

# To Upgrade or Not to Upgrade? The Release of New Versions to Survive in the Hypercompetitive App Market\*

Stefano Comino  
Dip. Scienze Economiche e  
Statistiche  
Università di Udine  
Udine (Italy)  
stefano.comino@uniud.it

Fabio M. Manenti  
Dip. Scienze Economiche ed  
Aziendali "M. Fanno"  
Università di Padova  
Padova (Italy)  
fabio.manenti@unipd.it

Franco Mariuzzo  
School of Economics & Centre  
for Competition Policy  
University of East Anglia  
Norwich (UK)  
f.mariuzzo@uea.ac.uk

## ABSTRACT

Very low entry barriers and an exceptionally high degree of competition characterize the market for mobile applications. In such an environment one of the critical issues is how to attract the attention of users. Practitioners and developers are well aware that managing app updates (i.e., releasing new versions of an existing app) is critical to increase app visibility and to keep users engaged, disguising a hidden strategy to stimulate downloads. We use unbalanced panel data with characteristics for the top 1,000 apps on iTunes and Google Play stores, for five European countries, to empirically investigate publishers' strategies concerning the release of updates. We find that only in the case of iTunes updates boost downloads and are more likely to be released when the app is experiencing poor performance. We interpret this finding as evidence that the lack of quality control by Google Play leads to an excess of updating of Android apps.

## CCS Concepts

•Applied computing → Economics;

## Keywords

Mobile applications, updates, downloads, iTunes, Google Play, quality check, buzz, multihoming.

## 1. INTRODUCTION

The market for mobile applications is one of the most dynamic segments in today's economy. In June 2016, more than 5 million apps were available on the various stores, which include Apple's iTunes, Android's Google Play and Microsoft's Windows Store. The growth in the number of available apps

\*An extensive version of this paper is available as [3].

has been accompanied by an exponential increase in downloads. According to Statista.com, the cumulative number of apps downloaded from iTunes from July 2008 to June 2016 was about 130 billion. Also, in terms of the number of developers and publishers involved in the app market, figures are quite impressive. As stated by Priori (2014), in February 2014 more than 600 thousand developers published at least one app on iTunes or Google Play. Indeed, producing and distributing a mobile application for a developer is relatively easy as it requires a small amount of investment to produce computer software.

The market for mobile apps is therefore characterized on the one hand by a fast growing demand from users and on the other by low entry costs, a large number of apps and developers, and high turnover rates. For these reasons the app market has been defined as a "hypercompetitive" marketplace where developers struggle to attract adopters [4]. In this dynamic context, with several million apps available for download, one of the most challenging problems faced by developers is to catch the attention of users [1].

In order to improve app visibility, stores created the so-called "top-ranked" apps charts, listing the most popular applications. Several scholars have shown that such charts promote the matching between users and developers with top-ranked apps enjoying a remarkable boost in downloads [2, 6, 7, 5]. In turn, top ranked-charts exacerbate another feature characterizing app markets: the skewness of the distribution of downloads. According to www.appbrain.com, on Google Play about 1 million apps out of 1.4 million have less than one thousand downloads each, while by comparison just thirteen thousand apps have more than one million downloads. Being in the top positions guarantees success. Therefore, the distribution of downloads is extremely skewed to the right with only a small fraction of applications capturing most of the market. This feature is confirmed in our data as highlighted in Figure 1, which displays the distribution of downloads for the top 100 apps in Germany for iTunes and Google Play; the two diagrams show the average number of downloads for each decile of the distribution. In the case of Google Play, for instance, the average number of monthly downloads of apps of the first decile is 436,674, which is twice the average number of monthly downloads of apps in the second decile (214,594) and about eight times more than the downloads of tenth decile (58,690).

