Information Systems Research

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

Data-Driven Promotion Planning for Paid Mobile Applications

Manqi (Maggie) Li, Yan Huang, Amitabh Sinha

INFORMATION SYSTEMS RESEARCH
Vol. 31, No. 3, September 2020, pp. 1007–1029
ISSN 1047-7047 (print), ISSN 1526-5536 (online)
http://pubsonline.informs.org/journal/isre

Data-Driven Promotion Planning for Paid Mobile Applications

Manqi (Maggie) Li,[a] Yan Huang,[b] Amitabh Sinha[c]

[a] Stephen M. Ross School of Business, University of Michigan, Ann Arbor, Michigan 48109; [b] David A. Tepper School of Business, Carnegie Mellon University, Pittsburgh, Pennsylvania 15213; [c] Amazon.com, Seattle, Washington 98109

Contact: manqili@umich.edu, https://orcid.org/0000-0002-5210-4368 (M(M)L); yanhuang@cmu.edu, https://orcid.org/0000-0002-2262-1540 (YH); amitabsi@amazon.com, https://orcid.org/0000-0002-6035-3671 (AS)

Received: December 18, 2017
Revised: March 9, 2019; December 30, 2019
Accepted: February 10, 2020
Published Online in Articles in Advance: August 27, 2020

https://doi.org/10.1287/isre.2020.0928

Abstract. In this paper, we propose a two-step data analytic approach to the promotion planning for paid mobile applications (apps). In the first step, we use historical sales data to empirically estimate the app demand model and quantify the effect of price promotions on download volume. The estimation results reveal two interesting characteristics of the relationship between price promotion and download volume of mobile apps: (1) the magnitude of the direct immediate promotion effect is changing within a multiday promotion; and (2) due to the visibility effect (i.e., apps ranked high on the download chart are more visible to consumers), a price promotion also has an indirect effect on download volume by affecting app rank, and this effect can persist after the promotion ends. Based on the empirically estimated demand model, we formulate the app promotion optimization problem into a longest path problem, which takes into account the direct and indirect effects of promotions. To deal with the tractability of the longest path problem, we propose a moving planning window heuristic, which sequentially solves a series of subproblems with a shorter time horizon, to construct a promotion policy. Our heuristic promotion policy consists of shorter and more frequent promotions. We show that the proposed policy can increase the app lifetime revenue by around 10%.

History: Ahmed Abbasi, Senior Editor; Marius Niculescu, Associate Editor.
Supplemental Material: The online appendices are available at https://doi.org/10.1287/isre.2020.0928.

Keywords: promotion planning • mobile apps • digital goods • demand estimation • dynamic pricing

1. Introduction

With the prevalence of mobile devices and widespread Internet access, mobile applications (apps) increasingly play essential roles in people's daily lives. In 2018, global mobile app revenues reached $365 billion (including revenues via paid downloads and in-app features) and are projected to exceed $935 billion in 2023 (Statista 2019). Among all mobile apps, some can be downloaded for free (free apps), and others can be purchased at a price (paid apps). Free apps generate revenue mostly from in-app features and/or charges for exclusive features, functionality, or virtual goods. For paid apps, the focus of this study, customers pay the price to download the app, and the majority of paid apps' revenue is from the initial purchases (Priceonomics 2016).

As in many other markets, price promotions are one of the widely used tools to boost sales for paid apps (e.g., Loadown 2014). Price promotions are easy to implement in mobile app markets—app developers can update the price of their apps in their online account with a few clicks, and the new price will be in effect within hours. In practice, developers of paid apps typically follow a simple pricing strategy. There is a regular price, which is in effect for most of the time in the app's life cycle; for specific periods of time (promotion periods), the price of the app temporarily drops to a lower level, then returns to the normal level. To be effective, such promotions, including their timing, length, and depth (discount percentage), need to be carefully planned. Although there is extensive literature on price promotions, the majority of the existing studies are about physical, nondurable goods that are sold in brick-and-mortar stores and involve repeat purchases. Mobile apps (like other digital goods, such as digital music and software products) exhibit unique features, including zero marginal cost[1] and a very large number of highly differentiated products from which consumers can choose, which raises questions about the applicability of the existing knowledge and practices about price promotions for mobile apps.

Because mobile app consumers have so many apps to choose from, mobile app platforms (e.g., Apple App Store, Google Play, and Amazon App Store) provide them with tools to assist product discovery. One of the most noticeable tools is the sales charts (ranking). On the one hand, sales rank reflects the current sales performance of an app; on the other hand, apps in higher positions enjoy a higher degree of exposure to potential customers, which, in turn, can lead to higher future sales. In the literature on the demand for mobile apps (e.g., Ghose and Han 2014), price is typically considered as a factor affecting demand; however, it is often assumed to have a fixed, immediate effect on demand. Little research looks specifically at temporary price promotions and their possible dynamic effects on the demand for mobile apps, for example, through the *visibility effect* (i.e., the potential impact of an app's sales rank on its future demand). The existence of the visibility effect leads to intertemporal dependence on product demand, which significantly complicates the pricing and promotion planning decisions for apps.

This paper aims to examine the dynamic promotion effects and address the promotion planning problem (PPP) for paid apps. Specifically, we intend to answer the following research questions. First, how do price promotions affect paid apps' sales; is the effect of price promotions constant over time; is there any intertemporal dependence in product demand (e.g., via visibility effect)? Second, how can app developers design an effective promotion schedule to maximize the lifetime revenue of their products based on the empirically estimated demand function, especially in the presence of the visibility effect? We first formulate a system of equations consisting of (1) an app demand function that considers the potential visibility effect and the direct promotion effect (in addition to other observable app characteristics) and (2) a rank function that maps daily download volume to app rank. The system of equations is then estimated with a data set containing the daily records of 377 mobile apps that appeared in the Apple iOS top 500 action game chart for at least five days from September 2014 to December 2015, which is merged from two data sets obtained from two third-party market research companies in the mobile apps industry. A unique feature of our data set is that it contains direct information on download volume. Possibly due to the lack of data on download volume, most existing studies of mobile apps use app rank as a proxy for app sales (Carare 2012, Ghose et al. 2012a, Garg and Telang 2013). The download data allows us to more accurately examine the characteristics of app demand and directly estimate the rank function. We then formulate the PPP into a longest path problem (LPP). Due to the NP-hardness of the problem, we propose a moving planning window (MPW) heuristic consisting of a sequence of subproblems with a size that is polynomial in the length of the planning horizon.

Our empirical results confirm that price promotions have a significant *direct, immediate* positive effect on app download volume; however, the magnitude of this effect is much smaller on later days in a promotion. Besides, an app's download volume is significantly affected by its position on the sales chart, indicating the presence of a visibility effect. Through this visibility effect, promotions also exert an *indirect* impact on future sales. The numerical results of our proposed heuristic promotion policy show that developers can benefit from offering shorter price promotions at a higher frequency. Compared with the constant-price policy,[2] our proposed promotion policy improves an app's lifetime revenue by around 10% regardless of the initial state (e.g., rank, app age, and normal price) of the app.

Our paper makes several contributions. First, we use a unique data set to empirically examine the time-varying immediate promotion effect and the effect of ranking on next-period demand (visibility effect) and quantify the short-term and long-term effects of price promotions on app demand. We find that the positive impact of promotion is amplified during the promotion period by the visibility effect, and may persist even after the promotion ends. These findings provide novel insights into the characteristics of app demand where the visibility effect is present, and they extend the empirical literature of the demand for mobile apps; they also contribute to the literature of price promotions by introducing a new mechanism for the long-term effect of promotions—the visibility effect. Second, we take the empirical findings as inputs and close the loop of data-driven decision making by formulating the app PPP based on a flexible demand function estimated from historical sales data. We provide a heuristic for the app PPP that significantly improves the app lifetime revenue. This part of our research contributes to the literature of dynamic pricing and promotion planning in two ways: (1) instead of assuming a highly stylized demand model, our PPP is formulated based on a sophisticated, empirically estimated demand model; (2) our PPP considers the intertemporal dependence in the demand for mobile apps through the visibility effect, a mechanism that has not been studied in the literature. Third, the proposed heuristic (with a reasonable amount of customization in the calibration of the demand model and parameter retuning) can be readily applied to mobile apps' promotion planning, and the two-step data-analytic approach can serve as a general framework for promotion planning for other digital goods.

2. Literature Review

In this section, we review the multiple streams of literature relevant to this paper. This paper is related to the large and continuously growing literature, especially the empirical literature, on product pricing and promotion strategies in the fields of information systems, economics, and marketing. In the economics literature, price has been considered one of the most important factors affecting product demand (e.g., Pashigian 1988, Berry et al. 1995, Bils and Klenow 2004). In the marketing literature, price promotions have been extensively studied; researchers have examined the effects of price promotions on the demand for the promoted product, category demand, store performance, brand evaluation, and so on, and the mechanisms underlying these effects (e.g., Raju 1992, Blattberg et al. 1995, Raghubir and Corfman 1999, Nijs et al. 2001, Horváth and Fok 2013). Most studies find a positive immediate effect of price promotions on the sales of the product being promoted. Some papers document the longer-term effects of promotions (see Pauwels et al. (2002) for a comprehensive review), including the postpromotion trough, the mere purchase effect, and the promotion usage effect. These long-term effects of promotions are modeled through consumer stockpiling, promotion-induced consumer trial and learning, reference price effect, and so on. Since the majority of these studies concern physical, nondurable goods that are sold in brick-and-mortar stores and involve repeat purchases, the mechanisms identified are more relevant to this type of product.

In this paper, we study the promotion strategies for mobile apps. Mobile apps differ from physical, nondurable goods in that they are typically purchased only once, there are usually a large number of highly differentiated product for consumers to choose from, and online retailers provide sales rankings to assist consumers with product discovery. Given these differences, the demand for mobile apps may exhibit some unique characteristics, and the short- and long-term effects of promotions on mobile apps and the underlying mechanism may be different from those for physical nondurable goods documented in the marketing literature.

There is also an emerging stream of literature specifically on the demand for mobile apps. For example, Ghose and Han (2014) build a Berry-Levinsohn-Pakes (BLP)-style (Berry et al. 1995) econometric model to analyze the demand for apps in iOS and Google Play app stores. Ghose and Han (2014) consider app price as one of the covariates in the demand model and briefly discuss the implications of their empirical results for price discounts. Their model assumes that price promotion has a constant effect on demand and does not account for the intertemporal dependence of app demand through the visibility effect. Our demand model explicitly captures the time-varying immediate promotion effect and visibility effect. Based on our model, we provide a framework for the app PPP that accounts for these nuanced effects and show the revenue improvement our proposed promotion policy can achieve over promotion policies that do not account for these effects. Lee and Raghu (2014) examine key seller- and app-level characteristics that affect the survival of apps in the top-grossing chart. Garg and Telang (2013) introduce a novel method to infer download volume, which is rarely available to researchers, from rank data. Han et al. (2016) jointly study consumer app choices and usage patterns and demonstrate the applications of their model and findings to mobile competitive analysis, mobile user targeting, and mobile media planning. Mendelson and Moon (2016) investigate customers' app adoption, usage, and retention. Wang et al. (2018) develop a machine learning model to detect copycats. Our empirical analysis of mobile app demand builds upon these studies and extends them by incorporating two unique characteristics of app demand—the time-varying immediate promotion effect and the visibility effect.

An increasing number of papers also examine non-content decisions made by developers, including choices of the revenue model, pricing, and promotion strategies that affect app revenue. Among these factors, the choice of apps' revenue model has been extensively studied in the literature (Lambrecht et al. 2014, Liu et al. 2014, Ragaglia and Roma 2014, Nan et al. 2016, Oh et al. 2016, Rietveld 2016, Roma and Ragaglia 2016, Appel et al. 2020). Several other studies look into app pricing problems: Roma et al. (2016) and Roma and Dominici (2016) analyze how platform choices and app age affect app developers' pricing decisions. To the best of our knowledge, three papers study apps' promotion strategies. Among them, Askalidis (2015) and Lee et al. (2020) focus on cross-app promotions, in which one app is featured/promoted in another app. In contrast, our work focuses on app promotions in the form of temporary price changes, which apply to all paid apps. Chaudhari and Byers (2017) examine a unique feature once available in the Amazon App Store, called the "free app of the day," and document the effects of the free promotion on app sales in the focal market, app sales in other app markets, and app ratings. App rank is the dependent variable in their empirical models as a proxy for app download volume. Our paper focuses on the within-platform effect of price promotions, considering the time-varying immediate promotion effect on current demand, the effect of app rank on future demand, and the mechanism behind the long-term effect of price promotions.

Many online retailers provide sales rank to make it easier for consumers to find best-selling products from the broad set of products (Garg and Telang 2013). Therefore, a product's sales rank (or, more broadly, position in any rankings) may affect the product's subsequent sales. Several empirical studies of online marketplaces and sponsor search advertising have documented this effect, which we call the "visibility effect" (Sorensen 2007; Ghose et al. 2012b, 2014;

Agarwal et al. 2011); several papers also document the existence of the visibility effect in mobile app markets (Carare 2012, Ifrach and Johari 2014). However, due to the lack of download volume data, Ifrach and Johari (2014) and Carare (2012) have not been able to quantify the magnitude of the visibility effect accurately. For our study, we obtain a unique data set that records actual download volume, which enables us to quantify the magnitude. Moreover, these papers focus on documenting the impact of rank on sales, whereas our paper focuses on the promotion planning of mobile apps in the presence of this visibility effect. Ours is the first paper to connect the promotion effect and the visibility effect in a closed loop—price promotions as a tool to boost sales rank and the visibility effect as the mechanism for the dynamic, long-term effects of price promotions. In addition to estimating the visibility effect, we provide prescriptive solutions for apps' promotion policies that account for the visibility effect.

