The Effect of Consumer Search Costs on Entry and Quality in the Mobile App Market

Daniel Ershov*
University of Toronto
daniel.ershov@mail.utoronto.ca

December 3, 2016

JOB MARKET PAPER
Click here for current version

Abstract

This paper examines the effects of consumer search costs on entry, product design, and quality in online markets. Using 2012-2014 data from the Google Play mobile app store, I take advantage of a natural experiment that reduced search costs for one product type (game apps) in early 2014. Difference-in-differences estimates show that entry increased by 33% relative to the control group (non-games), and that most additional entry was by "niche" products. These estimates also show that lower search costs reduced the quality of new entrants. To separate out the different welfare effects of this change, I develop and estimate a structural model of demand and supply. I show that there are large welfare gains from reduced marginal search costs, smaller gains from increased product variety, and very small losses from lower product quality.

*I would like to thank Victor Aguirregabiria, Avi Goldfarb, and Heski Bar-Isaac for their time spent reading and discussing this paper and for their helpful comments. This paper also substantially benefitted from discussions with Joel Waldfogel, Eduardo Souza Rodrigues, and seminar participants at the University of Toronto, and the 2016 EARIE and JEI conferences. All errors are my own.

1 Introduction

In many online markets, consumers can search through thousands, or hundreds of thousands, of products. This search is costly.[1] "Discoverability" in these markets is important, as products are hard to find.[2] As a result, consumer search costs should affect firm entry and quality incentives in online markets. Firms could choose to not enter markets with high search costs. Potential entrants may also underinvest in quality, conditional on entry. Anti-trust cases brought against Google provide an example of the importance of consumer search in online markets.[3] These cases are concerned with Google's potential foreclosure practices.[4] By placing its own products ahead of competitors' products in search results, Google increases the costs for consumers of finding competitors' products, and may deter entry.[5]

This paper examines how consumer search costs in online markets affect market structure, product variety, quality, and consumer welfare. I empirically study these effects using 2012-2014 data from the Google Play mobile app store, a large online market. App stores illustrate the main characteristics that distinguish online markets.[6] In particular, app stores have a large number of products. Thousands of new apps appear every week, and it is costly for consumers to search for new products. According to industry surveys (Nielsen 2011, Ipsos 2013), consumers' primary search method in these stores is to browse through category areas (e.g.,"Productivity Apps"). These category areas include "bestseller" lists, as well as other "featured" apps (see Section 2.1 for more discussion).

I take advantage of a natural experiment - a change in the structure of the Google Play app store. In December 2013, Google Play announced a split of mobile game categories, which grew from 6 to 18.[7] The split took place on March 2014. Google's split game categories matched iTunes' existing game categories, suggesting that category selection was not driven by pre-existing trends. The alternative view, that something particular to competition or entry in games caused Google to change them, does not seem to hold in the data.[8]

Industry observers believe that the split in the categories reduced consumer search

[1] e.g., Ghose, Goldfarb, and Han (2012), Athey and Imbens (2015)
[2] Sorensen (2007), Hendricks and Sorensen (2009)
[3] http://www.wsj.com/articles/inside-the-u-s-antitrust-probe-of-google-1426793274
[4] Chiou (2015), Crawford et al (2015)
[5] http://www.businessinsider.com/evidence-that-google-search-results-are-biased-2014-10
[6] e.g., Ellison and Ellison (2005), Levin (2011).
[7] See Appendix A for a full list of categories before and after the split.
[8] See Section 4 for more evidence.

costs and helped consumers find more relevant products.[9] Before the change, consumers browsing through the categories would see different app types together (e.g., Family Games and Action Games). Consequently, consumers looking for a particular app type would not necessarily find it easily.[10]

The data used consists of weekly and monthly snapshots of the US Google Play store from January 2012 to December 2014. This data includes the characteristics of all apps available in the market. Since non-game categories were not split, I use difference in differences (DID). DID estimates capture three key effects. First, entry increases in games relative to non-games by 33%. Second, most of the entry effects are driven by "niche" app types that were more difficult to find before the category split. Lastly, the quality of new games - as measured both by consumer ratings and app size in MB[11] - fell after the split.

The overall impact of these effects on consumer welfare is potentially ambiguous. If consumers like variety, then additional entry should be welfare increasing. However, consumers should also like quality. Conditional on the number of products, a greater share of low quality products would reduce consumer welfare. Moreover, in the presence of search costs, a larger number of low quality products could make it harder to find high quality products. This could also offset the welfare increase.

To measure the welfare effects of this change, I set up a structural model of demand and supply. On the demand side, following Moraga Gonzalez, Sandor and Wildenbeest (2015), I propose and estimate a model that merges a logit model of differentiated product choice with a search model. This demand model estimates consumer utility parameters and consumer search costs. I estimate two specifications - a static specification, and a dynamic specification where apps' past downloads affect their current market share.

To capture the effect of the category split, I allow the search cost parameters to be different before and after the split. I also allow the other demand parameters to vary between the pre-split and post-split periods, and I do not find any significant difference. The results show that search costs fell by as much as 50%. The demand estimates also suggest that a 1% increase in search costs reduces consumer utility by approximately 3 cents - or 2% of a paid app's average price.

The supply side of the model is a discrete choice game of incomplete information

[9] http://androidcommunity.com/google-play-store-sports-18-new-categories-for-games-20140317/

[10] Increasing the number of categories should not *always* reduce search costs. At some point, having too many categories can result in a more difficult search process. There is probably an optimal number (or range) of categories to display in this market, but this paper cannot say what this number (or range) may be.

[11] An indicator of the number of features in the app, and correlated with the other quality measure.

(Seim 2006, Augereau, Greenstein and Rysman 2006) where firms (apps) decide in which category to enter (if any) and the quality level of the application. It is an oligopoly model of market entry and choice of product characteristics (category and quality). In the specification of variable profit, I take into account that most of the apps are free to download, and firms make money from a combination of in-app purchases and in-app advertising. Therefore, I consider that the variable profit is a nonlinear (quadratic) function of the number of downloads. The specification of entry costs take into account that these costs depend on product quality in a nonlinear form. The structural estimation of this game identifies entry cost parameters and the parameters that capture how app downloads translate into profits.

After estimating the demand and supply models, I measure the change in consumer welfare due to the category split. I also decompose the total welfare effect into the changes due to greater product variety, product quality, and falling marginal search costs. The counterfactual simulations show that welfare increased by 60% after the split in the categories. A decomposition shows that most of the increase in consumer surplus comes from the reduction in marginal search costs. There is also an increase in consumer surplus due to greater product variety. This effect is larger than the fall in consumer surplus due to lower quality.

Reports suggest that the mobile app economy created over a million jobs in the US.[12] If search frictions reduce incentives to invest in app quality, it may be in the public interest to incentivize developers to create more high quality apps. I simulate two additional counterfactuals that reduce the entry costs in the market by 15% and 30%. This would correspond, for example, to a subsidy to app start-ups. The counterfactual simulations show that these policies encourage more entry by higher quality apps relative to the baseline, but their overall welfare effects are smaller compared to the effect of reducing search costs.

This paper is part of a long literature on the effects of consumer search costs on market outcomes - starting with Stigler (1961), Nelson (1970), and Diamond (1971).[13] There are a number of recent theory papers that examine the interaction between search costs, entry incentives, investment in quality, and product design.[14] The existing empirical literature on these topics is more sparse.[15] Data suggests

[12]http://www.asymco.com/2015/01/22/bigger-than-hollywood/

[13]See Stahl (1989), Anderson and Renault (1999), Armstrong, Vickers, and Zhou (2009), Chen and He (2011), and Zhou (2014), for more recent theory papers. See Syverson and Hortacsu (2004), Wildenbeest (2011), Chandra and Tappata (2011), De los Santos et al (2012), Koulayev (2014), and Bronnenberg et al (2016) for recent empirics.

[14]e.g., Bar-Isaac, Caruana, and Cunat (2012), Yang (2013), Cachon, Terwiesch, and Xu (2008), Larson (2013).

[15]Goldmanis, Hortacsu, Syverson and Emre (2010), Waldfogel (2011), Brynjolfsson, Hu, and

that the introduction of the internet in the 1990s reduced consumer search costs for books, movies, and music. At the same time, the number of these products available to consumers increased. The implication is that lower search costs increased entry. However, this evidence is primarily descriptive. This evidence also does not clearly distinguish changing search costs from other effects that could increase entry.[16]

This paper has three key contributions to the existing literature. First, no other paper, to my knowledge, uses an exogenous policy change to identify the effects of consumer search costs on firm entry. As stated above, past literature relies on descriptive evidence. Nonetheless, my results are consistent with past theoretical and empirical findings (e.g., Cachon, Terwiesch, and Xu 2008).

Second, this paper presents evidence on questions that have not yet been examined empirically. I show that lower search costs have an effect on product design (consistent with Bar-Isaac, Caruana and Cunat 2012, Yang 2013), as well as on product quality. The effects on quality are ambiguous in theory (Fishman and Levy 2015): on the one hand, when search costs fall in a market with vertical and horizontal differentiation, consumers can find the highest quality products, which should provide incentives to improve quality. On the other hand, if a firm invests in high quality, consumers who find their product can also find competing high quality products that are better horizontal matches. Thus, lower search costs can drive quality down. My results show that average quality falls, suggesting that the second effect dominates in this application.

Lastly, the structural model allows me measure the importance of the different welfare effects. I show that consumer welfare losses due to changes in app quality are smaller than welfare gains from greater product variety. These results suggest that with *higher* search costs, the relative high quality of the firms that do enter does not fully offset the negative welfare effects of foreclosure. As well, higher marginal search costs would also reduce welfare. This suggests that policies which raise search costs in online markets can have strong negative effects on consumer welfare.

The paper proceeds as follows: Section 2 provides an overview of the mobile app market. Section 3 describes the data and presents some summary statistics. The fourth section presents the reduced form results. The fifth section presents the specification and estimation of the structural model, and the counterfactuals. The final section concludes.

Simester (2011), Zentner, Smith, and Kaya (2013), and Aguiar and Waldfogel (2016).

[16] e.g., the costs of producing movies, books, and music fell in the same time period.

2 App Market Background

2.1 Users

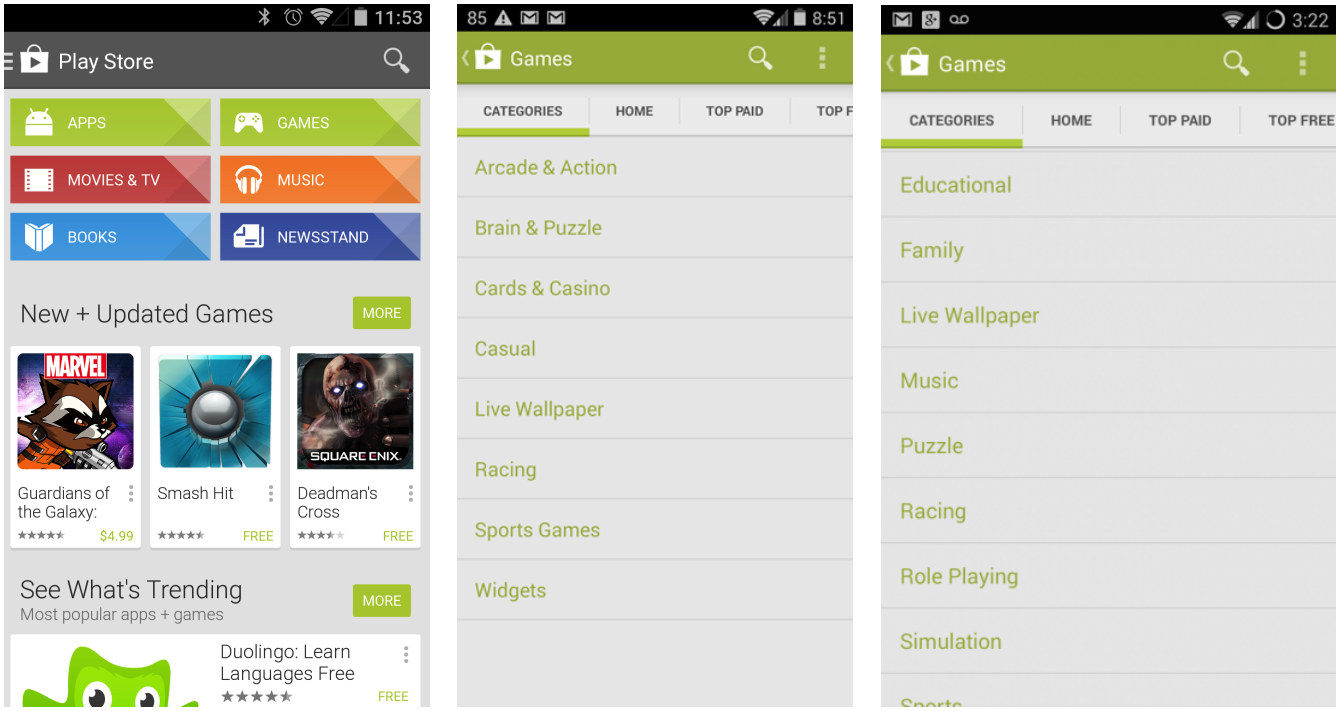
The Google Play store comes pre-installed on any phone that runs the Android OS. The first screen of the Google Play store appears below in the left panel of Figure 1. When a user open the Google Play store on their phone, they see a number of tabs. They can choose to look at Games, or Apps (non-game apps), or they can choose to look at individual "featured" games/apps/music/movies/books types that appear directly on the first page (this would be the "New + Updated Games" in the Figure). Once they choose a product type, Games for example, they get another series of panels with popular games. Alternatively, users can choose to look for more specific product types by choosing a "category." The middle panel of Figure 1 shows the choice of game categories in late 2013, and the right panel of Figure 1 shows the choice of game categories in mid 2014. I am exploiting the change in this feature as the natural policy experiment. Rather than 6 categories (plus the "widget" and "live-wallpaper" category), Google Play split their game categories into 18 different types in March 2014.[17]

