7M$	$1M$	0.0105	0.0084	88%	35%	57%	0.77%	13.7
$7.5M$	$1M$	0.0100	0.0079	83%	34%	53%	0.84%	13.7
$8M$	$1M$	0.0096	0.0075	78%	32%	49%	0.90%	13.7
$9M$	$1M$	0.0089	0.0069	70%	28%	42%	1.02%	13.7

Table 15: Sensitivity of Forecast to Initial Population Size

A.8 Product Strength as a Weighted Sum of Performance and Price

Let the gap of product strength between product i and product j be given by $f_{ij} = g_{ij} + wr_{ij}$, where g_{ij} represents performance improvement and r_{ij} represents the price improvement from product i to product j. Therefore, the "perf-only" specification, in which $f_{ij} = g_{ij}$, is a special case with weight $w = 0$. Tables 16 and 17 show the fit and forecast performance respectively as the weight w increases.

w	RMSE	MAE	MAPE
0.00	0.017607	0.01210	66.7%
0.01	0.017607	0.01210	66.7%
0.05	0.017612	0.01212	66.9%
0.10	0.017625	0.01215	67.2%
0.20	0.017666	0.01223	67.8%
0.30	0.017719	0.01232	68.3%
0.40	0.017773	0.01240	68.6%
0.50	0.017809	0.01241	68.6%

Table 16: Model Fit as the Weight of Price Increases ("constrained, best-comp")

w	RMSE	MAE	MAPE
0.00	0.0111	0.0089	94%
0.01	0.0112	0.0090	94%
0.05	0.0115	0.0093	98%
0.10	0.0119	0.0097	101%
0.20	0.0127	0.0104	108%
0.30	0.0135	0.0112	115%
0.40	0.0146	0.0122	124%
0.50	0.0155	0.0131	130%

Table 17: Forecast Performance as the Weight of Price Increases ("constrained, best-comp")