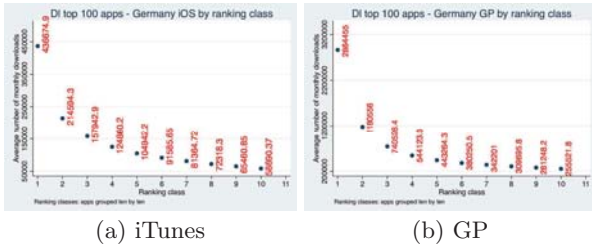


Figure 1: Distribution of downloads

To climb the top-ranked charts and to improve the visibility of their apps, publishers and developers look for any possible marketing and pricing strategy. In this article we focus on a very specific strategy that developers may exploit to attract users’ attention: the release of frequent updates.

Our data-set confirms that both on iTunes and on Google Play updates are released with an extremely high frequency: Google Play apps are updated on average every 13 days, while those on iTunes every 58 days. A more exhaustive way of presenting differences in updates by stores is by plotting the hazard function. In Figure 2 we display the estimated hazard function for updates by store, with months on the horizontal axes. We see clearly that Google Play apps are more likely to be updated than iTunes apps, i.e. each app version in Google Play tends to be of younger age. The difference is striking, as highlighted by the different domains for the two stores. Continuous updates is not a surprising feature, as it is widely recognized that by releasing updates developers stimulate user’s interest. Usually, developers promote the new versions of their products on blogs, social networks or simply in the “What’s new” section of the app store and this stimulates app visibility and, via this channel, sales.

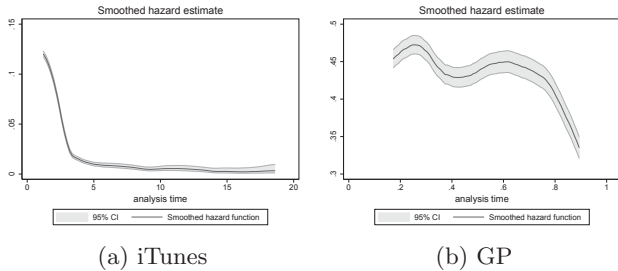


Figure 2: Smoothed hazard function estimates for iTunes and GP

We analyze the strategy of releasing frequent updates by exploiting the differences across the two main app stores, iTunes and Google Play. Interestingly, the two stores follow different policies regarding the publication of apps and updates; while Apple has adopted very strict guidelines and closely monitors the release of new apps and updates to avoid low quality apps being uploaded on iTunes, on Google Play apps can be published with a “simple click of the mouse” and

without a quality-check control. As a matter of fact Google Play apps are often criticized for their poor quality.<sup>1</sup>

In this paper, we provide intuition of a stylized theoretical model which investigates the developer’s decision about whether to update her mobile application.<sup>2</sup> We then summarize the results of the empirical analysis based on an unbalanced panel with characteristics on the top 1,000 apps on iTunes and Google Play stores for five European countries; the analysis aims at empirically testing the predictions we derive from the theoretical model.

The main result from our paper is that the difference in quality control between iTunes and Google Play have implications on developers’ decision to update their app and on their app performance.

Our paper fits into an emerging literature focussing on app updates. [11] surveyed 690 individuals (among developers and users) in order to study the specific release strategies followed by app publishers. [9] and [10] focus on updates in Google Play and Windows Phone app stores. They find that 40% of the releases in Google Play and 55% of the releases in Windows Phone have an impact on app performance, measured in terms of rating, ratings counts and download rank. Interestingly, the authors also find that the release text content matters. Updates highlighting new features (rather than bug fixes) are found to be more likely to impact positively on app performance.

The paper is organized as follows. In Section 2, we present the data and some descriptive statistics. Section 3 discusses the theoretical framework presenting a simple numerical example. The results of the empirical investigation are discussed in Section 4. Section 5 concludes.

## 2. THE DATA

In the analysis, we use monthly data on the 1,000 most downloaded apps in Germany, France, Italy, Spain, and the UK. Data was obtained from the Berlin-based consulting analytics Priori; the period of observation is September 2013-February 2014.