Once users choose a category, they can either look at a panel of "featured" products from that category, or they can look at top-lists, which display the apps with the largest number of downloads in approximately the past week.[18] The top lists are Top Paid, Top Free, Top Grossing, and Top New Paid and Top New Free, arranged in that order horizontally. The left panel of Figure 2 below shows the top list for all free Apps. At that point, users only observe apps' names, their icons, their position in the list, their average user ratings (the average "star" rating of the app), and their price. They observe the same information about featured apps. Once they click on a particular app listing they get to observe much more (see the right panel of Figure 2). In particular, users get to observe a number of screenshots from the app, the average rating of the app, how many people have downloaded this app, the size of the app in MB, and a textual description of the app. It is at that point that they choose whether they want to download (or purchase) the app or not.

[17]For a full list of categories before and after the split, refer to Appendix A.

[18]The exact algorithm that determines the position of an app in the top lists is not necessarily known, but it is widely believed to be related to downloads. That said, it may also be somewhat related to other measures of consumer usage and retention.

Figure 1



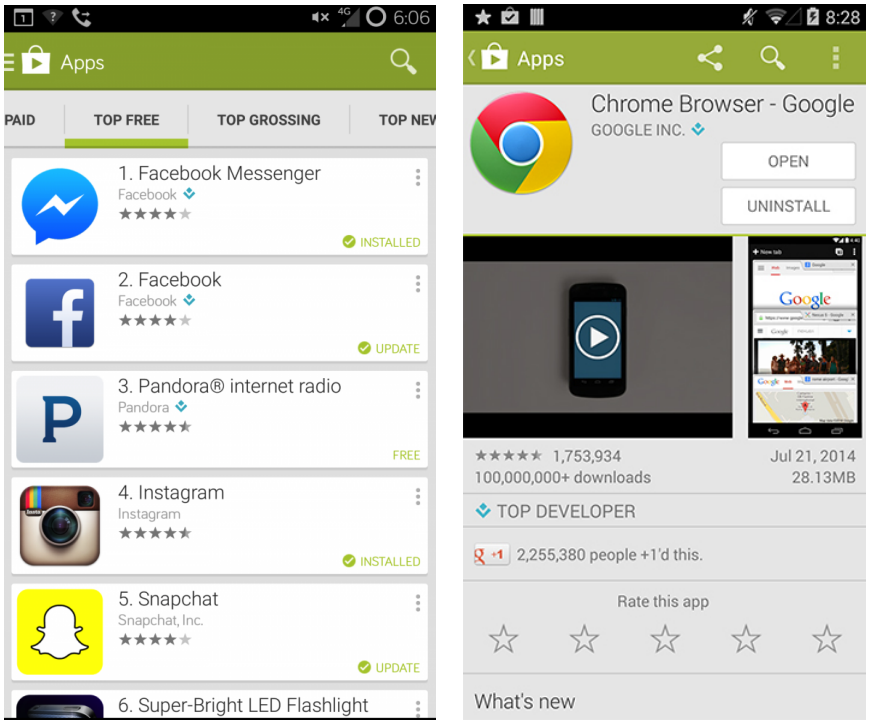
2.2 Developers

The costs of entering an app into the Google Play market are zero - a developer has to pay a one time fee of $25 to register with Google Play and then can publish apps for free.[19] On the other hand, developing an app can be costly. At the low end, the cost of programming an app (in computer equipment, software, and wages for programmers, designers, and debuggers) can be as low as a few thousand dollars. Companies that develop those low cost apps generally consist of a few individuals who work together for a few weeks. On the higher end, apps that link up to databases (e.g., calendar applications, notes applications, texting applications), can cost up to and above 50,000 dollars.[20] At the very high end, apps that require more inputs can cost as much as hundreds of thousands or millions of dollars (e.g., mobile games

[19] http://www.techrepublic.com/blog/software-engineer/app-store-fees-percentages-and-payouts-what-developers-need-to-know/

[20] https://www.applicoinc.com/blog/much-cost-build-app/

Figure 2



with advanced 3D graphics, or a social media application with video chatting).[21] Instagram, for example, spent over $500,000 in venture capital money on various technologies designed to accommodate a fast-expanding photography-based social network.[22]

When developers introduce a new app into the Android market, their key choice is the category into which they introduce their app into, since it matters for consumers' search.[23] On Android (unlike the Apple App store), the category choice is mutually exclusive and you can only choose one category at a time.

Developers make money from their apps using any of three ways. First, they can charge an upfront price for downloading their app. These are the "paid apps" and in Google Play they constitute a minority, about 20%, of the total number of apps. The prices of the apps on Google Play are relatively low. Median paid app price

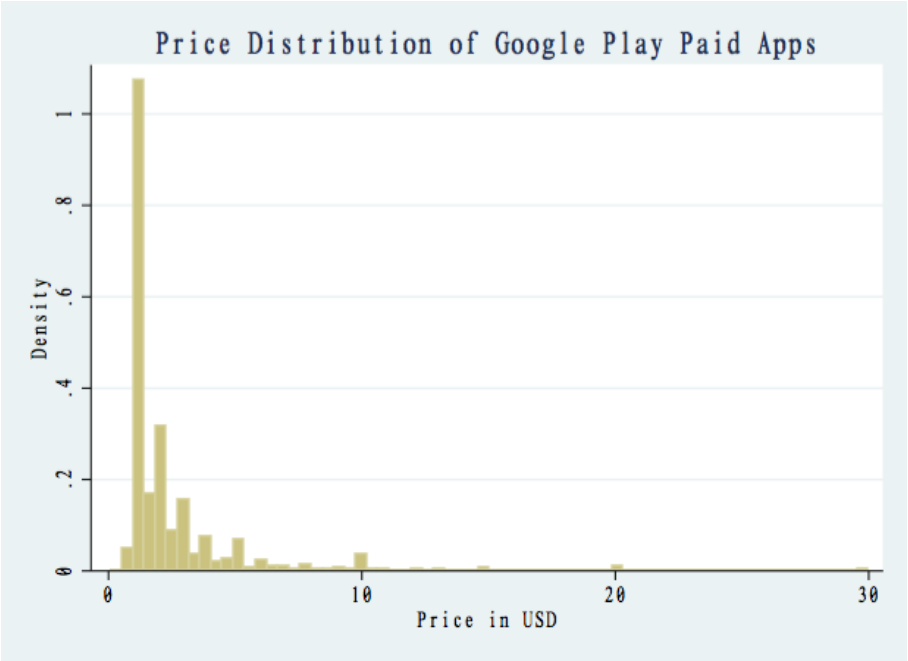
[21] http://savvyapps.com/blog/how-much-does-app-cost-massive-review-pricing-budget-considerations

[22] http://instagram-engineering.tumblr.com/post/13649370142/what-powers-instagram-hundreds-of-instances

[23] See previous subsection for more detail on the way consumers search the app market.

is $1.5, the price of an app in the 90th percentile is $5, and the price of an app in the 99th percentile of the price distribution is $29.99 (see Figure 3 below for a more detailed price distribution).

Figure 3

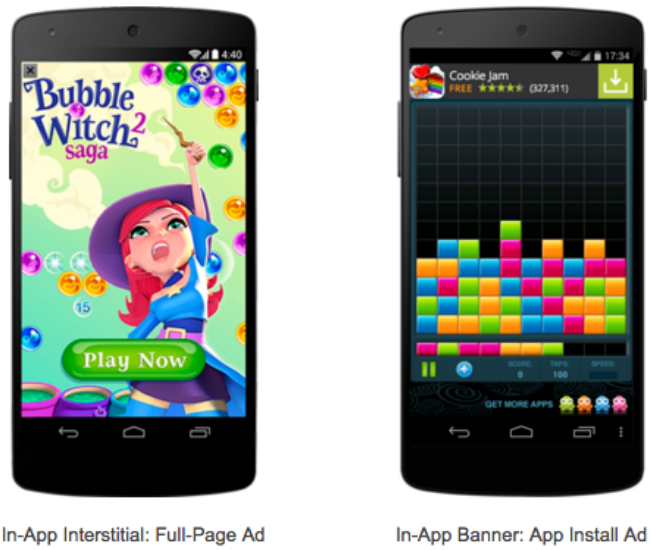


Second, the developers can allow users to download their apps for free, but place advertisements in the app. These can consist of something like full-page pop-up ads (see left panel of Figure 4), as banner ads (see right panel of Figure 4), or as video ads. The revenues from in-app advertising are similar to the revenues from search advertising - both depend on the number of users, the type of the ads, the frequency with which ads are shown, and other factors. In particular, there is another side to the market (which I will not model) where advertisers bid in a second price style auction to display their advertisements in different types of apps. This auction system seems to be similar to the Google search auction system, which suggests that the revenues for apps highly depend on their downloads and their type. Varian (2007) shows that bids in Google search auctions are increasing in rank and are convex (sometimes exponential), meaning that the bids for top slots are much higher than bids for lower slots. According to anecdotal evidence, it seems that the average price paid for 1,000 shown ads ("impressions") in the US is approximately $2-3.[24]

[24]https://www.quora.com/If-you-have-made-a-free-Android-app-with-10-000+-downloads-how-

It is likely that there is substantial variation across apps - with top apps receiving substantially higher rates, as they are more attractive to advertisers, whereas apps which are lower ranked faring much worse.

Figure 4



Source: https://www.google.com/admob/

Lastly, the developers can make money by allowing users to download their apps for free, but sell them features or products within the app (so-called "in-app purchases"). Those could include many things, including additional levels for games, subscriptions for magazines (or particular magazine columns), and so on. Those could be immensely profitable for the top apps - indeed, many top games such as Clash of Clans are free but can make hundreds of millions of dollars from a combination of in-app purchases and advertising.

Google (as the platform provider) also makes money from the app market. First, it obtains $25 for every new developer who enters the market. Second, Google takes a 30% cut of paid app downloads. It also takes 30% of the in-app-purchases made on free apps.[25] Of the remaining free apps who use app advertising to make money, about 50% are using Google's advertising platform, Admob, meaning that Google

much-ad-revenue-are-you-making

[25]https://support.google.com/googleplay/android-developer/answer/112622?hl=en

gets a cut from the advertising revenues.[26] Lastly, even if Google does not earn direct money from other advertising platforms, they do get data from all apps that helps them optimize their own search results and advertising.

3 Data

3.1 Data Description

The data in this paper comes from AppMonsta.com and consists of daily snapshots of all apps on the Google Play store, aggregated at the weekly or monthly level, starting from January, 2012, up to December, 2014.[27] This is the first paper to use this particular version of the dataset. Liu, Nekipelov, and Park (2014) use a similar dataset from the same source, but their time period is 2011-2012. The dataset includes app price (in USD), a histogram of the ratings the app received (ranging from 1 to 5), app size (in MB), the number of screenshots the app shows, the number of video previews the app shows, and a download range for the app (referring to the total number of downloads the app received over its lifetime). Additionally, I know the app's category, the name of the app's developer, and a textual description of the app, explaining what it "does." Lastly, I observe the "top lists" for every category, which are the top 500 best-selling free and paid apps in each category for each week.

3.2 Data Management - Predicting App Downloads

Most of the reduced form results in Section 3 (e.g., the effects of the policy change on entry, quality, and prices) do not rely on measuring app downloads. However, information on per-month app downloads is necessary for the estimation of the structural demand model in Section 4.