Finally, this paper is also related to the stream of research on dynamic pricing/promotion planning in the operations management literature. Traditionally, PPPs are formulated and solved as a special case of dynamic pricing problems (see Talluri and Van Ryzin (2006) and the references therein). Most of the previous studies on PPP consider promotion planning for physical goods. The main trade-off examined in these studies is between the demand increase during the promotion and the postpromotion demand dip (Assuncao and Meyer 1993, Popescu and Wu 2007, Su 2010, Cohen et al. 2017). In contrast, due to the existence of the visibility effect, for mobile apps, the download volume often stays at a relatively high level, as opposed to experiencing an immediate drop, after a promotion. Therefore, the promotion policies proposed in the existing literature, which are designed for physical goods, cannot be applied directly to mobile apps. In addition, to ensure tractability, most existing papers in the dynamic pricing literature assume demand functions have specific properties (e.g., linearity or diffusion property; see Bitran and Caldentey (2003) and the references therein for a detailed review). In practice, these demand properties often are not satisfied. We formulate the PPP based on a realistic demand function empirically estimated using real-world data. Since the demand function we face is much more complicated than those studied in the dynamic pricing literature, we propose a MPW heuristic to approximate the optimal promotion policy. We show that the proposed heuristic can significantly improve the app lifetime revenue and that it is important to consider the visibility effect in price promotions for mobile apps—ignoring the visibility effect will lead to a significant revenue loss.

3. Data and Descriptive Analysis

The data set we use comes from two major market research companies in the mobile app industry. We obtain a panel data set containing apps' daily download volume, rank, rating score, and price from one of the companies and augment the data by collecting static information (e.g., app release date, developer information, and cumulative update history) from the other company. Our data set contains information on the top 500 apps in the paid action game category in the U.S. iOS App Store from September 1, 2014 to December 31, 2015. The top 500 list is updated daily, and apps frequently move on to and off the list.[3] We remove 25 games that ever offered temporary free downloading during the study period, because free promotions will disrupt apps' ranking on the sales chart.[4] In addition, apps with less than five observations (i.e., appear in the top 500 paid chart for less than five days)[5] are also excluded, because there are not enough observations to make meaningful inferences about these apps. There are 377 unique apps in our final data set. The app-day level summary statistics of the key variables in our data are reported in Table 1.

Compared with suppliers in many other markets, suppliers in mobile app markets (i.e., app developers) have less flexibility in setting the price for their products, as they have a limited set of price choices. In the iOS App Store, app developers are provided with a menu of price choices ranging from \$0 to \$999.99. In the U.S., specifically, there are 87 prices on the menu from which app developers can choose (see Table 2 for details).[6] In Table 1 we can see that the observed prices of the action games represented in our data fall in a relatively small range—from \$0.99 to \$7.99. Therefore, in the rest of the paper, we consider only the eight candidate prices from \$0.99 to \$7.99 with \$1 steps. Although developers can change their app's price at any time, price changes are relatively infrequent in the data. Out of the 377 apps in our sample, only 138 apps experienced at least one price change, and 80 apps had multiple price changes in the time window spanned by the data. These price changes took the form of promotions, where the app price

Table 1. Summary Statistics

Variable	Mean	Standard deviation	Minimum	Maximum
Price	2.82	1.90	0.99	7.99
Daily download	209.46	1,373.36	1	59,773
App age	933.11	622.70	2	2,643
Rank	180.11	125.18	1	500

Note. $N = 100{,}531$.

Table 2. Price Tiers in iOS App Store

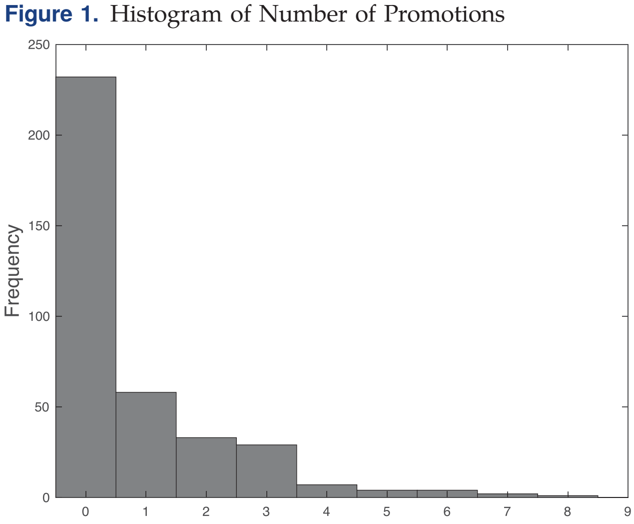
Price Range	Increment
\$0–\$49.99	\$1
\$49.99–\$99.99	\$5
\$99.99–\$249.99	\$10
\$249.99–\$499.99	\$50
\$499.99–\$999.99	\$100

dropped to a lower level for a short period (e.g., several days). Among the apps in our sample, on average, each app experienced 0.824 such promotions during our study period (min = 0 and max = 8; Figure 1 displays the histogram of the number of promotions each app experienced in our sample).

Each price promotion can be characterized by two parameters: *promotion depth* and *promotion length*. Let p^{original} be the original price of an app and $p^{\text{promotion}}$ be the discounted price effective during the promotion; the promotion depth is then defined as the percentage price decrease during the promotion $(p^{\text{original}} - p^{\text{promotion}})/p^{\text{original}}$, and promotion length is measured by the number of days the promotion lasts. The distributions of the depth and length of the promotions observed in our data are shown in Figure 2: the majority of promotions observed in our sample involved a 50%–70% discount and lasted 1 to 20 days. The detailed summary statistics of promotion depth and promotion length are provided in Table 3. In the data, the number of promotions an app experienced is not significantly correlated with app rating, and apps developed by less experienced developers (those that publish a smaller number of apps) tend to engage in slightly more promotions. Neither promotion depth nor promotion length is significantly correlated with app rating.[7]

We first visualize the relationship between daily rank and daily download volume on the log-scale in the left panel of Figure 3. Consistent with Chevalier and Goolsbee (2003), Ghose and Han (2014), and Garg and Telang (2013), our data suggests a linear relationship between the logarithm of rank and the logarithm of download volume. The middle and right panels of Figure 3 feature two sample apps in our data. The solid line in each plot represents app price, which drops from \$4.99 to \$0.99/\$1.99 for a short period of time. In both examples, app download volume (represented by the triangles) increases significantly during the promotion period; however, the magnitude of the sales increase declines gradually, which suggests that the promotion effect is not constant over time, and a promotion indicator in a demand model is not sufficient to capture this time-varying effect. Additionally, download volume is highly (negatively) correlated with app rank. On the one hand, the current-period rank reflects the current-period download volume[8]; on the other hand, apps ranked higher on the chart are more visible to customers and more likely to be featured/recommended by app platforms and, thus, have a higher chance of being discovered and purchased. This *visibility effect* distinguishes the PPP for mobile apps from that for most physical goods sold in brick-and-mortar stores. In the context of mobile apps, developers may sacrifice some revenue during the promotion period by providing a lower price; the extra download volume resulting from the discount price can push the app into a higher position in the sales chart. After the promotion ends, the app can continue enjoying the increased visibility and a higher download volume at the original price. A comparison of the two sample apps shows that the sales increase after promotion ends is more significant for the second app (right panel) than the first app (middle panel), likely because the first app's promotion is long and does not end before the promotion effect fades away. As discussed in more detail later, an app's postpromotion sales increase is affected by the length of the promotion it has experienced.

Figure 1. Histogram of Number of Promotions

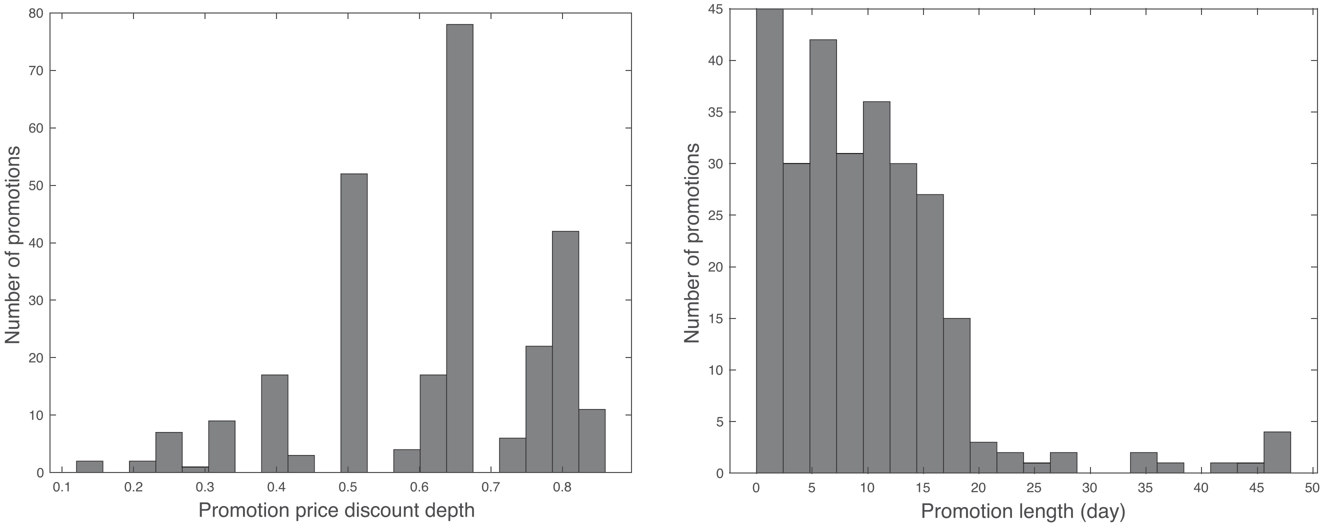


To explore how app download volume changes during and after a price promotion among all apps in our sample, we fit a descriptive regression (Equation (1)). The purpose of this preliminary analysis is to develop intuition and guide the construction of the main empirical model.[9] We have

$$\log(D_{it}) = \alpha_i + \gamma_1 \log(h_{it}) + \sum_{r=1}^{30} \beta_r^{on} I(d_{it} = r) + \sum_{s=1}^{30} \beta_s^{post} I(n_{it} = s) + \epsilon_{it}. \tag{1}$$

In Equation (1), D_{it} represents the download volume of app i on day t (the log transformation is applied because D_{it} is highly skewed); d_{it} (n_{it}) represents the

Figure 2. Distributions of Promotion Depth and Promotion Length



number of days app i is on (after) a price promotion in period t. The coefficient β_r^{on} (β_s^{post}) captures the change in the logarithm of the download volume r (s) days into (after) a promotion, after controlling for the logarithm of app age (in days, denoted as h_{it}) and the app-specific fixed effect (denoted as α_i). In the left panel of Figure 4, we display the point estimates of all β_r^{on}'s and β_s^{post}'s that are statistically significant at the 0.1 level. The vertical line separates the "on promotion" region and the "postpromotion" region. Again, it is evident that price promotion has an overall positive effect on app download volume, and this effect is time-varying. The positive effect persists even after the price promotion ends; β_s^{post} is significantly positive for several s values.[10] Two different effects may cause an increase in download volume: (1) the immediate promotion effect resulting from the decreased price and (2) a larger visibility brought by the higher ranking during the promotion. In this preliminary regression, we could not disentangle these two effects. In Section 4.1, we build a demand model to capture these two effects. In addition, we also find suggestive evidence that conducting promotions, in general, improves app revenue. The right panel of Figure 4 shows that the average cumulative revenue over time (the horizontal axis is the age of the app in days) of the apps that have ever engaged in price promotions is higher than those that have not.

4. Empirical Model and Estimation

4.1. Empirical Model

In this section, we build upon the observations presented in Section 3 and construct an empirical model of mobile app demand that describes how price promotions affect download volume.

We formulate a system of equations consisting of Equations (2) and (3), where Equation (2) is the demand function capturing how price promotions, last-period rank, and other app characteristics affect download volume. Equation (3) is the ranking function characterizing how download volume is reflected by rank. This model describes the mechanism driving the promotion effects shown in the left panel of Figure 4; it captures the interperiod dependence of rank and download volume and allows us to tease apart the visibility effect and the immediate promotion effect. We have

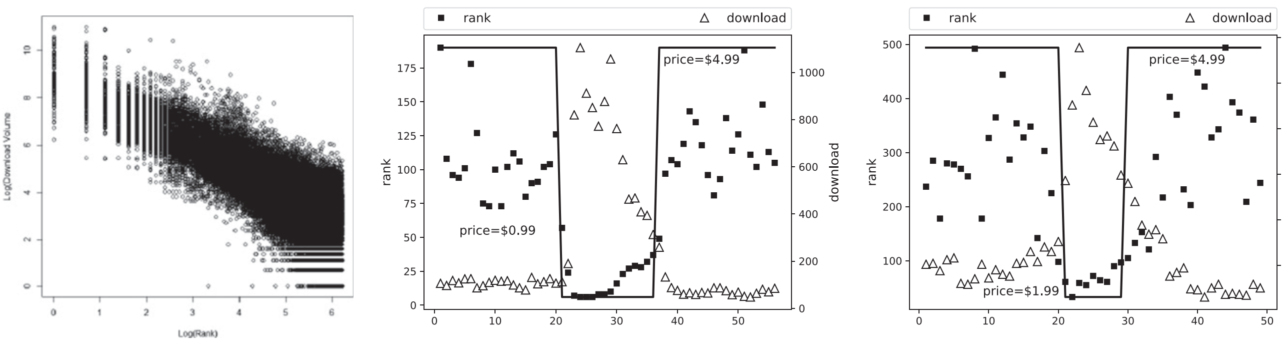
$$\begin{aligned}\log(D_{it}) = {} & \alpha_i + \beta_1 \log\big(r_{i(t-1)}\big) + \beta_2 \log(h_{it}) + \beta_3 \log(u_{it}) \\ & + \beta_4 q_{it} + \beta_5 \Delta p_{it} + \beta_6 \Delta p_{it} \cdot d_{it} \\ & + \beta_7 \Delta p_{it} \cdot d_{it}^2 + \beta_8 \Delta p_{it} \cdot d_{it}^3 + \cdots + \sum_{m=1}^{11} \eta_m M_{mit} \\ & + \sum_{w=1}^{6} \phi_w W_{wit} + \epsilon_{it}, \end{aligned} \tag{2}$$

$$\log(r_{it}) = \gamma_0 + \gamma_1 \log(D_{it}) + \varepsilon_{it}. \tag{3}$$

Table 3. Summary Statistics of Promotion Depth, Promotion Length and Number of Promotions per App