The dataset contains information on the following variables: name of the app, name of the publisher, app monthly and overall number of downloads by country, average customer rating of the app (scale from 1 to 5), the number of times the app was rated by users, the date of release of the app on the store, the cumulated number of versions (updates) of the app, the price of the app, whether the app has the in-app purchase option (0 – 1 variable) and, finally, whether an app is “local” (0 – 1 variable).<sup>3</sup>

All the information, but downloads, was obtained directly from the Google Play and iTunes app stores. Downloads were computed by Priori using proprietary metrics; estimating a relationship among downloads, indicators of the performance (e.g. user rating) and other publicly available information (e.g. financial statements or reputable press releases).<sup>4</sup> Only first-time installations are counted as downloads.

<sup>1</sup>www.appbrain.com estimates that nearly 15% of Google Play apps are of low quality.

<sup>2</sup>The model, as well a thorough econometric analysis, is developed in [3].

<sup>3</sup>An app is defined as local in a given country if 40% or more of its all time downloads occur in that country.

<sup>4</sup>For a sample of apps, Priori cross-checked these estimates by using real downloads data provided by partner developers.

For iTunes apps we complemented the data provided by Priori, recording information available on the App Annie web site (see [www.appannie.com](http://www.appannie.com)). This additional information includes: type of updates distinguishing between major updates (new versions of the software with significant jumps in the functionalities) and minor ones (new versions that fix bugs to avoid crashing),<sup>5</sup> the type of compatibility (Universal apps, apps for iPad only, for iPhone ...), the size of the app code as of May 2014 (data collection period), the age restrictions (4+, 9+, 12+, 17+) and whether the app was available in the language of the country or not. We also classified iTunes apps into “corporate apps” or not, where corporate apps are the apps representing an additional distribution channel (e.g. airlines or banking apps, newspaper or TV network apps etc.) or a tool aimed at spreading information about the services supplied by a company.

With only the exception of downloads (and ranking), all the available information is country invariant. For this reason we aggregate the data; hence, monthly downloads (and the growth in downloads) are the sum of the downloads in the five European countries in a given month. Summary statistics for the two stores are provided in Table 1.<sup>6</sup> It is interesting to note that:

- most of the top apps are free of charge. This is the case of nearly all Google Play apps and of more than 90% of the apps in iTunes.
- For a significant share of apps the in-app purchase option is available. This occurs for 29.7% of the cases in Google Play and for 56.1% in iTunes.
- Apps in both stores have a quite large user rating (above 4 out of 5). This observation is rather unsurprising provided that we are focussing on the most successful apps. The number of times an app has been rated is larger in Google Play (the figure is about twice as big as in iTunes).
- On average, apps in iTunes are older than in Google Play: 19 months compared to 15.
- Apps are updated very frequently in both stores. On average in Google Play apps are updated about 33 times since their publication, while this figure reduces to about 10 in iTunes.
- Downloads are much higher in Google Play than in iTunes, with Google Play apps on average downloaded nearly five times more than iTunes apps (249,803 compared with 58,156).
- Roughly 35% of apps generate at least 40% of their revenues in one country, i.e. they are local (similar figure in both stores).

<sup>5</sup>We were able to collect this information by looking at the three-digit code of the various versions. Programmers assign a unique three-digit number to each new version of the software. So, for example, version 2.12.4 means that there have been two major updates and sixteen minor ones (12+4).

<sup>6</sup>Since in several cases, the same app appears in the top 1,000 in more than one country in a given month, from the original 30,000 observations per store (1,000 apps, during 6 months, in 5 countries) we are left with 15,983 observations for Google Play, and 14,755 for iTunes.

- On average, in Google Play, a single developer distributes about 3 top-ranked apps. The figure for iTunes is 3.5.
- On average, an app enters the top-1,000 ranking in about 2 out of 5 countries in both stores (1.876 in Google Play and 2.030 in iTunes).
- In iTunes, about 2 out of 10 updates are major updates.

Our data confirms a couple of features that have already been found in the literature [1]: *i*) downloads exhibit an extremely skewed distribution, with top apps accounting for a large fraction of total downloads, and *ii*) large turnover/churn, with few applications which succeed in staying in the top 1,000 list in all the six months of observation. Regarding feature *i*), we have already shown Figure 1 the distribution of downloads for the top 100 apps in Germany for iTunes and Google Play; the distribution of downloads in the other countries of the sample is very similar.