The raw data includes a range of downloads that an app has accrued over its lifetime - for example [1-5] downloads, or [1,000-5,000] downloads, or [1 million - 5 million] downloads.[28] Since this range is observable monthly or weekly, it is conceptually straight forward to define "per-period downloads" as the difference in lifetime downloads - for example, it can be the difference in the lower bounds of lifetime downloads, or in the average of the lifetime downloads.

[26] http://www.appbrain.com/stats/libraries/ad

[27] Weekly aggregation is used to predict the downloads of each app (see following section), and the monthly aggregation for the rest of the analysis.

[28] See full list of ranges in Appendix B.

A key issue with this measure is that the size of the range increases with the number of downloads. The ranges start at 4 downloads ([1-5], [5-10]), but then increase to 40 ([10-50]), and eventually to 400 million ([100 million - 500 million]). This introduces two sources of measurement error, which become worse for more successful apps. (1) this measure will likely overstate the per period downloads for successful apps that move from one bandwidth to another. For example, if a firm has a bandwidth of 100 thousand to 500 thousand downloads, and then they move up to 500 thousand to 1 million downloads in the next period, it could mean that they sold 500 thousand units this period, or 3 units. (2) this measure will understate the per-month downloads for successful apps that do not move bandwidths, since moving bandwidths is harder the more successful you are. For example, an app in the [100 million - 500 million] download bandwidth can have millions of units sold every month while still remaining in the same bandwidth.

To recover the weekly or monthly downloads of apps in the Google Play market, I rely on two features of the data. First, the bandwidth of lifetime downloads for new entrants is equal to their per-week downloads (in the previous week, they had zero downloads). Second, I know that the weekly category rankings reflect the 500 most-downloaded apps in each category roughly over the past week.[29] As a result, if I look at the market at a weekly frequency, then I know the rankings and downloads of new apps (see Appendix B for summary statistics regarding these apps). This then allows me to use these apps to predict the downloads of other apps in the market, after making an assumption about the relationship between app ranks and downloads.

Several past studies of online markets with best-seller lists assume that the rank-downloads relationship is defined by a Pareto Distribution (Chevalier and Goolsbee 2003, Chevalier and Mayzlin 2006, Garg and Telang 2012).[30] That is, there is a negative exponential distribution where an app at some rank has exponentially fewer downloads than the app at the previous rank. I can use this assumption and fit a

[29]It is not known how the lists are determined, but Google releases (http://www.adweek.com/socialtimes/google-discloses-how-search-for-google-play-works-for-the-first-time-12-percent-of-dau-search-for-apps-daily/539639?red=im) as well as anecdotal industry evidence (https://www.quora.com/What-are-ways-for-your-app-to-rank-high-in-Google-Play) suggest that they reflect the downloads of apps over the previous several days.

[30]It is possible that the Pareto distribution is not entirely correct for predicting downloads in this market. In particular, Eeckhout (2004) and Gabaix (2016), among many others show that the Pareto distribution accurately predicts the rank-size relationship for the upper tail of the distribution but not for the lower tail. There, the relationship is more accurately characterized by the log-normal distribution. However, since the Pareto distribution has only one parameter, whereas the log-normal distribution has two, the Pareto distribution is simpler to estimate.

Pareto distribution (which consists of one parameter) for every week and category around the observations that I have for new apps. I do this by running an OLS regression of the logarithm of the rank of new app j in category c at week t on the logarithm of the downloads for every category and week:

$$ln(\text{Downloads}_{jct}) = \delta_c + \delta_{month} + \beta_{month} ln(\text{Rank}_{jct}) + \mu_{jct}$$

where the δs are category and month dummies, and where μ_{jct} is a random mean zero variable representing measurement error. The β_{month} is a separate slope coefficient for every month.[31] As the dependent variable *Downloads* I use the lower bound of the bandwidth (minimum downloads in a week).[32] After running the regression (see Appendix B for results tables), and knowing the number of apps in each category in each period, I can predict the downloads of all apps in the market (e.g., if a category has 2,000 apps, I can generate a prediction for the downloads of the 2,000th app). The difficulty is in allocating the individual apps to the rankings, since only the top 500 ranks are known. I sort unranked apps based on their number of cumulative lifetime downloads and their age, and then break up any ties by randomizing.[33]

This prediction algorithm depends, in part, on variation in the rankings of apps over time. In particular, new apps should be able to enter into the rankings at different points in the distribution for me to estimate the Pareto relationship accurately. This seems to be what is happening in the data. While there is a large proportion of apps that do not change their rankings from week to week, there are also many apps that move at least two spots on a weekly basis. A graph showing the distribution in weekly changes in app rankings is in Appendix B.

3.3 Descriptive Statistics

Table 1 shows some summary statistics at the app level, where there are approximately 33.7 million app-month observations in the dataset, consisting of 2.6 million unique apps. Of these, approximately 17% belong to game categories, whereas the rest are non-game apps.

[31] I have also experimented with a single slope coefficient, as well as slope coefficients that vary by category, and the results do not change qualitatively.

[32] Results using the upper bound or an average seem to greatly overstate the number of downloads - for example, each of the top 50 apps have over 10 million downloads weekly.

[33] To check that randomization does not in itself generate any of the results, I've rerun the analysis several times with different rankings.

Table 1: **Summary statistics at the App Level**

Variable	Mean	Std. Dev.	N
App Level			
Game App	0.168	0.374	2.6 million
Paid App	0.2	0.4	2.6 million
App-Month Level			
Min. Cumulative Downloads	38,261	1.9 million	33.7 million
App Size (in MB)	21.99	29.75	33.7 million
Predicted Downloads	559	25,169	33.7 million
Number of Screenshots	4.71	3.54	33.7 million
Number of Videos	0.09	0.28	33.7 million
Mean Rating	4.0	0.66	27 million
Price (for Paid Apps)	3.27	8.93	6.8 million

Note: Mean app rating calculated for apps with 5 or more ratings.

4 Reduced Form Evidence

4.1 Data Management - Categories for Reduced Form Analysis

In order to examine entry in this market at the category level, I need to calculate the number of unique apps that appear in each month in each category. As the number of categories increased in March 2014, the categories are not directly comparable before and after March 2014. However, since I can track apps over time, I can construct "counterfactual categories" for the pre-March 2014 data. For example, even though the "Family Games" category did not exist before March 2014, since I can see all the apps that switched into this category from the old categories ("Aracde & Action", "Card & Casino", "Board & Puzzle"), I can see how many apps would have been in the "Family Games" category prior to March 2014.

Of course, if an app entered and exited the market before March 2014, it does not have a post-March 2014 category. If I ignore these apps, this creates a measurement error that would increase as I go further back in time. I deal with this issue by exploiting the detailed textual descriptions of the apps found in the data. I use a Random Forest machine learning algorithm that first maps the descriptions of the classified post-March 2014 apps into categories, and then goes back and applies this mapping to the descriptions of apps that entered and exited the market before March

2014. This is very similar to how Liu, Nekipelov, and Park (2014) map Google Play categories into Apple iTunes categories.

These "counterfactual" categories do not take into account the strategic decisions of apps, but are a purely mechanical allocation based on the apps' descriptions. It is entirely possible that if strategic decisions were taken into account, the categories would be somewhat different. Section 5 tackles some of these issues.

Table 2 shows some summary statistics at the predicted category level. There are 42 categories, 18 of which are game categories, and the rest of which are non-game. The game categories, on average, are much smaller than the non-game categories in terms of the number of apps - the average game category is less than a quarter of the size of the average non-game category. In terms of the number of downloads - the total cumulative downloads of apps in that category - they are more similar.

Table 2: **Summary Statistics at the Category-Month Level**

Variable	Mean	Std. Dev.	N
Game Categories			
Number of Apps	7,256	12,990	630
Number of New Apps	452	954	630
Non-Game Categories			
Number of Apps	32,656	30,000	840
Number of New Apps	1,522	1,902	840

Table 3 shows the average number of apps in the predicted categories for the pre-split period, roughly splitting them up into quartiles. There are wide differences in category sizes. The smallest - most "niche" - app groups, such as Role Playing and Music Games, are roughly 1 percent of the largest. What this means, in the pre-2014 actual market structure, where there are only 6 game categories, consumers specifically looking for new music apps would have to search through the Arcade & Action category or the Brain & Puzzle category in the hopes of finding a relevant app. Since those categories would have also contained bigger app groups (e.g., Arcade, or Puzzle), role playing and music games would be difficult to find.

4.2 Entry

Figure 5 below is a simple plot showing the number of new apps appearing each month in the game categories and in the non-game categories. This graph suggests that games and non-games followed very similar trends in 2012 and 2013, with similar peaks and troughs. The ratio of the number of new games to the number of

Table 3: **Average Number of Apps in Game Categories in 2012**

Quartile	Quartile	Avg. N Apps in 2012
Top Quartile	Arcade, Card, Puzzle, Casual	11,788 (5,298)
2nd Quartile	Action, Board, Casino, Racing, Sports	1,251 (1,207)
3rd Quartile	Education, Strategy, Trivia, Word	209 (18)
Smallest Quartile	Adventure, Family Music, Role Playing, Simulation	91 (49)

new apps was nearly constant, holding steady at around 0.15 (approximately 7 new apps entered for every new game), regardless of the absolute changes in the number of entrants.[34] This changed following the announcement of the split in the game categories in late December 2013. Closely following the announcement, there is an increase in entry by game apps, but not by non-game apps. This increase in entry persists and peaks right after the actual split in the game categories in March 2014, but it remains afterwards as well.[35] Starting in January 2014, the ratio of new games to new non-games experienced a shock, moving to about 0.25 for the rest of the sample period - meaning that in 2014, only 4 new apps entered for every new game that entered.

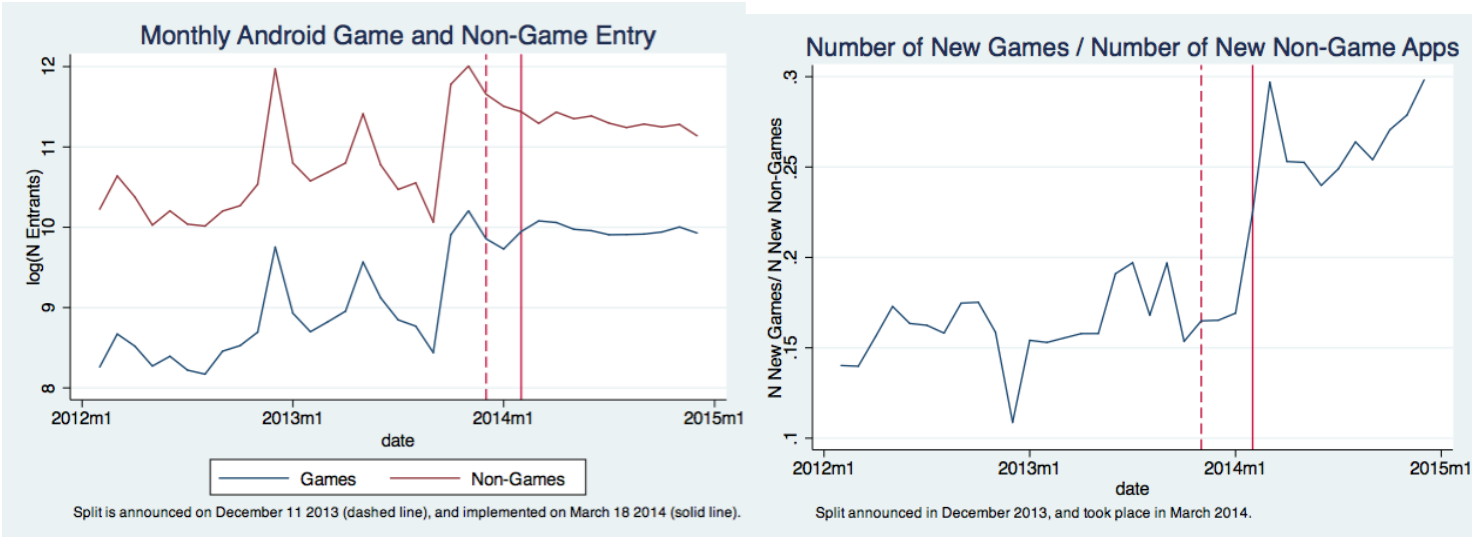
It is reasonable to interpret the change in entry as a response by developers who were incentivized by the announcement (and then creation) of new categories to produce more games rather than non-game apps.[36] This new entry could have come from multiple sources - either the developers started creating completely new apps from scratch to compete in the new categories, or the developers quickly released

[34]There can be many drivers to absolute entry levels for apps. For example, Christmas or "back-to-school" shopping, new phone or new OS releases. The key is that none of these drivers inherently favour games over apps.