Statistic	Mean	Standard deviation	Minimum	25th percentile	75th percentile	Maximum
Depth	0.617	0.159	0.143	0.503	0.752	0.858
Length	10.073	8.309	1	4	14	47
Number of promotions per app	0.824	1.395	0	0	1	8

Figure 3. Relationship between Sales Rank and Download Volume and Sample Apps; Left Panel: Log(Sales Rank) vs. Log(Download Volume); Middle and Right Panels: Sample App Promotions



In the equations, i is the index for mobile apps and t is the index for time (in days). The notation and descriptions of the variables in the model are summarized in Table 4. The dependent variable of the demand function is the logarithm of the download volume of an app on a day (D_{it}). To account for the effect of unobserved time-invariant app-specific characteristics (e.g., story-line and playability, which constitute each app's base quality) on download volume, we include an app-specific fixed effect α_i in our model. To capture the visibility effect evident in the data, we follow Carare (2012) and include the one-period lagged rank ($\log(r_{i(t-1)})$) as one of the independent variables.[11,12] We then include the logarithm of app i's age in days ($\log(h_{it})$) and the logarithm of the number of days since the last version update ($\log(u_{it})$) to account for the possible effects of app age and version age on app demand. Following Ghose and Han (2014), we also incorporate app ratings (ranging from 1 to 5 stars, denoted as q_{it}) in the demand equation. We use Δp_{it} to represent the depth of the promotion that app i is experiencing on day t.[13] (For all t's at which app i is not on promotion, d_{it} equals 0.) The coefficient of Δp_{it}, denoted by β_5, captures the baseline effect of a promotion with a depth of 1 on the logarithm of download volume. To capture the possibility that the size of the promotion effect varies on different days in the promotion period, we interact Δp_{it} with polynomial terms of d_{it}, with d_{it} representing the number of days into the current promotion. For example, if the price of app i drops on day τ, then $d_{i\tau} = 1$, $d_{i(\tau+1)} = 2$, $d_{i(\tau+2)} = 3$, and so on. The promotion effect app i experiences on day t is then $\beta_5\Delta p_{it} + \beta_6\Delta p_{it} \cdot d_{it} + \beta_7\Delta p_{it} \cdot d_{it}^2 + \cdots$. (We discuss later in this section how we determine the number of polynomial terms of d_{it} to keep.) Finally, we include a series of month dummies (M_{mit}) and a series of day-of-week dummies to control for seasonality and weekday/weekend effects, respectively.

For the ranking function, we follow Chevalier and Goolsbee (2003), Garg and Telang (2013), and Ghose and Han (2014) and assume a Pareto distribution between rank (r_{it}) and download volume (D_{it}), that is, $r_{it} = a \cdot D_{it}^{-b}$. Here b is the shape parameter and a is the scale parameter. Taking the logarithm of both sides of the equation and adding noise terms to the mapping, we can rewrite the ranking function as Equation (3). The effect of a promotion captured by

Figure 4. Evidence of Promotion Effects in Data; Left Panel: Coefficients of Days-on(After)-Promotion Dummies; Right Panel: Cumulative Revenue Over Time: Apps with and Without Promotions

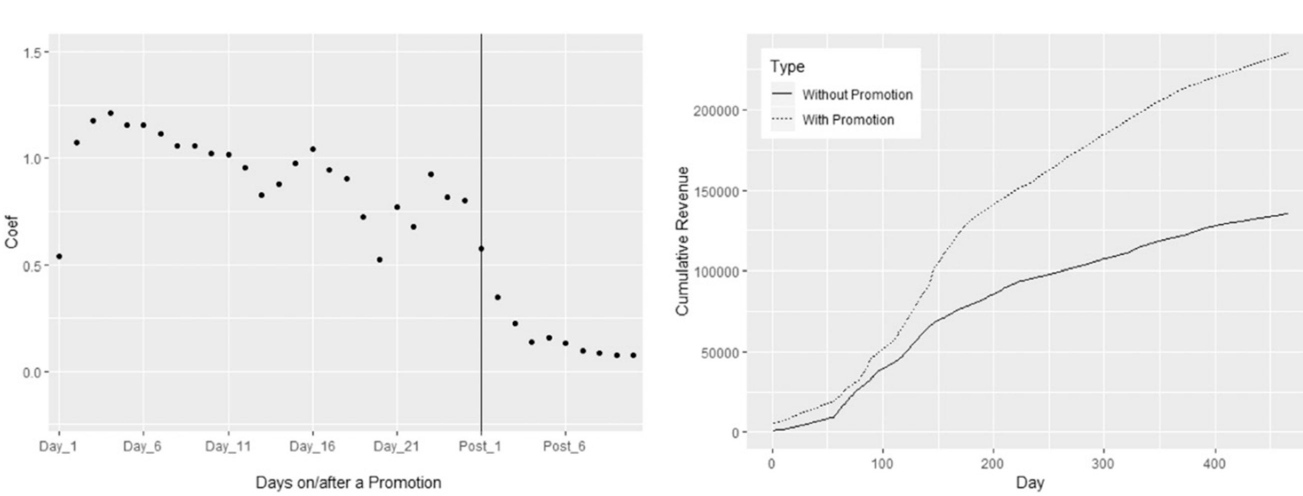


Table 4. Variable Description

D_{it}	Download volume of app i on day t
$r_{i(t-1)}$	Rank of app i on day $t-1$
h_{it}	The number of days since app i's release
u_{it}	The number of days since app i's last update
q_{it}	Current rating (on a scale of 1–5 stars) of app i on day t
Δp_{it}	Depth of the promotion for app i on day t
d_{it}	Days in promotion for app i on day t
M_{mit}	Month dummies
W_{wit}	Day-of-week dummies

"$\beta_5\Delta p_{it} + \beta_6\Delta p_{it} \cdot d_{it} + \beta_7\Delta p_{it} \cdot d_{it}^2 + \cdots$" in Equation (2) is the immediate promotion effect, which corresponds to the direct effect of a price drop on the current-period download volume. In addition to this direct effect, price promotions have an indirect effect on demand through the visibility effect—the direct effect of the promotion in period t on the period-t demand will lead to changes in the focal app's sales rank, and the period-t app rank can further affect the period-$(t+1)$ download volume. In fact, Equations (2) and (3) constitute a general form of a vector autoregressive (VAR) model. The total effect of a promotion on current and future app demand should be evaluated using an approach similar to the impulse response function, which will be elaborated in Section 4.3.

In the current model, we do not explicitly consider substitution between apps, because mobile games are highly differentiated (in terms of story-line, graphic design, game mechanisms, and game-play experience). Therefore, there is likely little substitution between games, as compared with physical, nondurable products (e.g., detergents and cereal). Similar assumptions are made in studies of other sectors of the media and entertainment industry: Chen et al. (2018) consider each book as a monopolistic product and Danaher et al. (2014) treat each music album as a monopolistic product. In Section 4.4, we compare alternative models with/without this assumption.

4.2. Instrumental Variables

Endogeneity is a known common challenge in demand estimation. In our demand model, the timing, depth, and length of promotions are potentially endogenous, because developers' promotion decisions may be based on the expected demand. Not only the depth but also the timing and length of promotions are reflected in the series of Δp_{it} in our model—in periods when app i is not on promotion, $\Delta p_{it} = 0$; in periods when app i is on promotion, Δp_{it} takes a positive value.[14] To address the endogeneity with respect to developers' promotion decisions, we instrument for Δp_{it}.

We follow Ghose and Han (2014) and consider the average price of all apps produced by the focal app's developer in the Google Play store on the same day as an instrument. The prices of Android apps (sold in the Google Play store) produced by the same developer are likely to be correlated with the focal iOS app's price. However, the demand shock to the focal iOS app is unlikely to be correlated with the prices and download volumes of apps sold in the Google Play store, because the two stores have different customer bases.[15] This is a BLP-style instrument (Berry et al. 1995). Since the price of an app is affected by the production and maintenance costs of the app and those costs are shared by the Android apps and iOS apps produced by the same developer, the prices of apps sold in the iOS and Google Play app stores are also likely to be correlated. It is common for a mobile game developer to publish the same game in both iOS and Google Play stores. Although the iOS version and Android version (sold in the Google Play store) of the same app may be written in different programming languages, they share common graphical modules and storylines. Therefore, the development costs of the iOS and Android versions of the same app tend to correlate with each other. Mobile games typically make major updates in both stores at the same time to ensure similar customer experiences across platforms. Furthermore, the iOS and Android versions of the same game are maintained by the same development team, and the two platforms usually share the same database; therefore, the maintenance costs associated with the iOS and Android versions of the same game are likely to be correlated. We further use a set of 13 version indicator variables (referred to as *Version Number*) to instrument for Δp_{it}. For example, the kth indicator variable takes the value of 1 when the current version is the kth version, and 0 otherwise. This set of indicator variables is likely to be correlated with Δp_{it} because the maintenance cost may vary with app version. Indeed, the first-stage F-test strongly rejects (p-value = 0.000) the hypothesis that the set of version indicator variables are jointly uncorrelated with Δp_{it}. If the number of version updates affects demand, it does so by affecting app quality (Ghose and Han 2014); since we have already controlled for app quality by incorporating app fixed effects and app rating, app version is unlikely to be correlated with demand shocks.

One plausible mechanism causing the endogeneity of promotion length is that app developers may dynamically determine the length of promotion after observing the realized promotion effect. That is, if a promotion is effective immediately after it is turned on, the developer may end it early; if not, the developer may let it run for a more extensive period. If this is true, we will find that shorter promotions tend to be more effective early on. We explore if rank improvement (current-day rank minus rank on the day prior to the promotion) on each day in the promotion period is different between short and long promotions, and we find that rank improvement is, in fact, slightly larger in the case of long promotions.[16] Therefore, there is no evidence for the endogeneity of promotion length. Even if such endogeneity exists, instrumenting for Δp_{it} can still address it.

Another endogeneity issue we need to address is with $\log(r_{i(t-1)})$. Although the last-period rank cannot be influenced by unobservables that affect the current-period demand, demand shocks for the same app may be correlated from period to period. Following the same rationale in Carare (2012), we use the logged two- and three-period lagged ranks as instruments for $\log(r_{i(t-1)})$. We carry out tests for weak instruments and overidentifying restrictions on the instruments for the two potentially endogenous variables, and the results support the validity of the instruments. Finally, we use all independent variables except $\log(r_{i(t-1)})$ in Equation (2) as instruments for $\log(D_{it})$ in Equation (3), although the ranking function is unlikely to have endogeneity problems, because it merely describes the mapping between download volume and app rank.

4.3. Estimation Results

We estimate the system of Equations (2) and (3) using two-stage least squares (2SLS).[17] We experiment with three alternative specifications of Equation (2) with first-, second-, and third-degree polynomials of d_{it}. The estimation results for the three alternative specifications are reported in Table 5. We find that models (1) and (2) generate similar immediate promotion effect curves, and both $\Delta p_{it} \cdot d_{it}$ and $\Delta p_{it} \cdot d_{it}^2$ are significant. Adding the $\Delta p_{it} \cdot d_{it}^3$ term into the model does not significantly improve the model fit, but due to multicollinearity (the correlation between $\Delta p_{it} \cdot d_{it}^2$ and $\Delta p_{it} \cdot d_{it}^3$ is 0.97), all the $\Delta p_{it} \cdot d_{it}^k$ terms become insignificant. Therefore, we choose Model (2) as the main demand function and use it for promotion optimization (to be discussed later in the paper). The instruments introduced earlier are used in the estimation of all these models.

The estimation results for the ranking function are reported in Table 6. The shape parameter of the Pareto distribution is estimated to be 0.762, which is comparable to those reported in earlier studies (see Garg and Telang (2013) for a summary of Pareto shape parameters estimated in the literature; the range is from 0.613 to 1.2). The scale parameter is known to vary across categories (Ghose and Han 2014). For iOS paid action games in the U.S. market, which are the focus of our study, we find that the top-ranked iOS paid action game's daily download volume is approximately 31,310 ($\exp(7.888/0.762)$), which is also similar in scale to Garg and Telang (2013).

4.3.1. Immediate Promotion Effect. The estimation results confirm that price promotions have a significant immediate positive impact on app download volume. If we plug the estimated β_5, β_6, and β_7 into Equation (2), we can calculate the daily immediate effect of a promotion as $1.173\Delta p_{it} - 0.110\Delta p_{it} \cdot d_{it} + 0.002\Delta p_{it} \cdot d_{it}^2$. The dashed line in Figure 5 shows the daily immediate effect of a promotion with a depth of 0.5 over a 25-day promotion period. We can see that the immediate promotion effect generally decreases over time.