Where feature *ii*) is concerned, Table 2 displays the high level of turnover that characterizes iTunes. The overall number of different apps that we observe during the six-month period in the five countries is 10,986;<sup>7</sup> 18.24% of these apps are observed every month (indicated in the table with “All months”). However, a substantial percentage (about 44%) of apps appear only in one month. The level of turnover is even larger in Google Play (table not reported). For the Android store, the overall number of applications that we observe increases to 13,034 and the share of apps that appears only one month is about 50%.

Another way of looking at turnover is through the volatility of app performance. Figure 3 focuses on apps that appear only once in our six-month period of observation; more specifically, the figure focusses on iTunes apps in Germany,<sup>8</sup> and shows the ranking of apps (grouped into groups of twenty) that entered the top 1,000 for only one month (and were not listed before or after that month).<sup>9</sup> We observe that the distribution is highly concentrated on high values of the ranking especially in position between 800 and 1,000). This evidence reinforces the observation app performance can be highly volatile and characterized by huge jumps in downloads.

### 3. A SIMPLE THEORETICAL MODEL

In [3] we develop a very stylized theoretical framework aimed at providing the main intuition for our econometric analysis. The theoretical model incorporates three important features characterizing the app economy. First, given the huge number of available apps, performance (both in terms of downloads and usage) depends not only on the technical quality of the software code and on its ability to meet user needs but also on the ability of publishers to make their apps visible, to stimulate “buzz” surrounding their apps. The second feature of the model is based on the evidence

<sup>7</sup>The minimum number of apps we could have observed in our sample is 1,000 (if the top 1,000 apps are the same in the five countries during the whole period) while the maximum is 30,000 (if the top 1,000 apps change every month and are different in the five countries under observation).

<sup>8</sup>We obtained similar diagrams for the apps of the other countries and the patterns were similar.

<sup>9</sup>Figure 3 was obtained by dropping the initial and last month of observation in order to spot apps that truly entered the top 1,000 ranking only once.

**Table 1: Summary statistics from the five countries**

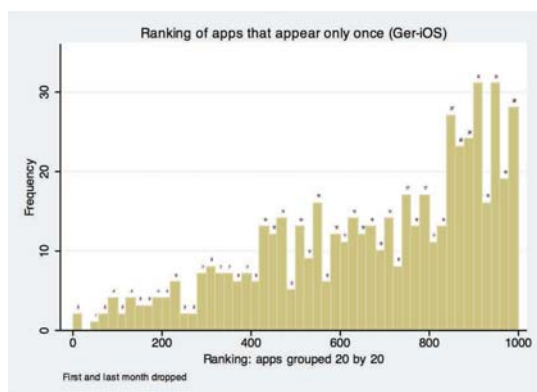
	Google Play			iTunes		
	N.	mean	std. dev.	N.	mean	std. dev.
Free	15,983	0.999	0.033	14,755	0.917	0.276
Price (if free=0)	17	2.447	1.148	1,229	2.759	2.907
In-app	15,983	0.297	0.457	14,755	0.561	0.496
Local	15,983	0.374	0.484	14,755	0.349	0.477
Age (in months)	15,983	15.036	13.993	14,755	19.341	15.991
Number of updates (all time)	15,983	33.430	78.487	14,755	10.079	9.990
Update (sampling period)	7,991*	0.542	0.498	8,531	0.453	0.498
# apps same developer	15,983	2.983	4.583	14,755	3.586	5.163
Number countries	15,983	1.876	1.395	14,755	2.030	1.491
Monthly downloads	15,983	249,803	1,244,139	14,755	58,156	166,472
Growth downloads**	8,225*	-0.026	0.763	8,591	-0.160	0.646
				App Annie data		
Age major version (in months)				13,723	10.536	9.924
Number major versions				13,852	2.055	2.004
Update major version				8,175	0.037	0.189
Size				13,851	63.934	149.595

\*Updates and growth are in first differences and hence have less observations.