[35]This is also true at the weekly level. The relevant figure is in Appendix C.

[36]The announcement did not set a strict date for the implementation of new categories, but mentioned that the change will happen in the first quarter of 2014: https://9to5google.com/2013/12/10/google-play-store-to-add-new-game-categories-in-february-2014/

Figure 5



already developed products into the market early in order to take advantage of the impending changes.

The relatively constant difference between game and non-game entry suggests that a difference-in-differences estimation strategy is appropriate to look more carefully at developers' entry decisions - comparing the decisions of the developers to produce game vs non-game apps before and after the split in the game categories occurs. The non-game categories thus serve as a control group, and the game categories serve as the treatment. As well, the panel nature of the data allows me to do a formal timing test of the treatment.

More formally, the logarithm of the number of apps that enter into category group $c \in \{GAME, NONGAME\}$ at time t - Y_{ct} - can be represented as:

$$Y_{ct} = \tau(Game_c \times Post_t) + Game_c + \delta_t + \epsilon_{ct} \tag{1}$$

where δ_t are time fixed effects, $Post_t$ is a dummy that is equal to one after the game categories have been split, and $Game_c$ is a dummy variable equal to one for the game category group and zero for non-game category group (including category and time fixed effects causes $Post_t$ to drop out of the estimating equation). The coefficient of interest in this regression is τ, which captures the treatment effect on entry for the game categories, relative to non-game (app) categories. If this coefficient is positive, then it means that developers produced more game apps as compared to non-games apps after the split.

Estimating Equation (1) pools together entry in all game and non-game categories into "category groups" (essentially it is the chart above in a regression form), but if I use app descriptions to allocate apps into post-split game categories for the pre-split period, I can run this regression at the category level. That is, c can refer to different categories (e.g., "Productivity Apps", "Sports Games", etc), as follows:

$$Y_{ct} = \tau(Game_c \times Post_t) + \delta_c + \delta_t + \epsilon_{ct} \tag{2}$$

where $Game_g$ is a dummy representing the categories which are game categories, and δ_g is a category specific fixed effect for the game categories (the average non-game category is the reference group).

Table 4 shows results from the first three regressions. Column (1) shows the regression with the data pooled across category groups, and Column (2) shows the regression for separated categories without group fixed effects, and Column (3) shows the same regression as (2) but with category fixed effects, where the reference group is the average non-treated category.

Table 4: **Regression Estimates for log(N Entrants)**

VARIABLES	(1) Category Groups	(2) Separate Cats	(3) w/ Cat FE
Games × Post Split Period	0.334***	1.466***	1.472***
	(0.072)	(0.130)	(0.101)
Games (Treated Group)	-1.706***	-2.561***	
	(0.093)	(0.092)	
Time FE	YES	YES	YES
Category FE		NO	YES
Observations	70	1,469	1,469
R-squared	0.965	0.541	0.794

Robust standard errors in parentheses
*** p<0.01, ** p<0.05, * p<0.1

The key treatment coefficient in the first column is approximately 0.3 which suggests that the following the split, developers entered 30% more apps in the game categories than in the non-game categories relative to the time before the split. This coefficient is also highly statistically significant at the 99 percent confidence level.

The treatment coefficient in Column (2) is approximately three times larger, since it looks at individual categories, which are smaller in size and thus produce higher relative changes (recall, the dependent variable is measured in logarithms). That is, 150 percent more apps entered into the average game category as compared to the average non-game category.

Appendix A, which shows the complete list of pre-split and post-split categories suggests another potential DID regression candidate. In particular, three of the game categories that existed in the pre-split period - "Sports Games", "Casual Games", and "Racing Games" - remain the same in the post-split period. As a result, they appear to be natural control group candidates for the game categories that were split. This is not the case. While the split in the other game categories may not have affected them as directly, it still changed their definition.[37] In other words, the treatment does affect the control group (in this case). The category specific treatment effects in the next two subsections suggest that the treatment did affect the three previously existing categories less on average. However, this appears to be related to their size (in terms of the number of apps), as new game categories of similar size had similar treatment effects.

4.3 Product Design

I can examine any changes to product design decisions by firms by looking at the heterogenous effects of the game category split. In particular, more niche game genres should benefit more from being split up than the larger (more mainstream) game genres. The reason is that it was harder for consumers to find more niche products (e.g., music games or educational games) as compared to more popular products (e.g., arcade games).

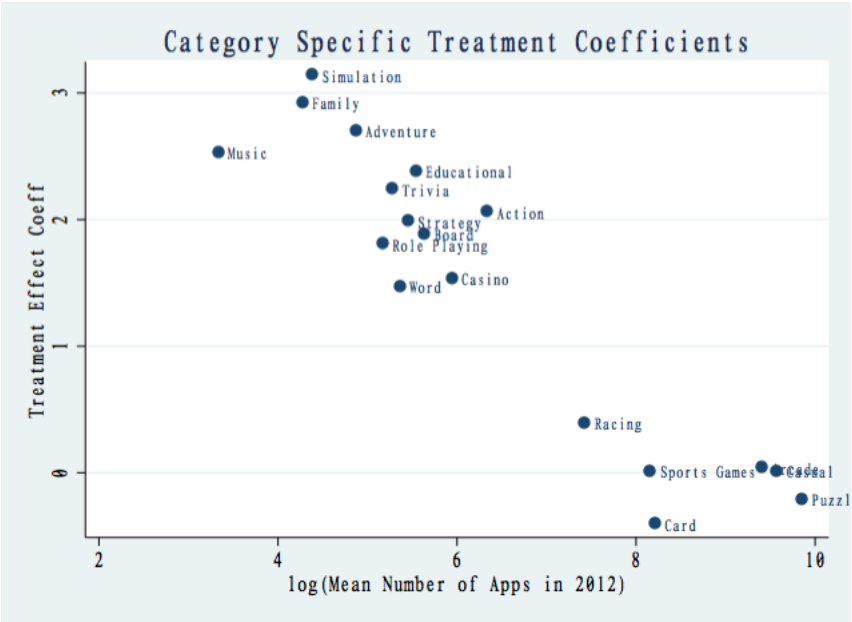
The category specific treatment effects regression is as follows:

$$Y_{ct} = \sum_{c} \tau_c \left(Post_t \times \delta_c\right) + \delta_c + \delta_t + \epsilon_{ct} \tag{3}$$

where the τ_cs are the game category specific treatment coefficients, and the rest of the notation is similar to that in the previous section. The (game) category specific treatment coefficient estimates from that regression are presented in Appendix C and in Figure 6 below.

[37]For example, some apps that were previously considered to be "Sports Games" or "Casual Games" may now consider themselves to be something else and change to one of the new categories. Then, new apps that enter the market later observe the changed roster of apps in the previously existing categories and may make different decisions as a result.

Figure 6



With the exception of a few notable categories, most of the category specific treatment effects are statistically significant at the 99 percent confidence level and positive. The regression coefficients make it clear that the change in entry was primarily driven by the change in search, since it is the smaller niche categories that have the largest coefficients. Coefficient estimates are monotonically falling with category size. Based on category size in 2012, the categories in the top quartile (e.g., Card, Casual, Puzzle) have coefficient estimates that are zero (or negative), meaning that the increase in the number of categories did not incentivize more developers to enter products there. This makes sense, since a developer who wanted to enter an "arcade" app (that is an app with characteristics which fit best into the "Arcade" category) she created before the category split would be likely to enter it into the very large category "Arcade & Action", of which "arcade" games constituted a large part. There is likely little difference in incentives between this category and the "Arcade" category in the probability of the relevant consumers finding her app. For more niche categories, the change in consumer search costs would have been large, as would the incentives of producers.

Of course, the treatment coefficients are relative, since the dependent variable is in logarithms. As such, a large treatment coefficient for the smaller categories may mean less in absolute terms than a small treatment coefficient for large categories.

However, consider that the treatment coefficients are monthly - suggesting that the Family app group, for example, grows at 300% per month as compared to the average non-game category, but it starts from a low baseline of having only 100 or so apps. Nonetheless, compared to a growth rate of the Casino category, which grows at 150% per month but started at a higher baseline of about 1,000 apps, it is conceivable that in a short period of time the number of Family apps would overtake the number of Casino apps.

These results are entirely consistent with past theoretical predictions (Bar-Isaac, Caruana and Cunat 2012, Yang 2013) which suggest that when search costs fall, producers will enter more niche products into the market as opposed to more mainstream products.

4.4 Quality

There have two proxies for quality in the data. The first proxy is more direct - the ratings that consumers give to apps. I use the app's percentage of one star reviews, following the methodology of Chevalier and Mayzlin (2006).[38] The second proxy I use is more indirect - the size of the app, in MB. This makes sense in this setting, since to give an app more features, developers have to write more code, which requires additional investment. It is possible to argue that better apps are more efficiently coded and would therefore be smaller. On the other hand, anecdotally, apps that are described as high quality integrate many features - such as Facebook or Twitter linking, or some sort of a database - that require additional lines of code and that would therefore increase the size of the app.[39] The direct and indirect measures are positively correlated - with a correlation coefficient of approximately 0.3.

4.4.1 Quality: Share of 1 Star Ratings

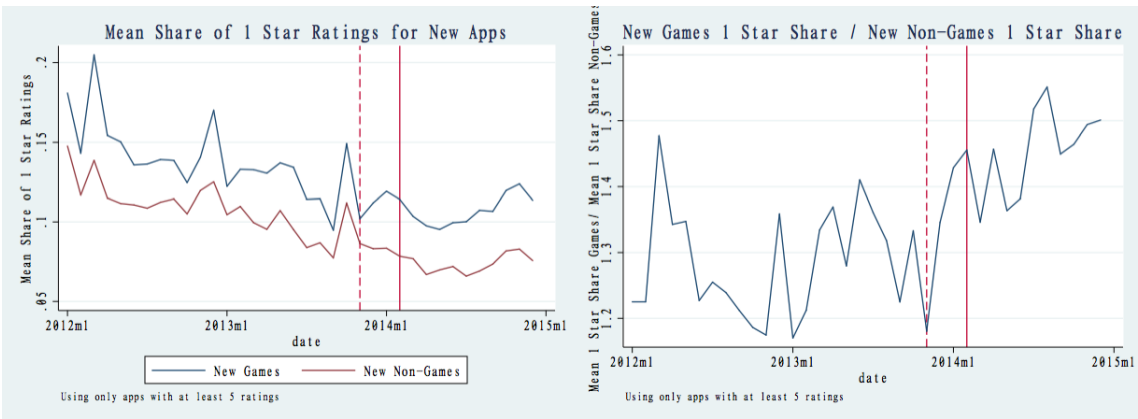
A time series of the mean share of one star reviews of new game and non-game apps (Figure 7 below) suggests that the two types of apps do not have the same share of 1-star reviews - game apps seem to be have more - these shares move very similarly, roughly until the announcement of the split in categories. Then, the mean share of 1-star game reviews starts increasing, while it falls (or stays constant) for new non-game apps. The ratio of the mean shares (games over non-games) is constant

[38] In order to avoid skewing the results, I also limit the sample to all apps that have at least 5 reviews.

[39] http://howmuchtomakeanapp.com/

over time, until the announcement at which point it begins increasing (right panel of Figure 7).

Figure 7



As for entry in the previous section, I run three sets of regressions, the results of which are presented in Table 5. Column (1) shows the regression with the data pooled across category groups, Column (2) shows the regression for separated categories without group fixed effects, and Column (3) shows the same regression as (2) but with category fixed effects, where the reference group is the average non-treated category:

The results point in the same direction as the raw data above. The average share of 1 star ratings increased after the split for games relative to non-games. Although the results at the game/non-game level are not statistically significant, the results at the category level are, and show that the share of 1 star ratings increased for game categories relative to the average non-game category by approximately 3 percentage points - suggesting a decline in quality. Considering that the average share of 1 star ratings is approximately 10%, a 3 percentage point increase is economically significant.