Table 5. Regression Results of Demand Function (Equation (2))

Dependent variable: $\log(D_{it})$	(1)	(2)	(3)
$\log(h_{it})$	−0.223*** (0.005)	−0.217*** (0.005)	−0.211*** (0.005)
$\log(r_{i(t-1)})$	−0.943*** (0.006)	−0.957*** (0.006)	−0.964*** (0.005)
$\log(u_{it})$	−0.052*** (0.002)	−0.052*** (0.002)	−0.053*** (0.002)
q_{it}	0.006 (0.004)	0.006 (0.004)	0.006 (0.004)
Δp_{it}	1.059*** (0.064)	1.173*** (0.110)	0.322** (0.132)
$d_{it} \cdot \Delta p_{it}$	−0.040*** (0.004)	−0.110*** (0.015)	0.019 (0.030)
$d_{it}^2 \cdot \Delta p_{it}$		0.002*** (0.0003)	−0.002 (0.002)
$d_{it}^3 \cdot \Delta p_{it}$			0.00004 (0.00003)
App fixed effects	Yes	Yes	Yes
Month dummies	Yes	Yes	Yes
Day-of-week dummies	Yes	Yes	Yes
Observations	100,531	100,531	100,531
R^2	0.538	0.539	0.540
Adjusted R^2	0.538	0.539	0.540
Residual standard error	0.514 (df = 100,508)	0.514 (df = 100,507)	0.513 (df = 100,506)

Note. df, degrees of freedom.
$^{*}p < 0.1$; $^{**}p < 0.05$; $^{***}p < 0.01$.

Table 6. Regression Results of Ranking Function (Equation (3))

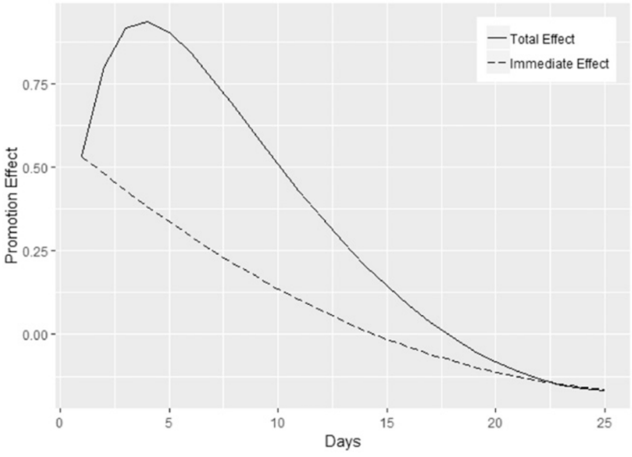
	Dependent variable: $\log(r_{it})$
$\log(D_{it})$	−0.762*** (0.002)
Constant	7.888*** (0.007)
Observations	100,531
R^2	0.622
Adjusted R^2	0.622
Residual standard error	0.635 (df = 100,529)
F-statistic	185,646.900*** (df = 1; 100,529)

Note. df, degrees of freedom.
$^{*}p < 0.1$; $^{**}p < 0.05$; $^{***}p < 0.01$.

(Figure 2 shows that 95% of the promotions in the data last no more than 25 days; therefore, we are only able to accurately estimate the promotion effect for $d_{it} \le 25$. The estimated promotion effect for $d_{it} > 25$ is subject to extrapolation.)

The promotion effect is negative for $d_{it} \ge 15$. One possible explanation is that, in each period, a number of new consumers are exposed to the focal app, and the number of such customers is affected by app rank. The consumers who have a willingness-to-pay higher than the normal app price will purchase the app immediately, whereas those whose willingness-to-pay is below the normal app price become aware of the product but will not purchase it immediately. Before a promotion starts, a significant number of the latter type of consumers may have accumulated; when the price promotion starts, those whose willingness-to-pay is higher than the discounted price will then purchase the app during the promotion period. Since each consumer will only purchase an app once, after the app is purchased, the consumer will leave the market. As the market runs out of such consumers and new consumers who have a willingness to pay higher than the discounted price arrive at a much smaller rate, the immediate effect of the price promotion gradually declines. The existence of app price aggregators may draw consumers who would otherwise arrive at a later time into the market earlier, leading to a negative net immediate effect when the promotion runs for an extensive period (i.e., more than 15 days).[18]

Figure 5. Immediate and Total Effect of a 25-Day Promotion (Promotion Depth = 0.5)



Moreover, the R^2 of our main model is larger than that of the model without the $\Delta p_{it} \cdot d_{it}$ and $\Delta p_{it} \cdot d_{it}^2$ terms ("Constant promotion effect" model in Table 7), showing the importance of capturing the time-varying magnitude of the immediate promotion effect. As we will show later in the numerical study, if developers ignore the fact that the immediate promotion effect is decreasing and assume a constant promotion effect, they tend to offer longer promotions, which may cause revenue losses.

4.3.2. Visibility Effect. The estimated coefficient of $\log(r_{i(t-1)})$ is negative and significant, indicating that as $\log(r_{i(t-1)})$ decreases (i.e., app rank improves), the next-period download volume increases. This effect is economically significant—a 1% decrease (improvement) in rank can lead to a 0.957% increase in the next period download volume. The coefficient of $\log(r_{i(t-1)})$ captures visibility benefits brought by (1) being in a higher position on the download chart in the iOS App Store; (2) an increased probability of being featured in app stores' homepage; (3) an increased probability of being recommended on social media platforms; and (4) an increase in user base ($D_{i(t-1)}$) that may affect future app demand through non-rank-related channels (e.g., through word-of-mouth).[19] Also, the R^2 of our main model is much larger than that of the model that does not consider the visibility effect ("Without visibility effect" model in Table 7), demonstrating the importance of the visibility effect in explaining app demand.

4.3.3. Total Promotion Effect. The existence of the visibility effect leads to intertemporal dependence in download volume. The immediate effect of a promotion will improve app ranking at the beginning of the next period; a higher ranking will in turn positively impact app demand in the next period. The visibility effect reinforces the benefit that an app can get from a promotion.

Computing Total Promotion Effect. In vector autoregression (VAR) models, the *impulse response function* is used to track the effect of a *one-time* shock in a *dependent variable* in a period on the current (period t) and future (periods $t+1, t+2, \ldots$) values of all dependent variables ($\log(D_{it})$ and $\log(r_{it})$ in our case). However, in our specific setting, the impulse response function cannot directly serve the purpose of evaluating

Table 7. Regression Results of Alternative Demand Specifications

	Dependent variable: log(download)		
	Main model	Constant promotion effect	Without visibility effect
$\log(h_{it})$	−0.217*** (0.005)	−0.212*** (0.005)	−0.571*** (0.006)
$\log(r_{i(t-1)})$	−0.957*** (0.006)	−0.964*** (0.005)	
$\log(u_{it})$	−0.052*** (0.002)	−0.053*** (0.002)	−0.059*** (0.002)
q_{it}	0.006 (0.004)	0.006 (0.004)	0.031*** (0.005)
Δp_{it}	1.173*** (0.110)	0.374*** (0.020)	1.486*** (0.137)
$d_{it} \cdot \Delta p_{it}$	−0.110*** (0.015)		0.002 (0.019)
$d_{it}^2 \cdot \Delta p_{it}$	0.002*** (0.0003)		0.0001 (0.0004)
App Fixed Effects	Yes	Yes	Yes
Month Dummies	Yes	Yes	Yes
Day of Week Dummies	Yes	Yes	Yes
Observations	100,531	100,531	100,531
R^2	0.539	0.536	0.283
Adjusted R^2	0.539	0.536	0.282
Residual standard error	0.514 (df = 100,507)	0.516 (df = 100,509)	0.641 (df = 100,508)

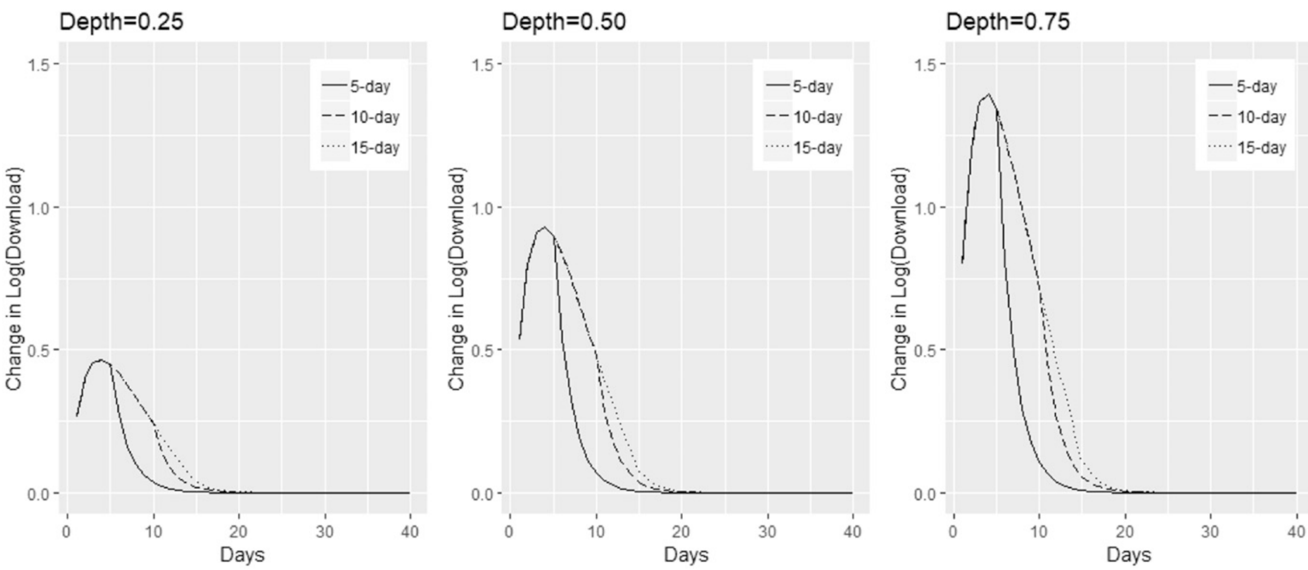
Note. df, degrees of freedom.
$^{*}p < 0.1$; $^{**}p < 0.05$; $^{***}p < 0.01$.

the total effect of a promotion for the following reasons: First, a promotion typically lasts multiple days. We treat each promotion as a whole, and evaluate the effect of a promotion with a given combination of length and depth on download volume over time. Second, the variable being varied (Δp_{it}) in our setting is not one of the dependent variables in the system.

To evaluate the total promotion effect, we follow closely the simulation method for generating the impulse response function explained in Hamilton (1994, page 319). The main modifications we make are that the "shock" is introduced to Δp_{it}, not to one of the dependent variables, and that the shock is not one-time, but lasts for a few days. Specifically, to evaluate the total effect of a promotion starting on day τ with a depth of g and a length of l days on app i whose day $\tau - 1$ rank was w, we carry out the following steps: First, we fix $\log(r_{i(\tau-1)})$ to $\log(w)$. Since the promotion starts on day τ and lasts for l days, in the simulation, Δp_{it} for all $t \in \{\tau, \tau+1, \ldots, \tau+l-1\}$ is set to g, and Δp_{it} for all $t \geq \tau + l$ is set to 0; d_{it} is set to $1 + t - \tau$ for $t \in \{\tau, \tau+1, \ldots, \tau+l-1\}$, and 0 for $t \geq \tau + l$. Following Hamilton (1994), we set all ϵ_t and ε_t to zero. Then $h_{i\tau}$ and $u_{i\tau}$ are set to their respective average values in the sample and evolve deterministically as $h_{i(t+1)} = h_{it} + 1$ and $u_{i(t+1)} = u_{it} + 1$. The month dummies, day-of-week dummies, and rating score q_{it} are omitted. We then simulate the daily download volume and app rank from day τ onward. That is, starting from $t = \tau$, for each day, we plug the values of the right-hand side variables into the estimated Equation (2) to simulate $\log(D_{it})$; then we plug the simulated $\log(D_{it})$ into the estimated Equation (3) to simulate $\log(r_{it})$. We use the same method to simulate the daily download volume and app rank for the same period of time without the promotion (where Δp_{it} and d_{it} are set to 0 throughout). Finally, we take the pairwise difference in $\log(D_{it})$ between the with- and without-promotion cases; the difference represents the total effect of the promotion on each day.

Discussion. The solid line in Figure 5 visualizes the total promotion effect. The gap between the solid and dashed lines represents the indirect effect of promotions reinforced by the visibility effect. The increasing total promotion effect that occurs in the first few days of a promotion is due to the larger immediate positive effect of the promotion and its resulting higher app rank, which generates more visibility and boosts demand significantly in the subsequent periods. As the promotion lasts longer, the decrease in the immediate promotion effect dominates the visibility effect; thus, the total promotion effect also declines. Figure 6 shows how the total promotion effect varies with promotion depth (0.25, 0.5, or 0.75) and length (5 days, 10 days, or 15 days). The first observation on each curve corresponds to the first promotion day. The figure indicates that the size of the total promotion effect and whether it persists after a promotion ends are affected by promotion length and depth. For example, the solid curve in the left panel corresponds to a promotion with a depth of 0.25 and a length of five days, whose total promotion effect is fairly large and remains positive after day 12 (seven days after the promotion ends). This promotion ends when the total promotion effect is still strong; therefore, the app enters the postpromotion period with a high rank (visibility). The intertemporal dependence of download volume and app rank, as captured by our VAR model, keeps the download volume at a relatively high level for several days, before it drops back to the normal level. If the promotion

Figure 6. Total Effect Resulting from a 5-, 10- or 15-day Promotion



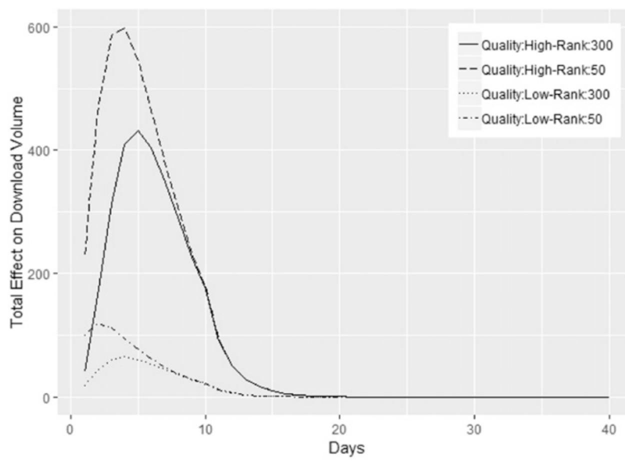
lasts for 15 days (the dotted line in the same plot), the persistent promotion effect (after day 15) is quite small.