\*\*Growth rates are expressed as log changes.

**Table 2: Pattern of apps, iTunes, all periods and countries**

Freq.	%	Cum.	Pattern
2,004	18.24	18.24	All months
1,098	9.99	28.24	Sept only
1,025	9.33	37.57	Feb only
762	6.94	44.50	Oct only
755	6.87	51.37	Nov only
637	5.80	57.17	Jan only
592	5.39	62.56	Dec only
423	3.85	66.41	Sept & Oct
354	3.22	69.63	Dec, Jan & Feb
3,336	30.37	100.00	(other patterns)
10,986	100.00		N. of different apps



**Figure 3: Distribution of downloads**

that, as discussed, performance is highly skewed with a limited number of apps able to capture most of the returns. Finally, in the model we assume that the release of an update

stimulates the “buzz” surrounding the app: bloggers, social networks and users are more likely to be attracted by an app, and therefore to comment on, when a new version is available for download. Note, however, that this increase in buzz does not necessarily translate into positive comments; as a matter of fact, users and experts might not welcome the changes made in the software and this may induce them to criticize the update, thus generating what we define as negative buzz. Formally, we assume that the release of the new version of the software increases the buzz surrounding the app and, at the same time, it makes more uncertain how the app is perceived by users.

We use the theoretical model to derive a series of testable predictions; the following numerical example provides the intuition of the model. App performance depends on the perceived quality of the app, which, in turn, is affected by the intrinsic/technical quality of the software code as well as by the buzz surrounding it. App returns are highly skewed: they can be very low or very large depending on the ability of the developer to hit certain thresholds in app perceived quality. Formally, app returns are 0 if the app perceived quality is below the threshold of 75, while they equal 1,000 in the case the perceived quality is larger than 75. The perceived quality of the app is not deterministic and it is affected by the buzz. Let us assume that the current version of the software turns out to have a high perceived quality equal to 100 with probability 1/2 and a low perceived quality equal to 60 with complementary probability. If the developer decides to release a new version of the software the buzz surrounding the app is larger and the perceived quality more uncertain. Specifically, let us assume that the perceived quality of the update is:

$$\begin{cases} 120 \text{ with probability } 1/4 \\ 80 \text{ with probability } 1/2 \\ 0 \text{ with probability } 1/4 \end{cases}$$

Comparing the two versions of the software, it turns out that the current one has a higher expected perceived quality, 80 compared to 70. However, only the high realizations of



the perceived quality matter provided that low realizations yield no returns. Hence, the developer prefers to release an update given that the 1,000 in returns will materialize with probability 3/4 (the probability that the perceived quality is, at least, 75) while they materialize with probability 1/2 without the update. In other words, the developer finds it profitable to update their app even though the new version of the software has a smaller expected perceived quality.

The model suggests that, given the peculiar features of the app market, publishers might have incentives to release new versions of their app even if they offer little/no improvement to the technical/intrinsic quality of the software. The release of a low quality update is more likely to take place when the current version of the software is expected to have a worsening in its performance. In this case, the publisher knows that with a large probability low returns will be generated and, therefore, she/he has all the incentive to update the app in an attempt to revive it. This “bet for resurrection” strategy” is more likely to be feasible for developers operating in Google Play (compared to those operating in iTunes) where there is no quality control and where, therefore, also low-quality updates can be released.

#### 4. EMPIRICAL MODEL AND RESULTS

The main message of the model is that the differences in quality control between iTunes and Google Play may have important consequences on developers’ decisions to update their apps and on the effects of such decisions on performance. We use our data to test the predictions of the theoretical analysis; in our empirical analysis, we approximate app performance with downloads.