There are two possible explanations for this: First, it could come from a decline in relative competition. Prior to the split, a music game app, for example, had to compete with all other music game apps as well as with other kinds of game apps to succeed. Once smaller categories were added, the market was more effectively segmented and a music game app only had to compete with other music game apps. As a result, developers did not have to make music game apps that were as good. Second, Fishman and Levy (2015) suggests that the effect could be due to the search costs themselves. In their model, as search costs effectively fall for game apps, the

Table 5: **Regression Estimates for Mean App Share of 1 Star Ratings**

VARIABLES	(1) Category Groups	(2) Separate Cats	(3) w/ Cat FE
Games × Post Split Period	0.003	0.033***	0.033***
	(0.004)	(0.003)	(0.003)
Games (Treated Group)	0.031***	0.007***	
	(0.003)	(0.002)	
Time FE	YES	YES	YES
Category FE		NO	YES
Observations	70	1,468	1,468
R-squared	0.932	0.232	0.401

Robust standard errors in parentheses
*** p<0.01, ** p<0.05, * p<0.1

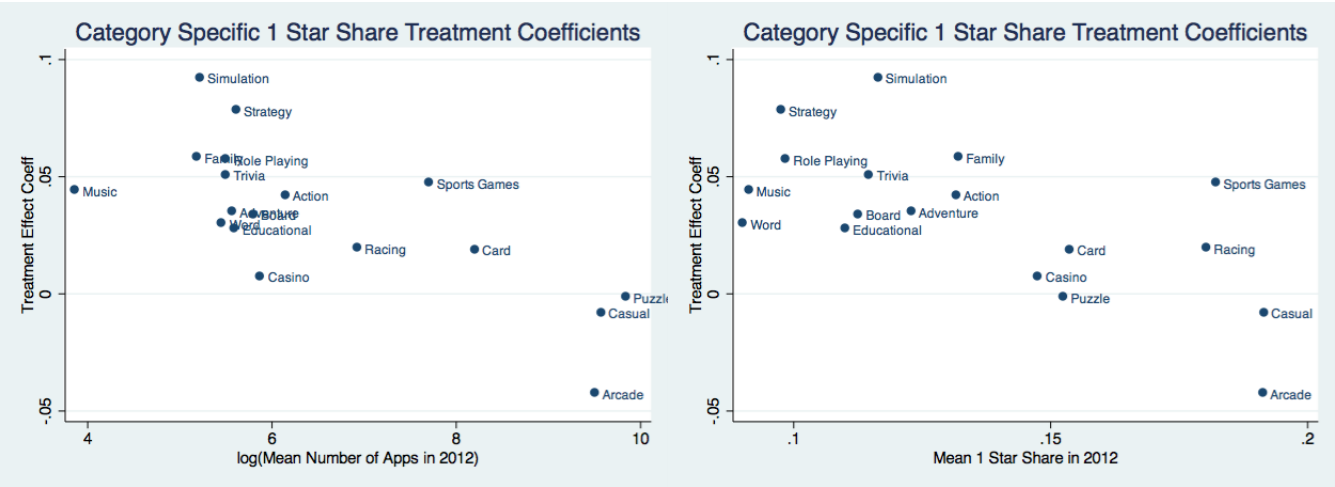
incentives to produce additional high quality products decline somewhat because consumers are more likely to look through a number of high quality products and find the best match, so the downloads for each high quality product potentially decline to the extent that it does not make sense for firms to make the investment.

I run an additional regression with category specific treatment effects, which provide further potential evidence to both of these effects in the data. The category specific treatment effects are pictured in Figure 8, where they are plotted against the average number of apps for each category in 2012 (left panel), and the average share of 1 star ratings in 2012 (right panel).

Figure 8 suggests that most of the effects came from the smaller game categories, with relatively small (or negative) average effects for the larger game categories. The largest quartile of categories includes Puzzle, Casual, and Card games. The treatment coefficients for those categories are on average close to zero or negative, as compared to the treatment coefficients for the smaller categories, which are mostly positive and statistically signifiant. Indeed, in the actual regression results the treatment coefficients for the largest four categories are also not statistically significant, suggesting that treatment has no effect on average entrant quality.[40] Similarly, the categories with the biggest treatment effects are the ones with the lowest average

[40]Note: full coefficient estimates are in Appendix C.

Figure 8



number of 1 star ratings in the pre-split period (2012) - i.e. the categories with the highest initial quality are the ones that are seeing the biggest subsequent decline in quality as a result of the split.[41]

Although I am not aware of similar previous studies that explicitly look at search costs and product quality empirically, Waldfogel (2011) looks at the quality of recorded music before and after Napster in a somewhat related setting. Napster did reduce search costs in the music industry (it became much easier to find a vast number of new songs), although it also had additional effects on copyright protection and the revenues of musicians. Waldfogel (2011) suggests that the quality of recorded music did not decrease as a result of Napster, and has in fact slightly increased since the pre-Napster period. This is quite different to my findings. The difference could be due to other changes in the music market at the time - such as reduced entry costs. These come through the reduced cost of recording songs, which could allow high quality artists with other barriers to entry to enter the market.

4.4.2 Quality: Mean Size of the App (in MB)

Table 6 shows regression results from the same type of regressions as Table 4, but using the mean size of the apps in MB as the dependent variable. The size of the app is representative of the quality of the app as described above. Although Column (1) at the game/non-game level suggests that average quality for games increased,

[41] This is not due to mean reversion for the small categories. See Appendix C for a placebo test.

Table 6: **Regression Estimates for Mean App Size (in MB)**

VARIABLES	(1) Category Groups	(2) Separate Cats	(3) w/ Cat FE
Games × Post Split Period	2.064***	-5.299***	-5.295***
	(0.601)	(1.234)	(1.074)
Games (Treated Group)	6.325***	16.160***	
	(0.411)	(1.101)	
Time FE	YES	YES	YES
Category FE		NO	YES
Observations	70	1,468	1,468
R-squared	0.949	0.265	0.554

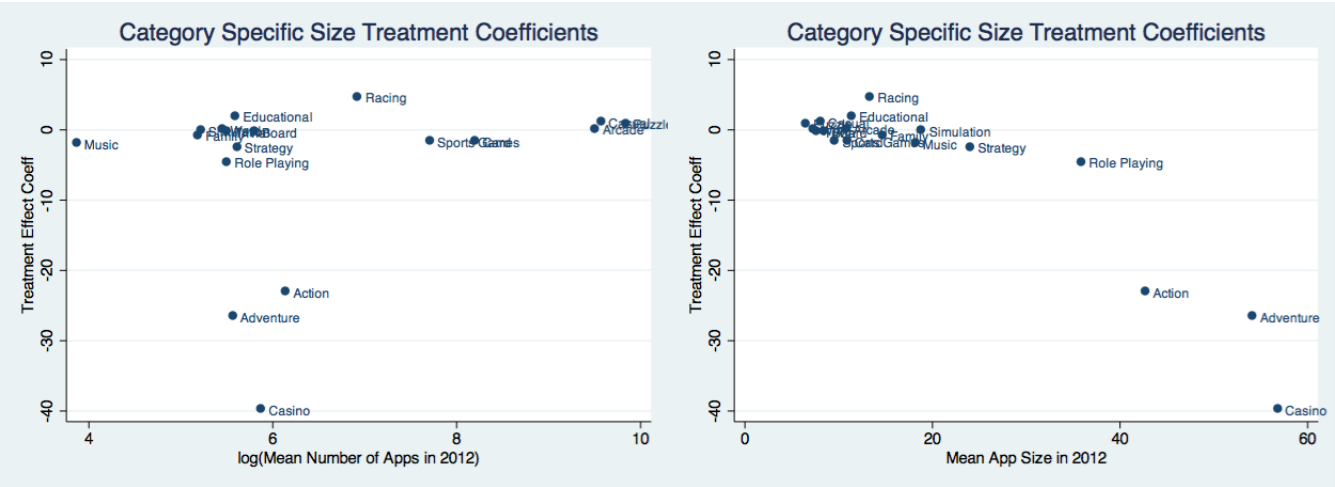
Robust standard errors in parentheses
*** p<0.01, ** p<0.05, * p<0.1

Columns (2) and (3), which are at the category level, suggest the opposite. They show that, on average, new game apps experienced a decline in average app size relative to new non-game apps following the split in the game categories, representing a relative decline in quality.

The (game) category specific heterogeneous treatment effects - presented in Figure 9 below - sheds some light on the difference between the aggregate and the category specific regression coefficients above. In particular, there is a large dispersion in the coefficients across the categories. Additionally, while most of the coefficients are negative, there are a few categories that have positive treatment effects, and which could have driven the overall results (the categories with the most negative coefficients are relatively small).

The relationship between the coefficient estimates at the category level and category size is similar to the 1 star share relationship - the correlation is roughly positive. It suggests that smaller game categories experienced more substantial drops in the quality of new apps as compared to the larger game categories. Similarly, the right panel shows the relationship between average category app size in the pre-split period, and the treatment coefficients. As in the ratings regression, it suggests that the highest quality categories - i.e. the categories with the highest average app size - experienced the biggest drop in quality after the split (had the largest negative coefficients).

Figure 9



4.5 Prices

Neither the average prices for all paid apps, nor do the average prices for new paid apps seem to follow parallel trends for games and non-games.[42] Average prices for all games fall as compared to non-games, driven by the absolute decline in game prices and the absolute increase in average non-game prices over the sample period (Figure 10). The absolute decline in game pricing can be a reflection of the increasing importance of in-app advertising and in-app purchases in the app economy. Developers view more expensive paid apps to be less profitable than less expensive paid apps or free apps, where developers recover their fixed costs more slowly over time. After the announcement of the split, the average prices of games start increasing again, meaning that the ratio of average game to non-game prices stabilizes in the post-split period.

Lower search costs from the game category split could be the cause of the price changes. In particular, notice that the average prices of new games (in the Left Panel of Figure 11) show a spike right after the split takes place. Similar to the effects on quality, this could be the result of lower competition in the market, increased demand, or of higher valuation consumers being able to discover more preferred game-apps more easily (Bar-Isaac, Caruana and Cunat 2012).

[42]The ratio of free to paid apps is constant between games and non-games. It is continuously falling for both throughout the sample.

Figure 10

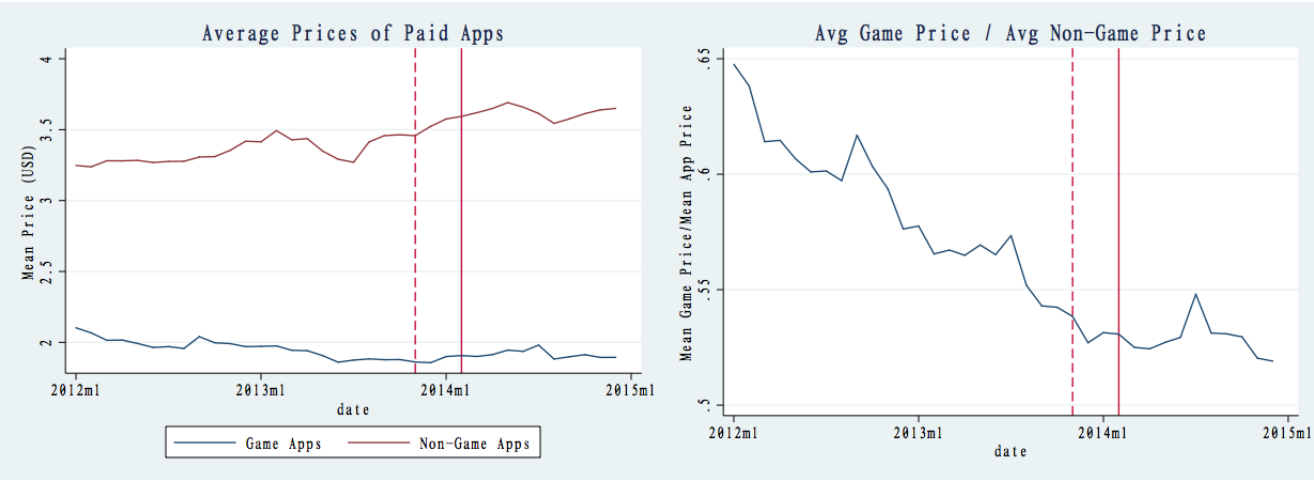
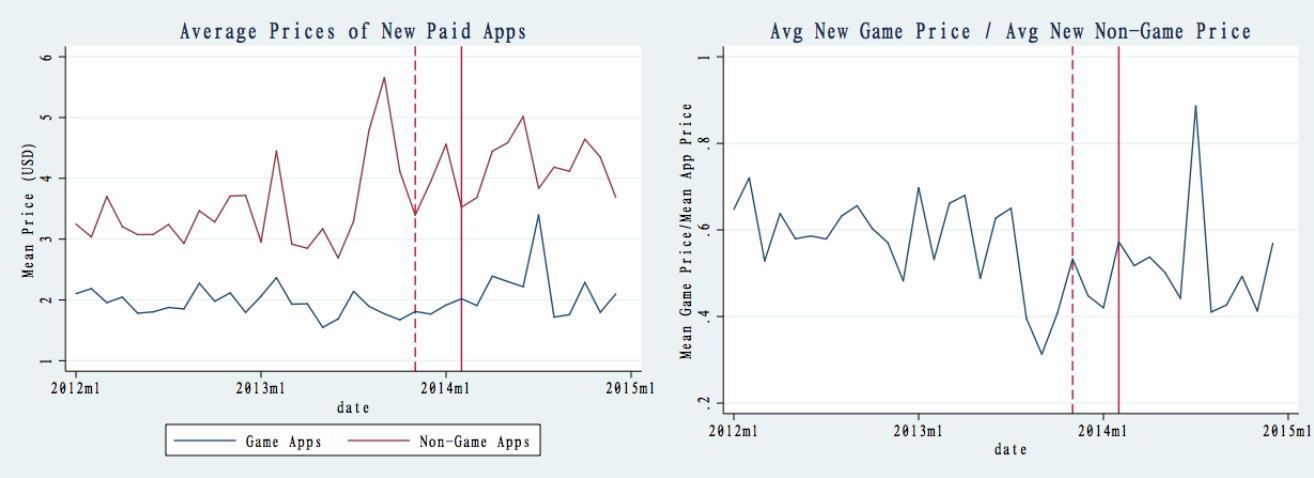