Note that, even under the homogeneous-effect demand function estimated using the full data set (column 2 in Table 5), the total long-term effect of a promotion on the absolute download volume (D_{it} as opposed to $\log(D_{it})$) varies with app quality (captured in the app-specific fixed effect) and the $\log(r_{it})$ at the beginning of the promotion. Figure 7 shows the total long-term effect of a ten-day 50%-discount promotion on the absolute download volume in the following four cases: (1) high quality and high rank; (2) high quality and low rank; (3) low quality and high rank; and (4) low quality and low rank.[20] As we can see from the figure, a promotion of the same length and depth has a larger total effect on the absolute download volume for higher-quality and higher-ranked apps. The total promotion effect remains positive after the promotion ends (i.e., after day ten), and the promotion effect persists for a longer period for apps of a higher quality.

4.3.4. Other Coefficient Estimates.

The estimation results also indicate a significant decay in download volume as apps become older—the estimated coefficient for $\log(h_{it})$ is negative and significant. (In Section 4.4, we will show that this result holds when we replace $\log(h_{it})$ with a series of app age dummies by month.) This is not surprising because a sales decay is commonly observed in markets for digital and creative products. The estimated coefficient for $\log(u_{it})$ is also negative and significant, suggesting that the download volume gradually drops after an update. The estimated coefficient for average app rating of the current version is insignificant, likely due to the small within-app rating variation (a variance of 0.15). The estimation result of the month dummies and day-of-week dummies (see Online Appendix F for details) indicates a strong seasonality effect and a weekend effect—the download volume is significantly higher in the first half of the year and during weekends. Finally, after all the coefficients in the demand function are estimated, the app-specific fixed effect for app i (α_i) can be calculated by taking the average of $\log(D_{it}) - (\mathbf{X_{it}}\hat{\boldsymbol{\beta}} + \sum_{m=1}^{11}\hat{\eta}_m M_{mit} + \sum_{w=1}^{6}\hat{\phi}_w W_{wit})$, where $\mathbf{X_{it}}$ is the vector of all covariates in Equation (2) except M_{mit} and W_{wit}, across different periods (t's) for that app. The estimated app-specific fixed effects ($\hat{\alpha}_i$'s) have a mean of 9.73 and a standard deviation of 0.50 across different apps. As mentioned earlier, an app's fixed effect captures all time-invariant characteristics of the app that drive its demand, which could include the time-invariant part of app quality

Figure 7. Total Effect by Quality and Rank



(i.e., base quality). The time-varying part of app quality is largely reflected in changes in an app's rating over time. As we will show later, the magnitude of the app fixed effect plays an important role in the PPP.

4.4. Robustness Checks

We perform a series of robustness checks to ensure that our empirical results are robust to our modeling assumptions and choices. We briefly describe these tests below; the detailed results for each robustness check are provided in Online Appendices D–I.

Alternative Model Specifications.[21] We estimate several variations of the main demand model, which includes (1) a model in which the $\beta_5 \Delta p_{it} + \beta_6 \Delta p_{it} \cdot d_{it} + \beta_7 \Delta p_{it} \cdot d_{it}^2$ term is replaced with an indicator variable that takes the value 1 when an app is on promotion and 0 otherwise; (2) a model in which Δp_{it} (promotion depth) is replaced with ΔP_{it} (absolute price change); (3) a model in which app age is captured by a series of dummy variables, with the kth dummy indicating that the app is k months old, instead of $\log(h_{it})$; (4) a model in which the time-varying immediate promotion effect is captured by a series of promotion-day dummies instead of the polynomial of d_{it}; (5) a model in which $\log(r_{i(t-1)})$ is replaced with $\log(D_{i(t-1)})$; (6) a model in which $\log(D_{i(t-1)})$ is included in addition to $\log(r_{i(t-1)})$; (7) a set of models in which multiple-period lagged values of $\log(r_{it})$ are included; and (8) a model in which the imputed daily rating count is included. The estimation results of these alternative demand models generally suggest that the estimated immediate promotion effect is robust to alternative specifications. Due to the high correlation between $\log(r_{i(t-1)})$ and $\log(D_{i(t-1)})$, we cannot simultaneously include these two variables in the model. We find that the model containing only $\log(r_{i(t-1)})$ has a better prediction accuracy both in- and out-of-sample than the model containing only $\log(D_{i(t-1)}$ and, therefore, use the former as the main model. It should be noted that, in addition to the visibility effect coming from the position an app occupies on the sales chart, the coefficient of $\log(r_{i(t-1)})$ in the main model can also capture the effect of $\log(D_{i(t-1)})$ through non-ranking-related channels. However, different interpretations of the coefficient of $\log(r_{i(t-1)})$ do not affect the setup and solution to the PPP we present later.

Correlated Demand.[22] In the main model, we assume that consumers make the purchase decision independently on each app. In reality, either due to direct substitution or budget constraints, consumers' decision on whether to purchase an app can be correlated with their decision on whether to purchase another app. This possibility is not yet considered in the main model. To see how this assumption affects the empirical results, we perform the following analysis: First, we estimate a discrete choice model, similar to the model in Ghose and Han (2014), which allows substitution between any pair of apps within the same category (all action games in our sample). Note that α_i and β_k in this discrete choice model are parameters in consumers' utility function (at the individual level) and have different meanings than those in Equation (2), which are coefficients in the demand function (at the aggregate level). Therefore, the parameter estimates for the individual utility function cannot be directly compared with the estimated coefficients in the demand function. One important assumption made in the discrete choice model is that in each period a consumer can choose zero apps or one app from the apps available in the choice set (all iOS paid action games), whereas the main model does not impose any assumption on how many apps a consumer can purchase at a time. We then use the estimated choice model and the estimated main model to predict the download volume for apps in a holdout sample; the five-fold cross-validation results for the choice model and our main model are compared in Table 8. As we can see from the table, the main model outperforms the discrete choice model in predicting app demand (i.e., has a smaller out-of-sample mean squared error). As discussed above, these two models can be viewed as two extreme cases and the reality is somewhere between them. The model comparison results suggest that, in our specific empirical context, data are better explained by the main model assuming no direct competition among apps than the choice model assuming mutually exclusive options. One possible reason is that products in the iOS paid action games category are highly differentiated and the substitution effects are not very significant.[23]

To further alleviate the concern related to treating apps as independent of each other, we perform two additional tests. First, we take all the apps that *never* engaged in price promotions in our sample and estimate a modified demand model that includes a new variable, the number of apps that are currently on promotion ($Num_t^{\text{OnPromotion}}$). (All variables with d_{it} in them drop out of the model, because there is no variation in those variables for this subset of apps.) If there is any competition among apps, we would

Table 8. Main Model vs. Choice Model: Out-of-Sample Mean Squared Error for Each Cross-Validation Round

Main model	Choice model
1.655	3.421
1.451	3.679
0.994	3.311
1.908	3.457
3.356	4.521

expect the newly added variable to have a negative impact on the focal app's download volume; that is, apps that are on promotions draw consumers away from the focal app. The estimation results suggest that the number of apps on promotion, in fact, has a positive effect on the focal app's download volume, but the size of this effect is rather small. Second, we estimate our main model with a subset of observations in which app rank is within the top 100. The rationale behind this test is that consumers may be more likely to substitute one highly ranked app with another highly ranked app, than with a lower-ranked app; hence, the competition among this subset of apps may be more intense. The estimated promotion effects in this subsample are very similar to those reported in Table 5. This set of analyses shows that there is little evidence for the substitution effect among apps in our data set. Future studies can consider similar tests and decide whether a correlated or uncorrelated demand assumption is more appropriate in their research contexts. Developing methods that explicitly incorporate the competition among products is a direction for future research.

Alternative Mechanisms.[24] A few alternative models could also capture the intertemporal dependence in demand. One is a model that considers the network effect (Economides 1996). The network effect refers to the positive effect that additional users of a good or service have on the value of that product or service to others, that is, the value of a product or service increases with the number of others using it. Empirical studies of network effects typically examine the relationship between the additional sales and the cumulative sales and use a significant positive relationship between these two variables as evidence for the existence of network effects (Katz and Shapiro 1985, Shankar and Bolton 2004). A more general model that represents the process of the adoption of a product in a population is the Bass model (Bass 1969). Both the network effect and the Bass model describe the relationship between new sales realized in a time period and the *cumulative* sales. We do not include the cumulative sales in our main demand model, because computing the cumulative sales requires the full history of apps' sales (download volume). Our data set only contains the top 500 iOS paid action games from September 1, 2014 to December 31, 2015. For apps that were released before September 1, 2014 and those that did not enter the top 500 chart immediately after their release, we cannot observe their complete sales history. (Only 23% of the apps in our data set entered the top 500 chart within 10 days after their respective releases. Summary statistics for this subsample are provided in Online Appendix H.)

To examine the relative importance of the *visibility effect* versus the *network effect*, we use the subsample of apps that appeared in the top 500 chart within ten days after their respective releases to estimate three alternative models: (1) a model that captures both the visibility effect and the network effect; (2) a model that captures only the visibility effect; and (3) a model that captures only the network effect. The estimation results of these models suggest that the visibility effect plays a much more significant role than the network effect in our data (i.e., the visibility effect explains much more variation in the daily download volume than the network effect). Therefore, ignoring the network effect in the demand model (due to data unavailability) is not a big issue.

Second, in the context of mobile apps, the number of active users of an app (the majority of whom are recent purchasers) may have a stronger effect on the app's future demand than the total number of customers who have ever downloaded the app (Mendelson and Moon 2016), through, for example, word-of-mouth (WoM). We introduce another alternative specification of Equation (2), where $\log(\sum_{k=1}^{10} D_{i(t-k)})$, the logarithm of the total download volume in the past ten days, is added to control for WoM or other effects of the increased active user base. We find that the effect of $\log(r_{i(t-1)})$ is still highly significant after controlling for $\log(\sum_{k=1}^{10} D_{i(t-k)})$, which provides further support for the existence of the visibility effect. The estimated immediate promotion effect remains largely the same in this alternative model as well.

An alternative explanation for the decline in the immediate promotion effect is that app developers may dynamically determine whether to continue a promotion after observing the realized promotion effect. For example, developers may end a promotion as soon as the app gets into a target position on the chart. If a promotion is not effective in the first few days, developers may want to let it run for more days. In that case, the observed declining immediate promotion effect might reflect a selection effect rather than a causal effect. We empirically test whether such a selection effect exists in our data and find no evidence for the selection effect.

4.5. Heterogeneity

In the main model, we assume that promotions of the same depth have a homogeneous immediate effect on $\log(D_{it})$.[25] It is possible that the magnitude of the immediate promotion effect is different for apps with different characteristics. To test this possibility, we perform the following analyses to explore the potential heterogeneity in the immediate effect of promotion on $\log(D_{it})$ with respect to (i) a set of time-invariant app characteristics and (ii) a set of time-varying app characteristics. The former set includes app quality (proxied by app all-time average rating and ranking), original price, and whether the app has a free version available or not; the latter set includes the average app rank seven days prior to the promotion, app age at the time of the promotion, and the time since the last update at the time of the promotion. We explore the potential heterogeneity in the immediate promotion effect with respect to (i) using stratified analysis to allow for maximum flexibility and (ii) including interaction effects.[26] Finally, to see whether the immediate promotion effect varies with calendar time, we divide the sample period into four equal-length intervals and estimate the main model with data falling in each interval separately. For brevity, we briefly describe the findings from these analyses here; detailed results are provided in Online Appendix I. We find that the immediate promotion effect is generally larger for apps that are more expensive (higher original price), newer, and without a free version; the effect varies little with app quality (measured by either the average app rank or the average app rating), the average rank within seven days prior to the promotion, and the recency of the last update. We also find that the immediate promotion effect is larger in the first half of the sample period (September 2014–April 2015) than in the second half (May–December 2015).

4.6. Prediction Accuracy

Since the estimated demand function and ranking function are the main inputs of the promotion planning optimization, it is important to ensure that our main model can accurately predict the sequence of the download volume given a price sequence. We perform the following prediction accuracy test: For each app in our data set, we use the first 70% of observations (in time) as the training set and the last 30% of observations as the testing set. We estimate the main model based on the training set and use the estimated model to generate predictions for observations in both the training set (in sample) and testing set (out of sample). We check if each observed daily download volume in the training/testing set falls into the 85%, 90%, and 95% prediction intervals for $\log(D_{it})$. If so, we count this prediction as a *hit*; otherwise, it is a *miss*. The prediction accuracy is then calculated as No. of hits/(No. of hits + No. of misses). The in-sample and out-of-sample prediction accuracy is tabulated in Table 9.[27] We find that the model-predicted download volumes are close to the observed download volumes, and the prediction intervals have a reasonable width.

Table 9. Prediction Accuracy

Prediction interval	85%	90%	95%
In-sample prediction accuracy	87.18%	91.15%	94.95%
Out-of-sample prediction accuracy	86.58%	91.30%	94.74%
Mean interval width (in-sample)	1.677	1.916	2.283
Mean interval width (out-of-sample)	1.677	1.916	2.284

(See Online Appendix J for a comparison of two sample apps and a detailed discussion of Table 9.) This set of results, combined with the small out-of-sample mean squared error presented in the first column of Table 8, shows that our model performs well in predicting download volume. Note that the high prediction accuracy also implies that the monopolistic assumption we make does not adversely affect the estimation of the demand function; the PPP and the MPW heuristic to be introduced can be adapted to any functional form of the demand model.

5. Promotion Planning Problem

In this section, we apply the empirical results to the optimization of price promotions for mobile apps. Our goal is to optimize the promotion schedule (including starting time and duration) as well as the promotion depth.[28] We formulate the PPP based on the demand system estimated in Section 4.1. The PPP is then transformed into an LPP. Due to the NP-hardness of the LPP, we further propose an MPW heuristic that can effectively find near-optimal solutions to the LPP. Finally, we numerically show the performance of the proposed heuristic for apps with different initial ranks, ages, and original prices.