Figure 4 provides a preliminary picture of the evidence of the different role of updates in stimulating downloads in Google Play and in iTunes. The figure shows the kernel density of the growth in downloads in the two stores distinguishing between updated and non-updated apps.<sup>10</sup> Interestingly, in Google Play the two density functions nearly perfectly overlap; this suggests that updated and non-updated apps perform very similarly in terms of downloads. On the contrary, in iTunes the density function of non-updated apps is more asymmetric to the left and concentrated on negative values of the growth rate of downloads. Hence, updates seem to improve app performance in iTunes, while they have no substantial impact in Google Play.

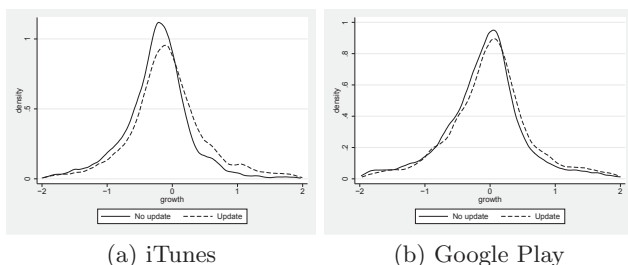


Figure 4: Download growth density function

<sup>10</sup>Growth rates are expressed in logarithms and refer to the overall number of downloads each app obtains in the five countries.

In order to better understand the role of updates, for each store separately, in [3] we study the dynamics of the rate of growth of downloads ( $g$ ) and of the decision to release an update ( $u$ ). Our econometric specification is, therefore, made of two equations: the growth equation, where we study the determinants of download growth, and the update equation, where we study the determinants of the release of updates. Specifically, the equations are modeled as a linear autoregressive processes of lag 1 and in the estimations we take the first differences in order to eliminate app unobserved heterogeneity, which may be correlated with the lagged dependent variable and the contemporaneous outcome of the app update. Thus in our estimates we regress the following econometric simultaneous equations:

$$\begin{aligned} \Delta g_{jt} &= \phi_{11} \Delta g_{j,t-1} + \phi_{12} \Delta u_{jt} + \tau_{1t} + \Delta \mathbf{x}_{1jt} \beta_1 + \Delta \varepsilon_{1jt} \\ \Delta u_{jt} &= \phi_{21} \Delta g_{j,t-1} + \phi_{22} \Delta u_{j,t-1} + \tau_{2t} + \Delta \mathbf{x}_{2jt} \beta_2 + \Delta \varepsilon_{2jt}, \end{aligned} \quad (1)$$

where  $\tau$  indicates a function of time; the vector  $\mathbf{x}$  includes a set of control variables, and the last term,  $\varepsilon$ , is the idiosyncratic error term. We tackle the endogeneity of  $\Delta g_{j,t-1}$ ,  $\Delta u_{jt}$  and  $\Delta u_{j,t-1}$ , by selecting a set of instruments that exploit the time series dimension of the panel and the fact that a sizeable number of developers distribute several apps in the top 1,000 ranking. The selected instruments are tested for their relevance and validity.

As regards the growth equation, we find that the release of an update positively affects downloads in iTunes while it has no significant impact in Google Play.<sup>11</sup> This result confirms the preliminary evidence shown in Figure 4. We interpret this finding in terms of the institutional differences characterizing the two stores. As argued, only in iTunes there is a strict quality check for apps and updates. Therefore, the absence of a significant impact of updates in Google Play might be due to the fact that in this store both high and low-quality updates get published so that the overall effect on downloads is not significant.

We also find that, similar to the evidence shown in [8], the growth rate of downloads increases with the number of other applications by the same developer. We refer to this as “portfolio effects”, which we interpret as a signal of developers’ ability to exploit the presence of complementarities among apps.