Figure 11



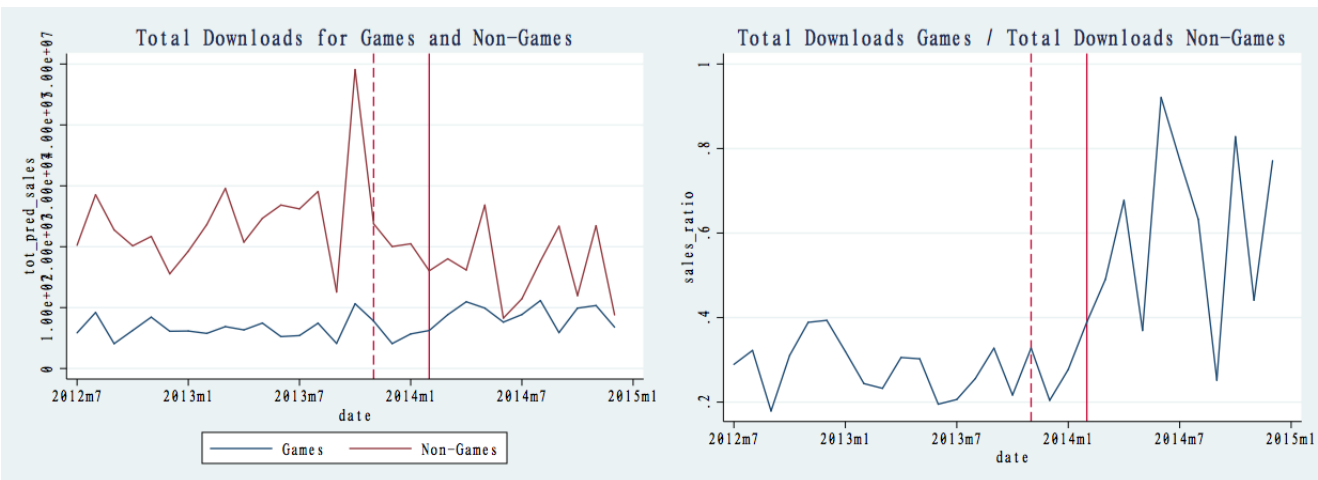
4.6 Downloads

The results in Sections 4.1-4.4 suggest that the increase in production (and reduction in quality) of game apps versus non-game apps was driven primarily by the increase in the number of game categories. The specific mechanism that would drive this change in developer incentives is the reduction in search costs for consumers and the improved visibility (and potentially easier marketing) for the products. If consumer search costs are lowered and consumers can more easily find their preferred products,

then I would expect downloads to increase for games vs non-games in the post-split period.

This is indeed the case. Figure 12 below shows the total number of downloads for games and non-games. The figure suggests that the aggregate number of downloads for games was relatively stable and trending together with the aggregate number of downloads for non-games. At the point of the split, the aggregate number of downloads for games started increasing, whereas the aggregate number of downloads for non-games stayed constant and then trended downwards slightly.[43][44]

Figure 12



5 Structural Model

5.1 Demand

The main difficulty in modelling app demand is that search naturally plays an important role in this market. Most previous literature makes assumptions about the order

[43]There is a slight pre-trend in the period prior to the split. This could be the result of higher quality games entering prior to the split occurring in order to position themselves to get a good ranking at the time of the split.

[44]If there was no market expansion, it could be possible to argue that the reduction in search costs and the increase in entry resulted in excessive entry and higher deadweight loss (Mankiw and Whinston 1986, Berry and Waldfogel 1999). Since there was market expansion, it seems that the market was initially not covered, and it is harder to claim the loss of welfare due to excess entry.

in which consumers examine products (ordered search, e.g., Hortacsu and Syverson 2004, Kim et al 2010, Bronnenberg et al 2016), or assumes that consumers arrive at products randomly (random search), to identify search parameters as well as utility parameters using aggregate data. In this case, such assumptions may not be reasonable. For example, it is difficult to justify assumptions about the order in which consumers look at products.[45] It is possible that consumers look at one category, and then switch to another category. However, a small recent literature allows for demand estimation using aggregate data without making assumptions about consumer search order, by focusing on the consideration sets of consumers (Goeree 2008, Ching, Erdem, and Keane 2009, Moraga Gonzalez, Sandor, and Wildenbeest 2015, Honka, Hortacsu, and Vitorino 2015). The following derivation of a demand model closely follows the approach of Moraga Gonzalez, Sandor and Wildenbeest (2015).

Suppose consumers search through N apps, where the utility that a consumer i receives from downloading app $j \in \{1, .., N\}$ is:

$$\begin{aligned} u_{ij} &= \delta_j + \epsilon_{ij} \\ &= X_j\beta + \xi_j + \epsilon_{ij} \end{aligned} \tag{4}$$

where X_j are observable product characteristics, ξ_j are unobservable product characteristics, and ϵ_{ij} is a random demand shock that is consumer and product specific and that is distributed iid with an extreme value type 1 distribution (mean zero, standard deviation of 1). The consumer also has the choice to not buy anything (or buy the outside option), in which case they obtain:

$$u_{i0} = \epsilon_{i0} \tag{5}$$

Initially consumers are not fully informed about all products: they do not know the ϵs. Search resolves this uncertainty. Consumers search the market by first choosing a consideration set A of products while incurring some search cost, finding out about the ϵs for those products, and then picking product j out of the subset A. The expected benefit from choosing subset A is the maximum expected utility the consumer would obtain from those products, which due to the logit error term can be expressed as following inclusive value:

$$E[\max_{r \in A} u_{ir}] = log[1 + \sum_{r \in A} exp(\delta_r)] \tag{6}$$

[45] Additionally, De los Santos et al (2012) suggests that online search is unlikely to be sequential.

where δ_r is simply $X_r\beta + \xi_r$. The consumer always has the outside option, regardless of the set they consider. To find set A, the consumer had to incur the following search costs:

$$c_{iA} = \sum_{r \in A} \gamma\psi_r + \lambda_{iA} \tag{7}$$

where ψ_r reflects the "distance" between the consumers and each product r in set A, and λ_{iA} is a consumer/choice set specific search cost shock, which is logit distributed mean zero with a standard error of 1.[46] This logit shock can be interpreted as some information shock (e.g., links) that gives consumers better access to a certain subset of products. Renaming $\sum_{r \in A} \psi_r$ as ψ_A, the utility of a consumer i of picking subset A is:

$$u_{iA} = log[1 + \sum_{r \in A} exp(\delta_r)] - \gamma\psi_A - \lambda_{iA} \tag{8}$$

The logit error term allows means that the probability consumer i picks subset A from the set of all possible subsets $\mathbb{A}$ is:

$$P_A = \frac{exp(\overline{U_A})}{\sum_{A' \in \mathbb{A}} exp(\overline{U_{A'}})} \tag{9}$$

where $\overline{U_A} = log[1 + \sum_{r \in A} exp(\delta_r)] - \gamma\psi_A$.

Similarly, the probability that a consumer picks product j from subset A can be expressed as follows:

$$P_{j|A} = \frac{exp(\delta_j)}{1 + \sum_{r \in A} exp(\delta_r)} \tag{10}$$

Product j belongs to a large number of subsets - A_j. As a result, I can express the unconditional probability of picking product j as follows:

$$s_j = \sum_{A \in A_j} P_A P_{j|A} = \sum_{A \in A_j} \frac{exp(\delta_j)}{1 + \sum_{r \in A} exp(\delta_r)} \frac{exp(\overline{U_A})}{\sum_{A' \in \mathbb{A}} exp(\overline{U_{A'}})} \tag{11}$$

$exp(\overline{U_A})$ can be expressed as $(1 + \sum_{r \in A} exp(\delta_r))exp(-\gamma\psi_A)$. It is possible to show that this expression is equivalent to the following closed form expression:[47]

[46] The γ coefficient on ψ can vary across products or product groups. However, the assumption that the "distance" of products in a consideration set is additive in the set's search costs is key for obtaining a closed form expression for the choice probabilities.

[47] A full derivation can be found in Moraga Gonzalez, Sandor and Wildenbeest (2015).

$$s_j = \frac{\frac{exp(\delta_j)}{1+exp(\gamma\psi_j)}}{1+\sum_{k\in N}\frac{exp(\delta_k)}{1+exp(\gamma\psi_k)}} \tag{12}$$

5.2 Demand Estimation - Logit

The market share of the outside option is:

$$s_0 = \frac{1}{1+\sum_{k\in N}\frac{exp(\delta_k)}{1+exp(\gamma\psi_k)}} \tag{13}$$

Following Berry (1994):

$$ln(\frac{s_j}{s_0}) = \delta_j - ln(1+exp(\gamma\psi_j)) \tag{14}$$

or:

$$ln(\frac{s_j}{s_0}) = X_j\beta - ln(1+exp(\gamma\psi_j)) + \xi_j \tag{15}$$

It is straight-forward to estimate this model using GMM, with BLP instruments accounting for the endogeneity of prices in the model. To properly specify this estimating equation, I need to decide which variables affect consumer purchasing utility but not search costs, and which variables affect search costs but not purchasing utility.

The number of products in the category of app j is an example of a search cost shifter. For example, consider products in categories with more apps (larger categories). The size of these categories should affect demand through the utility that consumers get out of the products - instead it should only affect the probability that consumers find these products. For example, if there are 10 products in a category bestseller list, it is easier for consumers to look through all 10. If there are 50 products, it takes longer for consumers to scroll through to discover all the products. Additionally, it is also easier for products to be featured in their category if there is a relatively smaller number of products in that category.

There are a number of utility shifting product characteristics that can be included in the model. The size of the app (in MB) could be such a shifter. As well, the average star rating of the app (e.g., 2 star, 3.5 star) would reveal to potential buyers the quality of the app. These can be entered as dummy variables. For example, relative to a baseline of very low rated apps (those with an average rating of 2 stars or less), a 4 star dummy would show how much more utility consumers obtain

from a higher rated app. Similarly, past literature on online markets considered information disclosure about a product as a signal of quality (Dranove and Jin 2010, Lewis 2011). With that in mind, the number of screenshots and video previews a developer includes would reveal information about the app to potential consumers. These would be two additional utility shifters.

A particularly simple logit demand model specification (Specification A) includes the product characteristics discussed above and the number of apps in the category (as the search cost parameter). More formally, the estimating equation, for app j in category c at time t, would be as follows:

$$\begin{aligned} ln(\frac{s_{jct}}{s_{0t}}) =& X_{jt}\beta + \beta_{paid}1\{p_{jt} > 0\} \\ & - \alpha p_{jt}1\{p_{jt} > 0\} \\ & - ln(1 + exp(\gamma ln(N_{ct}))) + \xi_{jt} \end{aligned} \tag{16}$$

where X_{jt} are product characteristics, and N_{ct} is the number of apps in category c. p_{jt} reflects the price of product j which is only important if that product is not free. I also allow paid products to have different utility intercepts than free products - as reflected in β_{paid}.[48]

This specification is similar to Ackerberg and Rysman (2005). Their model uses the number of products in the category to account for the mechanical increase in consumer utility with a larger array of products. The intuition is that with a larger variety, some products are "squeezed out" - for example, due to limited shelf space. This "squeezing out" effect can also be related to search, but it does not have to. Therefore, it is not clear if the γ parameter represents marginal search costs or the Ackerberg and Rysman logit model correction.

Although it would be normally difficult to disentangle the two components of γ, this setting has the benefit of a policy shock to the search costs of consumers in the market for a subset of products. Assuming that the logit correction is stable over time, a change in the γ parameter after the split would show that γ does include search costs and is not purely a functional form based adjustment.[49]

Although Specification A captures many key features of the market, it does miss an important element - persistence in product popularity due to search. If a product sold more in the past, it is more likely to be on category best-seller lists and to be featured on the store, making it easier for consumers to find. As a result, the

[48] I also run the model for free products only, since they represent 80% of the market.