5.1. PPP Model Formulation

For notational simplicity, we drop the app index i and simplify the demand function and ranking function estimated in Section 4.1.[29] The PPP can be formulated as shown below.

Promotion Planning Problem. We have

$$\max_{p_t, t=1,\cdots,T} \sum_{t=1}^{T} D_t p_t \tag{4}$$

$$\begin{aligned}\text{s.t. } \log(D_{it}) = {} & \alpha_i + \beta_1 \log\big(r_{i(t-1)}\big) + \beta_2 \log(h_{it}) \\ & + \beta_3 \log(u_{it}) + \beta_4 q_{it} + \beta_5 \Delta p_{it} \\ & + \beta_6 \Delta p_{it} \cdot d_{it} + \beta_7 \Delta p_{it} \cdot d_{it}^2 \\ & + \sum_{m=1}^{11} \eta_m M_{mit} + \sum_{w=1}^{6} \phi_w W_{wit} + \epsilon_{it},\end{aligned} \tag{5}$$

$$\log(r_{it}) = \gamma_0 + \gamma_1 \log(D_{it}) + \varepsilon_{it}, \tag{6}$$

$$h_t = h_{t-1} + 1, \tag{7}$$

$$\Delta p_t = \begin{cases} \frac{p_{t-1}-p_t}{p_{t-1}} & p_t < p_{t-1}, \\ \Delta p_{t-1} & p_t = p_{t-1}, \Delta p_{t-1} > 0, \\ 0 & p_t > p_{t-1} \text{ or } p_t = p_{t-1}, \Delta p_{t-1} = 0, \end{cases} \tag{8}$$

$$d_t = \begin{cases} d_{t-1} + 1, & \Delta p_t > 0 \\ 0, & p_t > p_{t-1} \text{ or } p_t = p_{t-1}, \Delta p_{t-1} = 0, \end{cases} \tag{9}$$

$$p_t \in \{0.99, \cdots, p_0\} \text{ if } p_{t-1} = p_0, \tag{10}$$

$$p_t \in \{p_{t-1}, p_0\} \text{ if } p_{t-1} < p_0. \tag{11}$$

We assume that the original price (p_0) is set by the developers prior to the promotion optimization. Constraints (10) and (11) describe the feasible price choices given the last period price. When an app is at its original price (p_0), the developer can choose to offer a promotion on day t by choosing a p_t that is lower than p_0 or keeping the original price ($p_t = p_0$). Otherwise, when an app is already on promotion ($p_{t-1} < p_0$), the developer can only choose to keep running the promotion for the current day ($p_t = p_{t-1}$) or increase the price back to the original level ($p_t = p_0$). When the price returns to its original level, we set both Δp_t and d_t back to zero until the next promotion starts. (See Equation (8) and Equation (9) for the complete specification of Δp_t and d_t.)

This formulation is difficult to solve directly since it is neither linear nor convex. Therefore, we bring the idea of LPP into solving the PPP. Below, we describe the intuition of the LPP formulation; technical details are provided in Online Appendix K. In the LPP, we build a graph by defining the nodes as the state (S_t) of an app at time t, which is defined by the combination of its current ranking, price, promotion status, and other time-varying characteristics. For example, suppose an app is at state S_t and it has been on promotion for four days ($d_t = 4$) with a discount price of \$2.99 and an original price of \$3.99 ($\Delta p_t = \frac{1}{3.99}$). The decision the app developer makes is whether to continue the promotion or end the promotion and return to the original price on day $t + 1$. If the promotion continues, the state of the app on day $t + 1$ (S_{t+1}) will evolve to $d_{t+1} = 5$ and $\Delta p_{t+1} = \frac{1}{3.99}$. If the promotion ends, then S_{t+1} will be $d_{t+1} = 0$ and $\Delta p_{t+1} = 0$.[30] These two cases create two alternative *nodes* in state S_{t+1} connecting from the same node in S_t. We then assign *weights* to each arc connecting these nodes as the revenue that an app can generate in a day by following the defined pricing policy. We formulate the LPP by building the graph with all possible nodes that can be reached given a starting status and connecting them with weights assigned to each arc. To find the revenue-maximizing pricing policy for the next T days starting from state S_0, we need to first find all the paths that connect the starting node to one of the nodes in S_T. Then we sum all the weights along each path and find the path with the maximum total weight. The total weight is equal to the total revenue the app can get by following that price trajectory. On any given day we can keep the current price or change it (either raise or drop); hence, at least two nodes in the next time period connect from the current node. Therefore, the number of nodes in state S_T grows exponentially as T gets larger.

With limited computational power, we cannot build the full graph with every single possible price trajectory for a large T and solve the problem exactly. To efficiently find a near-optimal pricing policy, we propose an MPW heuristic. For a PPP within the planning horizon of T days,[31] the MPW heuristic deals with subproblems of a small *planning window* of T_W days. We then solve the subproblems sequentially by moving the planning window forward with a *step size* of n ($n < T_W$) days each time. We assume that the developer will follow the optimal pricing policy solved from the subproblem for the next n days. We can then get the initial point of the next subproblem using Equations (5) and (6).[32] The next subproblem will then be solved given the new initial state for the next T_W days. These steps are repeated until the planning window covers day T, which is predetermined by the developer. The pricing policy from the heuristic is the price trajectory that the developer follows along the way of solving these subproblems. (See Online Appendix L for more details of the heuristic.) In our numerical results, we show that the MPW can get a near-optimal objective value while significantly saving computational time.

Fatigue. In our basic LPP, every promotion is assumed to be independent of each other, even if one promotion immediately follows another (i.e., two promotions are offered within a short period of time). Under this assumption, the optimal promotion plan will suggest that developers conduct promotions as frequently as possible, increasing the price back to its original level for one day and then immediately conducting the next promotion to exploit the largest promotion effect on the first day of each promotion.

In practice, however, when two promotions are offered back to back, the second one may not be as effective as the first (Blattberg et al. 1995, Kumar and Pereira 1995). The reduced effectiveness of the second promotion is often referred to as *fatigue*. As shown in the empirical analysis (Figure 5), the promotion effect drops with d_{it}. When two promotions are offered back to back, the second promotion is effectively an extension of the first; therefore, its promotion effect may be smaller in magnitude. To make our model more realistic, we introduce *fatigue* into our model. Let $\mathcal{F}(s_t)$ denote the decrease in the second promotion's baseline effect (β_5) if it starts s_t days after the previous promotion ends. Ideally, we would like to estimate $\mathcal{F}(s_t)$ using real-world data. However, in our data set, promotions are infrequent, and back-to-back promotions are even rarer. Among all action games in our sample, only 90 games had more than one promotion during the study period, and the average time gap between two promotions is 68 days. Due to the limited variation in the gap between two consecutive promotions in the data and the lack of data for cases where the gap is small, we are unable to empirically estimate $\mathcal{F}(s_t)$. Note that the fatigue function should have the following properties: $\mathscr{F}(s_t) \le 0, \forall s_t$ and $\mathscr{F}(s_t)$ is increasing in s_t. Other than having these properties, the functional form of such fatigue is not restricted (see Online Appendix M for two examples of fatigue functional forms). As $s_t \to \infty$, two consecutive promotions can be viewed as independent of each other, and the baseline promotion effect will converge to 1.173 ($\hat{\beta}_5$ reported in Table 5). Clearly the total revenue from a given promotion policy can be affected by the scale of the fatigue. However, we can still compare the revenue across alternative promotion policies under the same scale of fatigue. In the following numerical analysis, we use a fatigue function of $\mathscr{F}(s_t) = \zeta \cdot \log(1 + 1/s_t)$ and set the scale parameter ζ to 1.173/log(2).

6. Promotion Planning Result

In this section, we compare the performance of our proposed MPW heuristic against a number of alternative policies. We show that our heuristic outperforms the other promotion policies in most cases. Moreover, the heuristic can effectively provide a near-optimal solution for the NP-hard PPP introduced in Section 5.1.

6.1. Window Size Sensitivity

Before applying the MPW to mobile apps in our data set, we show the performance of the heuristic under various window sizes and compare the outcome with the first best. The heuristic is tested on a hypothetical *average app* where the seasonality/day-of-week variables and the app fixed effect are set to the corresponding average values across all the apps in our data set. We then construct a set of initial states by iterating over the set of initial ranks $\{20, 30, \cdots, 100\}$, app ages $\{200, 250, \cdots, 600\}$, and original prices $\{\$1.99, \cdots, \$8.99\}$. We use MPWs with a window size of $2, 4, \cdots, 12$ to solve for the promotion policy for the next 12 days. Thus, the outcome of the promotion policy where $T_W = 12$ is the best for this small-scale problem.[33] In the left panel of Figure 8, we show that the average revenue across different initial states is near-optimal when the window size is above four days. Meanwhile, the right panel of Figure 8 shows that a smaller window size can save a significant amount of computational time because it reduces the number of price trajectories that need to be evaluated. In the examples below, where a larger problem is being solved (T is larger), we use a window size $T_W = 10$ for the computational efficiency of our numerical tests.

6.2. Performance Comparison Among Alternative Policies

We use a sample app in our data set to explain how the performance comparison is carried out. The exact optimal promotion strategy and its resulting improvement is app specific, depending on its age, rank, and quality (captured in the app fixed effect). Figure 9 shows a performance comparison among several alternative promotion policies on an app that has an original price of \$2.99 and is on the top 500 chart for 86 days. Under the current pricing strategy, the app is on promotion at \$0.99 for 14 days at the beginning of the study period, then maintains a constant price of \$2.99 afterward. In this example, the total revenue throughout the 86-day time horizon is \$10,173 under the current practice. With our MPW heuristic, the simulated total revenue is \$12,588. Note that we use the real app age and version age, and the estimated month and day-of-week dummies to simulate app sales under the MPW policy. The result of our MPW heuristic suggests that the app could benefit more from shorter but more frequent promotions.

Another observation from Figure 9 is that the actual download volume of an app contains some unobservable random disturbances. To ensure a fair comparison, we need to either introduce these disturbances into our simulation of the performance of the constant-price, MPW, and random policies or remove these disturbances from the data for the current policy. For ease of implementation, we choose to remove the

Figure 8. Performance of the MPW with Various Window Sizes ($T = 12$); Left Panel: Average Revenue from MPW as Window Size Increases; Right Panel: Number of Possible Pricing Trajectories Need to Be Evaluated as Window Size Increases

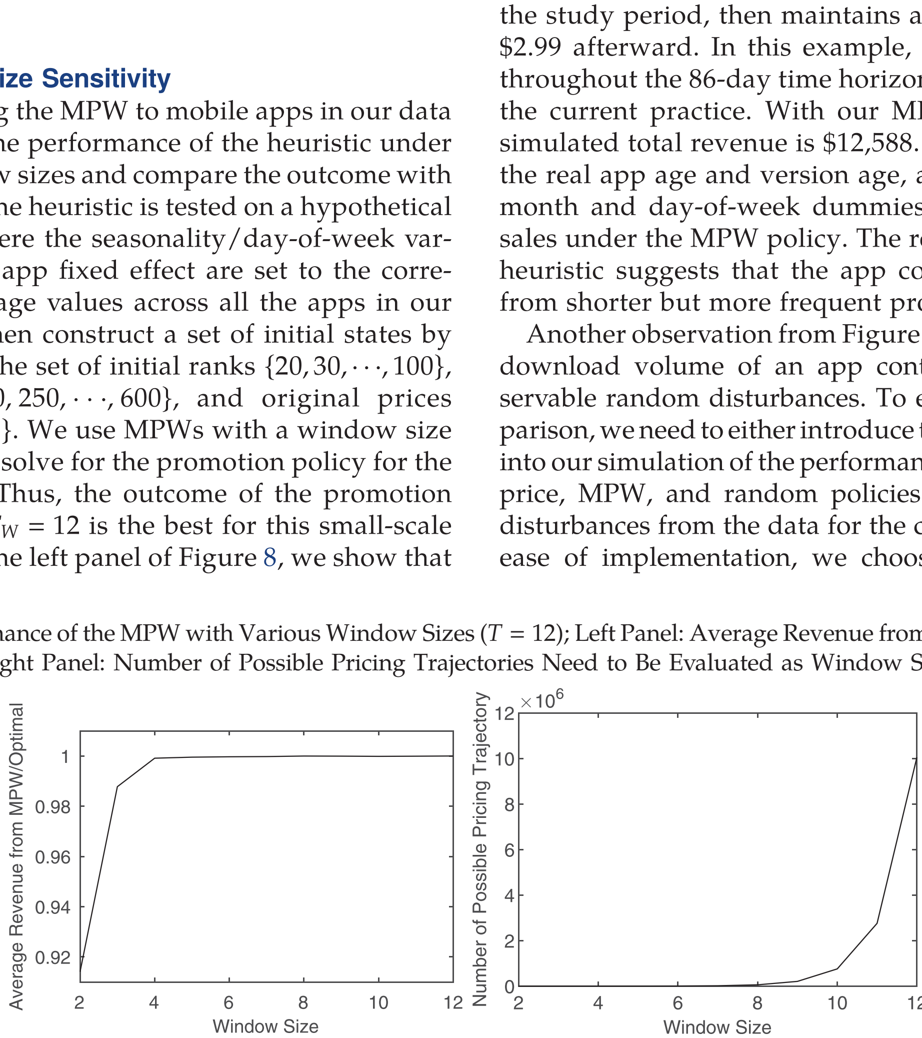
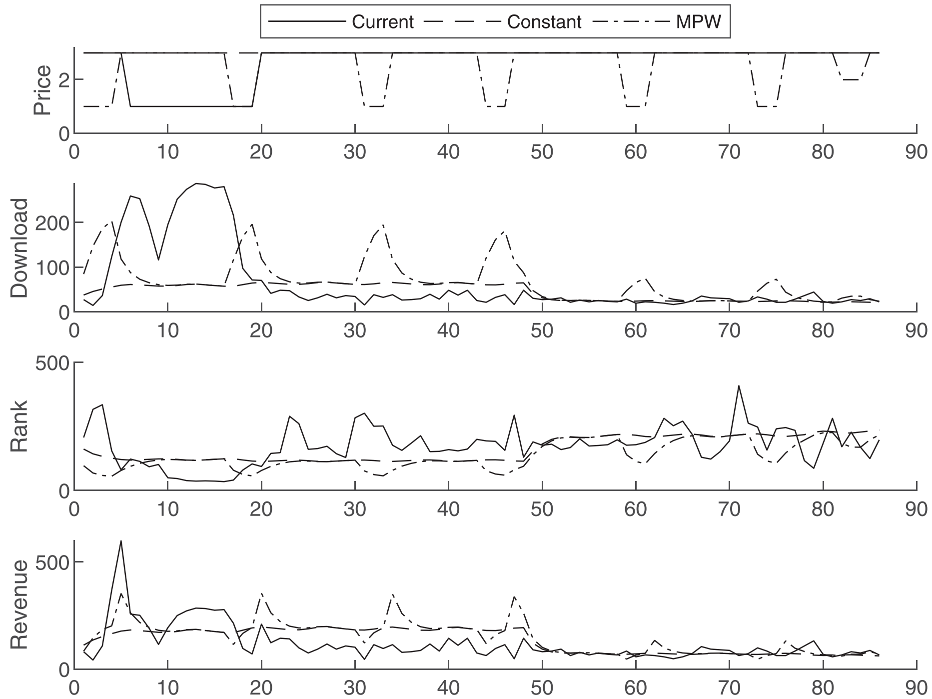