As regards the update equation, our theoretical analysis suggests that an important determinant of the developers’ decision to release a new version of an app is its past performance: an update is more likely to be released when the developer observes a worsening of the performance of the current app. In the estimations we measure the past performance of an app in terms of lagged growth rate of downloads. Our data confirm this prediction only for iTunes apps; for Google Play, the past performance does not have an impact on the decision to release an update. Again, we interpret this difference between the two stores on the basis of the difference in quality control between iTunes and Google Play. Since iTunes apps must pass a quite severe quality control, developers are less free to publish updates: developers who have written an update may therefore decide to delay publication until it is really necessary, i.e. when

<sup>11</sup>A detailed presentation of these results as well as additional robustness checks are in [3].

the performance of the app worsens it triggers a response to counter the drop in downloads. In contrast, on Google Play, developers can publish apps any time they want with a simple click of the mouse. In this environment, updates are continuously published, thus diluting the impact of past performance on the decision to release a new version of the app.

For iTunes apps, we have been able to distinguish between major (i.e. significant changes in functionalities) and minor updates (i.e. bug fixing and minor changes to the software code). Therefore, we run separate regressions of both the growth and the update equations distinguishing between major and minor updates. We find that both types of updates positively impact the growth rate of downloads; at the same time, we find that the worsening of app performance increases the likelihood of releasing minor updates only. This finding suggests that minor updates are better suited to be used as a strategic tool to counter poor past performances. The results of these estimates are available in the full version of the paper [3]

## 5. CONCLUSION

App developers release new versions of their mobile applications with an extremely high frequency. In this paper, we focus on app updates and present a stylised model describing the determinants of the decision to release a new version of the software; our analysis suggests that updates might be published strategically in order to revive apps and stimulate the buzz surrounding them. An interesting insight of our theoretical framework is that when an app experiences very poor performance, a developer might be tempted to release even a low quality update, in the hope of reversing the trend.

The discussion of an empirical analysis undertaken in [3] has shown that updates play a quite different role in stimulating downloads on iTunes than do on Google Play: while in the former updates boost downloads, their effect is not significant in the latter. These results are consistent with our conjecture, that the lack of a quality check on Google Play can cause a phenomenon of too frequent updating: developers release both high and low quality updates, which, on average, do not impact downloads.

## 6. REFERENCES

- [1] T. Bresnahan, J. Davis, and P.-L. Yin. *Economic value creation in mobile applications*. 2015. in A. Jaffe and B. Jones eds: *The Changing Frontier: Rethinking Science and Innovation Policy*, Chicago Scholarship Online.
- [2] O. Carare. The impact of bestseller rank on demand: Evidence from the app market. *International Economic Review*, 53(3):717–742, 2012.
- [3] S. Comino, F. M. Manenti, and F. Mariuzzo. Updates management in mobile applications. iTunes vs Google Play. CCP Working Paper 15-4 v3, 2015.
- [4] D. Datta and K. Sangaralingam. Do app launch times impact their subsequent commercial success? *IEEE 2013 International Conference on Cloud Computing and Big Data*, Dec 2013:205–210, 2013.
- [5] R. Garg and R. Telang. Inferring app demand from publicly available data. *MIS Quarterly*, 37(4):1253–1264, 2013.
- [6] A. Ghose and S. P. Han. Estimating demand for mobile applications in the new economy. *Management Science*, 60(6):1470–1488, 2014.
- [7] B. Ifrach and R. Johari. The impact of visibility on demand in the market for mobile apps. Available at SSRN: <http://ssrn.com/abstract=2444542>, 2014.
- [8] G. Lee and T. S. Raghuram. Determinants of mobile apps success: Evidence from app store market. *Journal of Management Information Systems*, 31(2):133–170, 2014.
- [9] W. Martin, F. Sarro, and M. Harman. Causal impact analysis applied to app releases in Google Play and Windows Phone Store. *RN*, 15:07, 2015.
- [10] W. Martin, F. Sarro, and M. Harman. Causal impact analysis for app releases in Google Play. In *Proceedings of the 2016 24th ACM SIGSOFT International Symposium on the Foundations of Software Engineering*, FSE '16, 2016. To appear.
- [11] M. Nayebi, B. Adams, and G. Ruhe. Release practices for mobile apps—what do users and developers think? In *2016 IEEE 23rd International Conference on Software Analysis, Evolution, and Reengineering (SANER)*, volume 1, pages 552–562. IEEE, 2016.