[49] In fact, the change in γ would help recover a lower bound on the "search cost" component of the parameter.

past downloads of the products should play a role in the search costs of consumers. Specification B includes the lag of the downloads of product j as part of the search costs. Namely, the demand system includes the usual X_{jt} coefficients (which includes the prices), but there is an additional search cost component - the logarithm of past downloads:

$$ln(\frac{s_{jct}}{s_{0t}}) = X_{jt}\beta - ln(1 + exp(\gamma_1 ln(N_{ct}) + \gamma_2 ln(q_{jt-1})) + \xi_{jt} \tag{17}$$

where q_{jt-1} are the downloads of product j at period $t-1$. Consequently, the model is now dynamic, and it has standard dynamic panel data estimation problems, as described in Nickell (1981) and Arellano and Bond (1991). Namely, if there are persistent product level effects (i.e. unobserved app quality), then OLS or GMM estimates are inconsistent. Even if fixed effects are included (or the model is differenced), there is still correlation between the lagged q and the differenced error term.

I estimate this model as follows: first, I difference Equation (17) above, obtaining:

$$\Delta ln(\frac{s_{jct}}{s_{0t}}) = \Delta X_{jt}\beta - ln\left(\frac{1 + exp(\gamma_1 ln(N_{ct}) + \gamma_2 ln(q_{jt-1})}{1 + exp(\gamma_1 ln(N_{ct-1}) + \gamma_2 ln(q_{jt-2})}\right) + \Delta\xi_{jt} \tag{18}$$

where the Δ represents the difference between the variable in period t and the variable in period $t-1$. Then, as instruments for the differenced equation, I use ΔX_{jt}, as well as $ln(N_{ct-1})$ and $ln(q_{jt-2})$. Even though the estimating equation is non-linear, the intuition for these instruments is exactly the same as it is for the linear dynamic panel data instruments - namely, that the lag of downloads from two periods ago is correlated with the non-linear function of $N_{ct}, N_{ct-1}, ln(q_{jt-1}), ln(q_{jt-2})$ by construction, but that it should be uncorrelated with the differenced error term.[50]

A standard problem with estimating dynamic panel data models in differences is that of weak instruments. This may be an even greater problem in this application, since the instruments need to be correlated with a non-linear function of variables.[51] To improve the estimates, I follow Blundell and Bond (1998) and add to the GMM system the levels estimating equation (Equation 17), and the *differenced* lag of downloads for product j: $ln(q_{jt-1}) - ln(q_{jt-2})$. Again, this instrument is by construction correlated with the lag of the downloads of product j, but it is by

[50]Similarly to Sweeting (2013) and Aguirregabiria and Ho (2012).

[51]In the future, I will attempt to construct and use "optimal" instruments to get around this problem.

assumption uncorrelated with the error term.[52]

In Tables 7 and 8 below, I show estimates of Specification A and Specification B for Games and Non-Games before and after the split. Table 7 uses the Game app sample, and Table 8 uses the Non-Game app sample. Columns 1 to 4 show results from Specification A - Columns 1 and 3 for the pre-split data, and Columns 2 and 4 for the Post-split data. Similarly, Columns 5 to 8 show results from Specification B. The sample includes only free apps in Columns (1), (2), (5), and (6), and free as well as paid apps in columns (3), (4), (7), and (8).

Table 7: **Structural Demand Coefficients Estimates for Game Apps**

	Specification A				**Specification B**			
	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
ln(N Apps in Category)	0.679***	0.304***	0.606***	0.419***	0.649***	0.404***	0.502***	0.391***
	(0.002)	(0.001)	(0.002)	(0.001)	(0.004)	(0.001)	(0.002)	(0.001)
ln(Size)	0.044***	0.025***	0.070***	0.036***	0.051***	0.025***	0.055***	0.025***
	(0.001)	(0.001)	(0.001)	(0.001)	(0.001)	(0.001)	(0.001)	(0.001)
N Videos	0.321***	0.326***	0.242***	0.237***	0.248***	0.168***	0.233***	0.148***
	(0.003)	(0.003)	(0.003)	(0.003)	(0.003)	(0.003)	(0.002)	(0.002)
N Screenshots	0.013***	0.023***	0.024***	0.024***	0.028***	0.018***	0.023***	0.017***
	(0.000)	(0.000)	(0.000)	(0.000)	(0.000)	(0.000)	(0.000)	(0.000)
ln(lagged Downloads)					-0.149***	-0.449***	-0.230***	-0.423***
					(0.004)	(0.008)	(0.004)	(0.008)
Paid			-1.681***	-2.332***			-1.335***	-1.477***
			(0.009)	(0.005)			(0.010)	(0.018)
Price			-0.0954***	-0.0487***			-0.0818***	-0.0497***
			(0.005)	(0.002)			(0.004)	(0.002)
Avg. Star Rating Dummies	YES	YES	YES	YES	YES	YES	YES	YES
Observations	1,893,941	2,005,052	1,936,683	1,958,586	1,313,874	1,261,074	1,739,260	1,513,323
R-Squared	0.395	0.373	0.534	0.568	0.526	0.710	0.689	0.795

Columns 1-4 show Specification A. Columns 5-8 show Specification B.
Columns 1,2,5, and 6 include only free apps. Columns 3,4,7, and 8 include both free and paid apps.
Apps with 2 stars or less are the "baseline" category for the star rating dummies.
Price instruments for Columns 3, 4, 7, and 8 include the average characteristics of all other apps of the same developer.
Instruments for Columns 5, 6, 7, and 8 include the 2nd and 3rd lags of app downloads.
Robust standard errors in parentheses
*** p<0.01, ** p<0.05, * p<0.1

[52]The validity of this instrument relies on stationarity assumptions about q. Since those may be unreasonable in a fast changing market such as the Google Play store, I attempted to use alternative instruments - the difference in the lag of the Xs. Again, this instrument should be naturally correlated with the downloads of the previous period according to my model, but it should not be correlated with the error term ξ. The results using both sets of instruments are qualitatively identical.

Table 8: **Structural Demand Coefficients Estimates for Non-Game Apps**

	Specification A				Specification B			
	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
ln(N Apps in Category)	0.321***	0.312***	0.340***	0.335***	0.274***	0.286***	0.303***	0.296***
	(0.001)	(0.001)	(0.001)	(0.001)	(0.001)	(0.001)	(0.001)	(0.001)
ln(Size)	0.014***	-0.015***	0.011***	-0.012***	0.024***	0.002***	0.018***	0.001***
	(0.000)	(0.001)	(0.000)	(0.000)	(0.000)	(0.000)	(0.000)	(0.000)
N Videos	0.267***	0.306***	0.177***	0.222***	0.242***	0.205***	0.175***	0.142***
	(0.002)	(0.003)	(0.002)	(0.002)	(0.002)	(0.002)	(0.002)	(0.002)
N Screenshots	0.041***	0.011***	0.037***	0.010***	0.061***	0.014***	0.054***	0.012***
	(0.000)	(0.000)	(0.000)	(0.002)	(0.002)	(0.000)	(0.000)	(0.000)
ln(lagged Downloads)					-0.119***	-0.249***	-0.083***	-0.259***
					(0.001)	(0.001)	(0.000)	(0.001)
Paid			-2.09***	-1.849***			-2.176***	-1.489***
			(0.001)	(0.002)			(0.002)	(0.002)
Price			0.015***	0.031***			0.016***	0.029***
			(0.001)	(0.000)			(0.000)	(0.000)
Avg. Star Rating Dummies	YES	YES	YES	YES	YES	YES	YES	YES
Observations	11,933,563	10,797,648	15,729,906	12,834,380	8,312,153	7,213,606	11,269,035	8,697,018
R-Squared	0.363	0.388	0.527	0.490	0.432	0.516	0.559	0.602

Columns 1-4 show Specification A. Columns 5-8 show Specification B.
Columns 1,2,5, and 6 include only free apps. Columns 3,4,7, and 8 include both free and paid apps.
Apps with 2 stars or less are the "baseline" category for the star rating dummies.
Price instruments for Columns 3, 4, 7, and 8 include the average characteristics of all other apps of the same developer.
Instruments for Columns 5, 6, 7, and 8 include the 2nd and 3rd lags of app downloads.
Robust standard errors in parentheses
*** p<0.01, ** p<0.05, * p<0.1

The results from Specification A largely confirm the main hypothesis. The coefficient on the number of apps in the category falls from either 0.6 to approximately 0.3 or 0.4 after the split for games in Table 7 in all specifications (though the results from Specification B are less stark). This suggests that search costs fell in the games market. By contrast, the coefficient on the number of apps in the category for non-games in Table 8 remains roughly constant between the two periods.

The price coefficients in Columns (3), (4), (7) and (8) of Table 7 suggest that demand for mobile games in this market is inelastic. The median paid app has negligible market share, and a price of $1.5, meaning that demand elasticity is at most 0.15.[53] This could be due to low price variation - there are many consumers in the market, and there are also many products with very minor price differences (recall, the vast majority of products are under 5$) - consumers do not seem to be

[53]Mean paid app price is $3.3, and demand elasticity in that case is 0.3.

particularly elastic when prices change from \$1 to \$2, since these are very small changes in "real" terms.

There may also be identification issues with having a large number of very small firms and few markets and using BLP instruments. As Armstrong (2016) shows, BLP instruments can be inconsistent in these environments, since they end up having very little correlation with market shares.[54] The weak instruments could also account for the relative instability of the utility parameters in the samples which include paid apps and games in Tables 7 and 8 (Columns 3, 4, 7, and 8).

The results from Specification A suggest that the value of increasing the size of a game by 1% (equivalent to 220 KB for the median game) is between 0.5 cents and 0.8 cents for consumers; equivalently, increasing the size of an median app by 5 MB would increase the utility of consumers by 75 cents, or half the price of the median app.

Similarly, the cost to consumers of increasing the number of apps in the category by 1% was approximately 3 cents (2% of a median app's worth). This can reflect the marginal cost of "scrolling" through the app store.[55]

The coefficients on average rating dummies (not shown here for ease of reading) are positive (relative to a baseline of apps with an average star rating of 2 or less). Apps with higher average ratings give consumers more utility. The exception to this are apps with an average rating of 5 (out of 5). The coefficient for the 5 star dummy is positive, meaning that consumers receive higher utility from a "5 star app" than a "2 star app." It is, however, lower than all other coefficients, including of a "2.5 star app." The reason for this is a popular strategy among firms in online markets of buying fake 5 star ratings. Luca and Zervas (2016) show that, in the context of Yelp, restaurants with low quality are more likely to buy fake reviews.[56] This is likely what is happening in this market as well: low quality apps buy a large number of fake 5 star reviews which pushes their average rating up to being close to 5.

5.3 Period Specific Search Cost Coefficient Estimates

In addition to the regressions above that pool together observations from different periods, I also estimate *period-specific* search cost coefficients using a series of cross

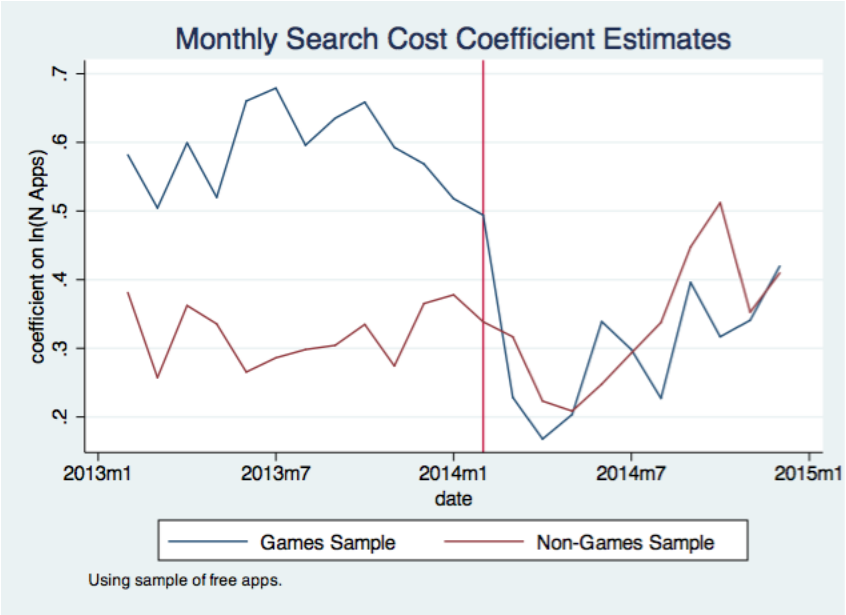
[54] In the future, I intend to use an alternative set of instruments to correct for those issues.