Figure 9. Performance Comparison Between MPW, Random Policy and Current Practice for an App with p_0 = \$2.99; Fatigue Scale $\zeta = \beta_5/\log(2)$, Plan Window $T_W = 10$



disturbances. We feed the observed daily app price into our simulation algorithm to generate the total revenue under the current policy with the random disturbances removed. We then compare the performance among (1) the constant-price policy; (2) the policy suggested by the proposed MPW heuristic; (3) the random promotion policy;[34] and (4) the current practice, with (1) as the baseline. We do this for all the apps in our data set with an original price of \$1.99 or higher[35] and report in Table 10 the statistics of the total lifetime revenue under each of the four policies, all as a *fraction* of the revenue under the constant-price policy. The results show that both the current promotion policy and the MPW-suggested policy outperform the constant-price policy on all apps, and the MPW policy has a higher total revenue than the current practice on average. The MPW policy improves the total revenue by about 10% over the constant-price policy and the current practice, on average. We further conduct a sensitivity analysis for the performance comparison and find that the MPW heuristic generates the highest revenue under various app conditions (see Online Appendix N for details). Additionally, we find that the random policy performs worse than the constant-price policy, highlighting the importance of carefully designing price promotions, and demonstrating the value of our MPW heuristic.

The spikes on the MPW curve in the bottom plot of Figure 9 reflect the revenue increase after the promotion ends when app price returns to its normal level. Our MPW heuristic does not necessarily generate a higher revenue during the promotion than other policies; however, it does a better job of balancing the potential revenue loss due to a lowered price and the significantly increased download volume during the promotion period and taking advantage of the carry-over effect of a higher download volume and sales rank in the postpromotion period when the app price is returned to the original level.

Table 10. Performance Comparison with Real App Data

Statistic	Mean	Standard deviation	Minimum	25th percentile	75th percentile	Maximum
Constant price (baseline)	1	1	1	1	1	1
Current practice	1.061	0.593	1	1	1	9
MPW	1.185	0.315	1.077	1.098	1.168	4.799
Random policy	0.890	0.247	0.394	0.771	1.000	3.696

As a result, the MPW heuristic can generate higher long-term revenue.

In the previous analysis, we have not considered potential additional revenues from in-app purchases or in-app advertising. According to Ghose and Han (2014), in-app purchase options have a positive direct impact on download volume, whereas in-app advertising has a negative direct effect on download volume. Moreover, both in-app purchases and in-app advertising can increase the revenue from each unit of sales. Although these in-app features are less prevalent among paid apps, we perform a series of analyses to examine how these features may affect the optimal promotion schedule (see Online Appendix O for details). We find that the optimal promotion policy suggested by the MPW heuristic is not significantly affected by the inclusion of these in-app features. It is only when in-app purchases can generate a large amount of additional revenue relative to the upfront purchase price that the MPW heuristic will suggest a slightly different promotion timing.

6.3. Significance of the Visibility Effect and Dynamic Promotion Effect

Next, we discuss the characteristics of the promotion schedule prescribed by our MPW heuristic. Figure 10 shows a 50-day promotion schedule generated by the MPW heuristic for a hypothetical app released 200 days ago with an initial rank of 20 and an original price of \$2.99. A few observations about the optimal promotion policy are worth noting. First, the MPW heuristic suggests that promotions be offered more frequently than what we observed in the data. Within a 50-day planning horizon, our MPW heuristic suggests that promotions with a duration of three to seven days be offered four times, whereas most apps in our data had only one to three promotions within a 16-month period (Figure 2). One possible explanation for the infrequent promotions in practice is that developers underestimate or ignore the visibility effect. In fact, when the visibility effect is removed from the demand function, the optimal policy is to keep the original price throughout the 50-day planning horizon (Figure 10, MPW without visibility). Second, as shown in Table 11, assuming constant promotion effect (MPW without dynamic promotion) will lead to a significant loss in revenue.

7. Conclusion

We propose a two-step data-analytic approach, combining econometric methods and optimization techniques, for mobile apps' promotion planning. In the first step, we empirically examine the short-term and long-term effects of price promotions on the download volume of mobile apps, using historical data of 377 iOS apps in the paid action game category that appeared in the top 500 chart for that category in a 16-month study period. Our empirical analysis reveals and quantifies the declining immediate effect of price promotions on app sales and the visibility effect. The latter is the underlying mechanism driving the long-term effect of price promotions. In the second step, we formulate the PPP as a LPP based on the demand function estimated in the first step.

Figure 10. Comparison Among MPW, MPW Without Visibility Effect (Constant Price), and MPW Without Dynamic Promotion Effect for an App with $r_0 = 20$, $h_0 = 200$, $p_0 = \$2.99$; Fatigue Scale $\zeta = \beta_5/\log(2)$, Plan Window $T_W = 10$

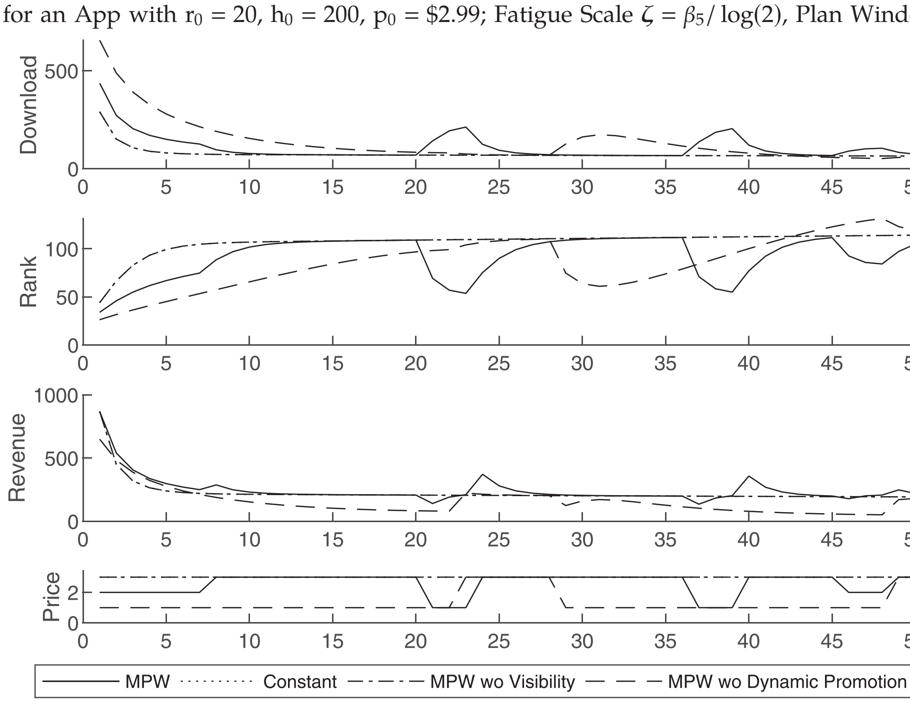


Table 11. Total Revenue Under Different Pricing Policies Relative to Revenue Under Constant-Price Policy

Statistic	Mean	Standard deviation	Minimum	25th percentile	75th percentile	Maximum
MPW	1.108	0.005	1.094	1.105	1.112	1.114
MPW without visibility	1.009	0.004	1	1.0	1.0	1
MPW without dynamic promotion	0.749	0.040	0.678	0.723	0.768	0.832

To overcome the complexity of the problem, we propose a MPW heuristic to numerically solve the PPP. The existence and magnitude of the visibility and the time-varying promotion effect significantly affect the solution to apps' PPP. We find that if the visibility effect is underestimated or ignored, the optimal pricing policy is to keep a constant price throughout an app's life cycle. If the decline in the immediate promotion effect over time is ignored, developers tend to offer longer promotions. As a result, the postpromotion revenue will be lower, because at the end of the promotion, the rank of the app is not as high as it would have been if the promotion were shorter. In this case, the visibility effect is not taken advantage of in the postpromotion period.

Our study makes significant contributions to both theory and practice. First, our work is among the first to study the PPP for (paid) mobile apps. We show that shorter and more frequent promotions work better for mobile apps due to the declining immediate promotion effect and the existence of the visibility effect. Second, our data-driven promotion planning policy outperforms current common practice in the industry, demonstrating the power of combining econometric analysis and optimization techniques to design effective price promotions. Our modeling framework (including the demand estimation and PPP) is completely driven by real data and imposes minimal assumptions on the demand function. The proposed heuristic can be conveniently used by app developers[36] to achieve real-time promotion planning. Our proposed framework is readily applicable to the promotion planning for other digital products that exhibit similar promotion effects and visibility effects, such as desktop software. It may also be applied to physical goods sold on e-commerce platforms where sales rank affects the position on the platform's website in which products are displayed, especially those that can be considered one-time purchase products. We caution that the specific empirical model needs to be updated and re-estimated when the framework is applied to other digital goods or physical goods sold on e-commerce platforms whose demand is affected by a different set of factors and that our algorithm assumes changing the price is costless and has an objective of maximizing revenue (which is equivalent to maximizing profit when the marginal cost is zero). In contexts where these assumptions/objectives do not apply (e.g., producers/retailers have limited flexibility to make price changes, pricing decisions are jointly made by multiple parties with different objectives, or the marginal cost is not zero), modifications to the setup of PPP are also necessary.

Our study has several limitations. First, due to data unavailability, the fatigue function used in the PPP is based on prior theoretical work, but not directly estimated from the real-world data. Future research could consider collecting relevant data or running field experiments to estimate a more realistic fatigue function or test the proposed promotion planning heuristic in the field.[37] Second, since the range of prices observed in our data set is from \$0.99 to \$7.99, the estimated demand function may be unable to describe the relationship between price promotions and app demand when app price is higher than \$7.99. Although we do not expect dramatic differences in the shape of the app demand function in the price range of (\$7.99, $+\infty$), the estimated demand function is worth additional empirical validation. Third, we are not able to pinpoint the exact mechanism for the decreasing immediate effect of promotions on app demand with current data. Finally, we do not have information about when apps are featured or recommended, and for this reason we interpret the estimated visibility effect as a combination of the direct visibility effect from moving to a higher position in the sales chart and the possible indirect visibility effect through the increased chance of being featured or recommended (and perhaps even the positive effect of an increased user base through non-ranking-related channels). In spite of these limitations, our study generates important insights into the design of effective price promotions for mobile apps and shows the advantages of using data-driven models for solving PPPs, especially for products in markets that exhibit the visibility effect, where future demand depends on past sales rank.

Acknowledgments
This work was done while Amitabh Sinha was at the University of Michigan, and completed prior to joining Amazon.

Endnotes
[1] That is, once created, there is practically no recurring manufacturing, shipping, and storage cost.

[2] More than 60% of the apps in our sample use a constant-pricing policy.

[3] An important implication of this sample selection is that the demand function estimated in this paper is directly applicable to apps in the top 500 chart in the paid action game category; its prediction of the demand for apps ranked below 500 is subject to extrapolation. It also implies that we do not have data for the days on which an app is not on the top 500 chart. We examine how much the missing observations affect the estimation results and find the effect is small. The details of this test can be found in Online Appendix A.

[4] The Apple iOS App Store's ranking is unique for free and paid apps. If a paid app's price drops to zero, it will be moved from the paid category to the free category and lose its ranking in the paid chart. (If the price of a paid app drops but the app remains a paid app, it will not lose its ranking in the paid chart.)

[5] We try two alternative cutoffs, 10 and 15 observations, and show that the empirical results are robust to the selection of the cutoff value. See Online Appendix B for the detailed estimation results.

[6] The price tier for other countries can be found at http://www.equinux.com/us/appdevelopers/pricematrix.html (accessed on July 15, 2015).

[7] See Online Appendix C for details.

[8] We interviewed a few iOS app developers, and they all believe that the current-period download volume is the main determinant of app rank at the end of the current period, although the exact ranking algorithm is not public.

[9] This regression has not accounted for promotion depth and length and app characteristics, and we do not intend to make any conclusive statements based on the estimation results of this regression. Later we will construct a vector autoregressive model, which describes the *mechanism* driving the effects found in the estimation results of Equation (1), as the main model. Understanding the mechanism driving the promotion effects is important for app promotion optimization.

[10] Not all promotions last 25 days; that is, not all apps "visit" each point on the figure.

[11] The coefficient of lagged rank captures not only the direct visibility benefits brought by moving from a lower to a higher position on the app store's download chart, but also the indirect visibility benefits resulting from being featured in app stores. App store editors periodically select popular apps and display them on the app stores' homepage as "featured apps" or "editor's choices" and recommend them on social media platforms. Additionally, since $\log(r_{i(t-1)})$ and $\log(D_{i(t-1)})$ are highly correlated, the visibility effect, reflected by the coefficient of $\log(r_{i(t-1)})$, may also capture the effect of $\log(D_{i(t-1)})$ through non-ranking-related channels.

[12] We considered a few alternative specifications, including one that contains more lagged values of $\log(r_{it})$, one that uses $\log(D_{i(t-1)})$ to replace $\log(r_{i(t-1)})$, and one that incorporates both $\log(D_{i(t-1)})$ and $\log(r_{i(t-1)})$. The estimation results of these alternative models and the model comparison are briefly discussed in Section 4.4; the details are provided in Online Appendix D.

[13] We consider an alternative demand function that includes the absolute price change instead of the fractional price change. The model has a slightly worse fit as compared with the main demand model (Equation (2)); see Section 4.4 for details of this analysis.

[14] For example, consider a seven-day window and fix the promotion depth to 0.5. A promotion that starts on the second day and lasts for three days would result in a sequence of Δp_{it} of $\{0, 0.5, 0.5, 0.5, 0, 0, 0\}$; a promotion that starts on the third day and lasts for five days would result in a sequence of Δp_{it} of $\{0, 0, 0.5, 0.5, 0.5, 0.5, 0.5\}$. Note that d_{it} is simply the natural day count for days on which $\Delta p_{it} > 0$.

[15] Like Ghose and Han (2014), we do not consider cross-platform demand correlation through media channels shared by users of both platforms, which plausibly exists in reality. In addition, we do not explicitly consider any marketing campaigns that app developers conduct in conjunction with price promotions.

[16] See Online Appendix C for details of this analysis.

[17] We also obtain a three-stage least squares (3SLS) estimator for the system of equations, which exploits the correlation of the disturbances across equations and, thus, is asymptotically more efficient, but at the same time is less robust to any model misspecifications in one part of the system. We find that the 3SLS estimator is very similar to the 2SLS estimator. Although 3SLS has some efficiency benefits in that it reduces the standard errors of the coefficient estimates, the model as a whole has a poorer fit. See Online Appendix E for a detailed comparison between the 2SLS estimator and 3SLS estimator.

[18] These are only potential explanations (speculations) for the decreasing immediate promotion effect over time within a single promotion; identifying the underlying mechanism for the decreasing immediate promotion effect is beyond the scope of this paper and cannot be achieved with our current data set.

[19] Due to data limitations, we cannot separately estimate the effects of (1), (2), (3), and (4). However, the exact mechanism underlying the estimated visibility effect will not affect the PPP that we will introduce later.

[20] A low- (high-) quality app is defined as an app with a fixed effect that equals the first (third) quartile of the estimated app fixed effects across all apps. The calculation of the app fixed effect is discussed in Section 4.3.4. In reality, the "low-quality and high-rank" case is unlikely to exist, as a low-quality app is unlikely to reach a high rank.

[21] Details of this set of analyses are provided in Online Appendix D.

[22] Details of this set of analyses are provided in Online Appendix G.

[23] Solving an optimization model that considers competitors' actions is much more challenging.

[24] Details of this set of analyses are provided in Online Appendix H.

[25] As shown in Figure 7, it does not imply that the total effect of promotions on the absolute download volume is homogeneous.

[26] Because Δp_{it} is potentially endogenous, any interaction terms that contain Δp_{it} are also potentially endogenous.

[27] The prediction accuracy for an app is significantly affected by the number of observations we have about that app. Out-of-sample prediction is very challenging for apps that have only a few observations in our sample.

[28] Since all price changes in our data set take the form of price promotions, we do not have data to empirically examine the effect of an increase in app price that goes beyond the original price. Therefore, our PPP formulation does not allow increasing app price beyond the original price. We also assume that the original price for each app is predetermined. Determining an app's original price requires a different analysis and is outside the scope of this study.

[29] We also drop the error term from both models for the PPP formulation. Including the error term in the demand model does not significantly affect the proposed heuristic.

[30] The evolution of other state variables such as the month dummies, app age, and the number of days since the last update will be the same in both cases.

[31] We assume that developers will ignore the sales from day $T+1$ onward in the PPP.

[32] We focus on the rank and the download volume here, since all the other variables can be updated automatically without considering the promotion policy.

[33] Again, due to the NP-hardness of the LPP, it is not feasible to exactly solve a large-scale problem.

[34] The random policy is defined as follows. In each time period, if an app is not currently on promotion, the developer of the app would randomly choose a price level that is lower than or equal to its normal price;

if the app is already on promotion, the developer would randomly decide whether to end the current promotion.

[35] We do not consider free promotions in our study; thus, the apps with an original price of $0.99 are removed from this comparison.

[36] App developers can update their estimate of the demand function using data on their own apps (or the data that they believe is the most relevant). The ranking function is not app specific, but category specific; app developers can directly use the estimated ranking function in this paper if the app for which they are designing the promotion plan is in the paid action game category. To estimate the effect of promotions, app developers can also use historical data or experiment with different promotion depths and lengths to get their app-specific estimate. Note that, even if a better-calibrated model is found, the formulation and solution for the PPP and the structure of the heuristic proposed in the paper remain valid; the parameters or the demand model simply need to be retuned or modified.

[37] The procedure to solve for the optimal promotion policy will remain the same regardless of the changes in the fatigue function.

References

Agarwal A, Hosanagar K, Smith MD (2011) Location, location, location: An analysis of profitability of position in online advertising markets. *J. Marketing Res.* 48(6):1057–1073.

Appel G, Libai B, Muller E, Shachar R (2020) On the monetization of mobile apps. *Internat. J. Res. Marketing* 37(1):93–107.

Askalidis G (2015) The impact of large scale promotions on the sales and ratings of mobile apps: Evidence from Apple's app store. Preprint, submitted June 23, https://arxiv.org/abs/1506.06857.

Assuncao JL, Meyer RJ (1993) The rational effect of price promotions on sales and consumption. *Management Sci.* 39(5):517–535.

Bass FM (1969) A new product growth for model consumer durables. *Management Sci.* 15(5):215–227.

Berry S, Levinsohn J, Pakes A (1995) Automobile prices in market equilibrium. *Econometrica*. 63(4):841–890.

Bils M, Klenow PJ (2004) Some evidence on the importance of sticky prices. *J. Political Econom.* 112(5):947–985.

Bitran G, Caldentey R (2003) An overview of pricing models for revenue management. *Manufacturing Service Oper. Management* 5(3):203–229.

Blattberg RC, Briesch R, Fox EJ (1995) How promotions work. *Marketing Sci.* 14(3):G122–G132.

Carare O (2012) The impact of bestseller rank on demand: Evidence from the app market. *Internat. Econom. Rev.* 53(3):717–742.

Chaudhari H, Byers J (2017) Impact of free app promotion on future sales: A case study on Amazon Appstore. Preprint, submitted November 27, https://ssrn.com/abstract=3078067.

Chen H, Hu YJ, Smith MD (2018) The impact of e-book distribution on print sales: Analysis of a natural experiment. *Management Sci.* 65(1):19–31.

Chevalier J, Goolsbee A (2003) Measuring prices and price competition online: Amazon.com and Barnes and Noble.com. *Quant. Marketing Econom.* 1(2):203–222.

Cohen MC, Leung NHZ, Panchamgam K, Perakis G, Smith A (2017) The impact of linear optimization on promotion planning. *Oper. Res.* 65(2):446–468.

Danaher B, Huang Y, Smith MD, Telang R (2014) An empirical analysis of digital music bundling strategies. *Management Sci.* 60(6):1413–1433.

Economides N (1996) The economics of networks. *Internat. J. Indust. Organ.* 14(6):673–699.

Garg R, Telang R (2013) Inferring app demand from publicly available data. *Management Inform. Systems Q.* 37(4):1253–1264.

Ghose A, Han SP (2014) Estimating demand for mobile applications in the new economy. *Management Sci.* 60(6):1470–1488.

Ghose A, Goldfarb A, Han SP (2012a) How is the mobile Internet different? search costs and local activities. *Inform. Systems Res.* 24(3):613–631.

Ghose A, Ipeirotis PG, Li B (2012b) Designing ranking systems for hotels on travel search engines by mining user-generated and crowdsourced content. *Marketing Sci.* 31(3):493–520.

Ghose A, Ipeirotis PG, Li B (2014) Examining the impact of ranking on consumer behavior and search engine revenue. *Management Sci.* 60(7):1632–1654.

Hamilton J (1994) *Time Series Analysis* (Princeton University Press, Princeton, NJ).

Han SP, Park S, Oh W (2016) Mobile app analytics: A multiple discrete-continuous choice framework. *MIS Quart.* 40(4):983–1008.

Horváth C, Fok D (2013) Moderating factors of immediate, gross, and net cross-brand effects of price promotions. *Marketing Sci.* 32(1): 127–152.

Ifrach B, Johari R (2014) The impact of visibility on demand in the market for mobile apps. Preprint, submitted February 11, https://ssrn.com/abstract=2444542.

Katz ML, Shapiro C (1985) Network externalities, competition, and compatibility. *Amer. Econom. Rev.* 75(3):424–440.

Kumar V, Pereira A (1995) Explaining the variation in short-term sales response to retail price promotions. *J. Acad. Marketing Sci.* 23(3): 155–169.

Lambrecht A, Goldfarb A, Bonatti A, Ghose A, Goldstein DG, Lewis R, Rao A, Sahni N, Yao S (2014) How do firms make money selling digital goods online? *Marketing Lett.* 25(3):331–341.

Lee G, Raghu TS (2014) Determinants of mobile apps' success: evidence from the app store market. *J. Management Inform. Systems* 31(2):133–170.

Lee GM, He S, Lee J, Whinston AB (2020) Matching mobile applications for cross promotion. *Inform. Systems Res.* Forthcoming.

Liu CZ, Au YA, Choi HS (2014) Effects of freemium strategy in the mobile app market: an empirical study of google play. *J. Management Inform. Systems* 31(3):326–354.

Loadown T (2014) Using version updates and price changes to improve mobile app discovery. Aceesed May 12, 2015, https://docs.google.com/file/d/0B9Pjy-KrfwRCZ1VwZFpxZHZIN2c/edit.

Mendelson H, Moon K (2016) Growth and customer loyalty: Evidence from the app economy. Preprint, submitted November 26, https://ssrn.com/abstract=2777764.

Nan G, Wu D, Li M, Tan Y (2016) Optimal pricing of information goods under the freemium strategy in the presence of piracy and network effects. Preprint, submitted June 20, https://ssrn.com/abstract=2798421.

Nijs VR, Dekimpe MG, Steenkamps JBE, Hanssens DM (2001) The category-demand effects of price promotions. *Marketing Sci.* 20(1):1–22.

Oh H, Animesh A, Pinsonneault A (2016) Free vs. for-a-fee: The impact of a paywall on the pattern and effectiveness of word-of-mouth via social media. *MIS Quart.* 40(1):31–56.

Pashigian BP (1988) Demand uncertainty and sales: A study of fashion and markdown pricing. *Amer. Econom. Rev.* 78(5):936–953.

Pauwels K, Hanssens DM, Siddarth S (2002) The long-term effects of price promotions on category incidence, brand choice, and purchase quantity. *J. Marketing Res.* 39(4):421–439.

Popescu I, Wu Y (2007) Dynamic pricing strategies with reference effects. *Oper. Res.* 55(3):413–429.

Priceonomics (2016) Mobile app industry revenue statistics: Where's the money come from? Accessed on September 10, 2016, https://medium.com/@sm_app_intel/mobile-app-industry-revenue-statistics-wheres-the-money-come-from-/82581a45186d.

Ragaglia D, Roma P (2014) Understanding the drivers of the daily app rank: The role of revenue models. *Proc. 26th Annual Conf. Productions and Oper. Management Soc.* (Productions and Operations Management Society, Washington, DC), 8–11.

Raghubir P, Corfman K (1999) When do price promotions affect pretrial brand evaluations? *J. Marketing Res.* 36(2):211–222.

Raju JS (1992) The effect of price promotions on variability in product category sales. *Marketing Sci.* 11(3):207–220.

Rietveld J (2016) Creating value through the freemium business model: A consumer perspective. *Acad. Management Proc.* 2016(1):11073.

Roma P, Dominici G (2016) Understanding the price drivers of successful apps in the mobile app market. *Internat. J. Electr. Marketing Retailing* 7(2):159–185.

Roma P, Ragaglia D (2016) Revenue models, in-app purchase, and the app performance: Evidence from Apple's app store and Google Play. *Electr. Commerce Res. Appl.* 17(May–June):173–190.

Roma P, Zambuto F, Perrone G (2016) The role of the distribution platform in price formation of paid apps. *Decision Support Systems* 91(November):13–24.

Shankar V, Bolton RN (2004) An empirical analysis of determinants of retailer pricing strategy. *Marketing Sci.* 23(1):28–49.

Sorensen AT (2007) Bestseller lists and product variety. *J. Indust. Econom.* 55(4):715–738.

Statista (2019) Worldwide mobile app revenues in 2014 to 2023 (in billion U.S. dollars). Accessed November 15, 2019, https://www.statista.com/statistics/269025/worldwide-mobile-app-revenue-forecast/.

Su X (2010) Intertemporal pricing and consumer stockpiling. *Oper. Res.* 58(4):1133–1147.

Talluri KT, Van Ryzin GJ (2006) *The Theory and Practice of Revenue Management* (Springer Science and Business Media, New York).

Wang Q, Li B, Singh PV (2018) Copycats vs. original mobile apps: A machine learning copycat-detection method and empirical analysis. *Inform. Systems Res.* 29(2):273–291.