[55] Because the search costs do not enter the utility function linearly, the marginal rate of substitution between search cost variable ψ and price is: $\frac{MU_\psi}{MU_{price}} = \frac{exp(\gamma\psi)}{1+exp(\gamma\psi)}\frac{\gamma}{\alpha}$

[56] Li, Bresnahan and Yin (2016) show evidence that mobile apps buy fake downloads, though they do not discuss reviews.

sectional regressions separately for every month.[57] Figure 13 below plots the search cost coefficients for games and non-games over time. For this estimation I use Specification A and the sample of free games and non-games - equivalent to the first two columns of Tables 7 and 8.

Figure 13



The figure shows that the largest change in the game coefficients happens precisely at the moment of the split in the categories, rather than a gradual change over time.[58] The search cost coefficient for games hovers around 0.5-0.6 for the year prior to the split in the categories, and drops to less than 0.3 right as the categories are split.[59]

[57] I can also do the same exercise by pooling the observations from all periods together and then estimating interactions between the variables and time dummies. However, estimating a non-linear model with a large number of observations and fixed effects tends to present a computational problem, as the GMM criterion function becomes relatively flat and the discrete jumps from the fixed effects make convergence to the global minimum difficult (see Greene 2004 for a similar discussion about MLE estimation).

[58] Two representative monthly coefficients are plotted in Appendix D, and although they show some instability over time, the average coefficient level before the split is comparable to the average coefficient level after the split.

[59] There is a small estimated drop in search costs the period before the split. This may be driven by my inability to completely capture the the unobservable heterogeneity of the products. For example, if many high quality products enter the market in this period and obtain high downloads,

By comparison, the non-games search cost coefficient does not show any drastic movement around the split. There is some general instability in the coefficients over time. In particular, the figure suggests that search costs for both games and non-games were increasing in mid to late 2014. Unlike the change right after the period of the split, however, this change in the coefficients is common.

5.4 Simple Welfare Counterfactual (without Supply Side)

Consumer welfare from a set of products in a standard logit demand model can be calculated as the expected value of the products:

$$CS = ln(\sum_{j \in N} exp(\delta_j - ln(1 + exp(\gamma \psi_j)))) \tag{19}$$

this measure of consumer surplus is not just the purchase utility, but also includes the search costs associated with the different products.[60] This allows me to simply examine the change in consumer surplus as the difference between the expected utilities (which is the result of the changing coefficients):

$$\begin{aligned} \Delta CS =& ln(\sum_{j \in N} exp(\delta_j^{powt} - ln(1 + exp(\gamma^{powt} \psi_j)))) \\ &- ln(\sum_{j \in N} exp(\delta_j^{pre} - ln(1 + exp(\gamma^{pre} \psi_j)))) \end{aligned} \tag{20}$$

In addition to looking at the differences within a certain period, I can also examine the changes in total consumer surplus as a result of the policy change. The policy change should affect consumer surplus through two distinct welfare channels: the increase in product variety, and the reduction in search costs. By calculating consumer surplus (CS) in the post-split period but with the pre-split coefficients, I can estimate the changes in consumer surplus that come purely form changes in product variety. Then, the difference between this value and CS in the post-split period with the post-split coefficients is the value of only changing search costs (without changing product variety).

this could appear as an overall decrease in search costs, as more products in the category will have higher sales.

[60] This does not necessarily fully capture the search costs that consumers experience since those depend on the consideration set. It effectively assumes that consumers have consideration sets of single products, or alternatively a big consideration set of all the products.

Table 9 below provides the baseline utility estimates (for the Games market) and the changes in utility for the different estimated specifications. Since there is a different number of periods (months) in the pre-split and post-split states, surplus is calculated at the monthly level, and the table below shows the average total monthly consumer surplus in the pre-split and post-split periods. Since not all specifications include paid apps and prices, welfare is not measured in dollars. Instead, I set the average pre-split consumer surplus as a 100% baseline, and then calculate the percentage change from that baseline in the post split period. For the specifications with the paid apps, I do not account for changes in prices that result from changes in competition due to changing search costs. I assume that prices remain constant.

Table 9: **Estimated Changes in Consumer Surplus**

	(1)	(2)	(3)	(4)
Average Pre-Split CS	100%	100%	100%	100%
Change in Product Variety	+22%	+17%	+16%	+11%
Reduction in Marginal Search Costs	+40%	+12%	+28%	+9%
Overall Average Increase in CS	+62%	+29%	+46%	+ 20%

Column (1) represents Specification A without Paid Apps
Column (2) represents Specification A with Paid Apps
Column (3) represents Specification B without Paid Apps
Column (4) represents Specification B with Paid Apps

The main takeaway from the welfare results in Table 9 is the importance of changes in search costs to consumer welfare. In two of the specifications, the change in search costs contributes over 60% of the welfare gained by consumers as a result of the policy change. The reason for this large effect is the change in the search cost coefficients. For example, marginal search costs drop by half in the first two columns of Table 7. The other specifications generate smaller coefficient changes and therefore smaller welfare changes. Also, as noted in the discussion of Table 7, coefficient estimates with the paid app sample (e.g., Columns 3 and 4 of Table 7) are not very stable due to a weak instruments problem. This instability mitigates the welfare effects of the drop in search costs, and would explain the difference in results between Columns 1 and 2, and 3 and 4 in Table 9. Even in those specifications, the reduction in marginal search costs still contributes over 40% of the increase in consumer surplus.

This welfare decomposition helps answer two key questions that the reduced form results suggest: (1) Has the policy change increased consumer welfare? (2) What

is the relative contribution to consumer welfare of changing product variety versus changing marginal search costs?

I can gain additional insight by further decomposing the welfare effects of the change in product variety into the welfare effects from the increase in product variety *while holding the quality distribution constant*, and the welfare effects from the change in the quality distribution. In particular, holding the number of products fixed, consumers would prefer to have higher quality products. They then lose welfare from the reduction in product quality. However, if the distribution of product quality in the market was different, firm entry choices may be different as well. In order to do this decomposition, I need to model the supply side and firm entry decision.

Moreover, even the relatively simple decomposition in Table 9 does not assign welfare changes correctly. Specifically, though marginal search costs (the γ) do not change between Rows 1 and 2 of Table 9, the number of categories changes in the data (from 6 to 18 categories). As the number of categories increase (for a given number of products), the number of products per category falls, and search costs fall. This means that some of the welfare gains assigned to "increase in variety" in Table 9 actually come through a reduction in search costs.

To correctly decompose the welfare changes would require a counterfactual state where the marginal search costs are the same as in the pre-split period, *and* the number of categories is 18 rather than 6. This would change the entry incentives of developers and the number of products that are available in each category. As a result, it is impossible to do with the demand side of the model alone. Rather, I would need to account for entry incentives by apps, and to model and estimate the supply side of the market.

By modelling the supply side of the market, I can explore additional counterfactuals. For example, if search frictions reduce incentives to invest in app quality, it may be in the public interest to incentivize developers to create more high quality apps. This can be done by reducing fixed costs in such a way as to incentivize the entry of more higher quality products. Conversely, this counterfactual can show how policies that would raise fixed/entry costs would affect the market.

5.5 Static Entry

As described in Section 2.2, key strategic choices by developers include app characteristics and quality (how much to invest in an app), whether to enter the app into the market, whether to make the app free or paid, and what (mutually exclusive) category to enter the app into. For example, a developer who develops a non-game app can technically enter it in 24 potential categories. These categories all offer par-

ticular levels of attractiveness for the potential entrant based on the number of other firms that the app expects to compete with in the category. To model firm entry decisions, I use a simple model of entry under incomplete information.[61] For now, I will not model the choices of firms to produce free or paid products. I will focus only on free games, using the demand model from Columns (1) and (2) in Table 7.[62]

In each period, there are NP_t potential free game apps in the market. Consider the developer of app j, who chooses whether to invest in quality $w \in \{1, 2, ..., W\}$, as well as entry into categories $c \in \{1, 2, ..., C\}$ in period t. I define quality via the δ measure recovered from the demand model, which is an estimated weighted index of various app characteristics.[63] The δ is then subdivided into groups - low quality apps with low δ, medium quality apps, high quality apps, and very high quality apps (the top 1% of δ).

I will call the entry variable a_{jcwt}, which is equal to 0 or 1 if the firm chooses to enter an app of type w into category c at time t. The profits of an app j of type w from entering into category c at time t can be expressed as follows:

$$\pi_{jcwt} = \theta_1 \hat{q}(w, N_{ct}) + \theta_2 \hat{q}(w, N_{ct})^2 - FC_w - \epsilon_{jcwt} \tag{21}$$

where $\hat{q}$ are the predicted downloads from the demand estimates, and θs are the parameter which reflect how the downloads are converted into profits.[64] I use a quadratic specification to reflect the institutional background of revenues in this industry. As explained in Section 2, advertising based revenue reflects the bidding of advertisers for slots in different apps - as Varian (2007) shows, advertiser bids are convex in the desirability of the placement. This means that top apps with the largest per-period download volumes should attract the highest ad revenues, which can be exponentially larger than the ad revenues of less downloaded apps. I therefore expect the parameter θ_2 to be positive.

The downloads are a function of the number of competing apps in the category (N_{ct}), as well as of the type of the app (see previous section for more details). The number of competitors is determined jointly through the entry decisions of all firms ($N_{ct} = \sum_w \sum_j a_{jcwt}$). FC_w is a parameter representing the fixed costs of entry for

[61] I assume throughout that firms enter each app independently of the other apps that they own.

[62] Recall that the low price elasticity estimates from the demand model suggest that a majority of the firms have negative marginal costs.

[63] This is a simplifying assumption. An alternative specification would define groups around different interactions of product characteristics: e.g., apps which are less than 1 MB but have an average rating of 4 stars, and apps which are 1 to 10 MB and have an average rating of 3.5 stars. However, this form of grouping quickly increases the state space.

[64] For paid apps, there is no need for this parameter, but for free apps, this is necessary.

an app of type w, and ϵ_{jcwt} is a firm/market/time specific cost shock, assumed to be drawn iid from an Extreme Value Type 1 (logit) distribution. I assume that the shock of each firm is known to that firm, but unknown to other firms or to the econometrician. I also normalize the profits of an app from staying out of the market to zero ($\pi_{i0t} = 0$). This means that the characteristics are all estimated relative to the outside option.

Due to the presence of the idiosyncratic information shock ϵ, firms do not know the shocks that other firms receive and therefore do not know others' decisions while making their own, but they do know the distribution of the cost shock G_ϵ. A firm j can, therefore, form an idea of the *expected* profit of any other firm i in category c at time t:

$$E[\pi_{icwt}] = \theta_1 E[\hat{q}(w, N_{ct})] + \theta_2 E[\hat{q}(w, N_{ct})^2] - FC_w \tag{22}$$

where $E[\hat{q}(w, N_{ct})]$ are the *expected* sales in category c at time t. The expectation is conditional on the exogenous variables, including market characteristics, and all entry decisions in the previous periods.

The number of firms in the market is potentially endogenous to the structural error term. The reason is that a high idiosyncratic draw for a particular market for a particular firm (high ϵ_{jcwt}) could be due to some unobservable (to the econometrician) characteristics in market c that would encourage other firms to enter as well. Then, markets with high draws of ϵ would also have more competitors - and the number of competitors would be correlated with the error term. In a game of *incomplete* information with iid draws, however, the firm's information set is by assumption closer to the econometrician's information set.

More specifically, a firm enters market ct, $a_{jcwt} = 1$, if:

$$E[\pi_{jcwt}] - \epsilon_{jcwt} > max_{n \in \{1,\ldots,C\}, w' \in \{1,\ldots,W\}} E[\pi_{jnw't}] - \epsilon_{jnw't} \tag{23}$$

Assuming that the ϵs are logit distributed, this suggests that the probability of entry into category c as type w is:

$$Pr(a_{jcwt} = 1) = Pr_{cwt} = \frac{exp(E[\pi_{jcwt}])}{1 + \sum_{w' \in \{1,\ldots,W\}} \sum_{n \in \{1,\ldots,C\}} exp(E[\pi_{jnw't}])} \tag{24}$$

Since the idiosyncratic error component was the only thing that was different across different firms trying to enter the same market. Therefore, the probabilities of entry for two firms in the same market are the same; $Pr(a_{jcwt} = 1) = Pr(a_{icwt} = 1)$, and I can generalize the probability for a given market as Pr_{cwt}.

Since this probability is a function of the expected profits in a category, it is a function of the expected downloads in the category, which in turn, is a function of an expectation about the number of products. It is possible to show that the expected downloads of a firm (as a function of the number of competitors) is the following binomial expression:

$$E[\hat{q}(w, N_{ct})] = \sum_{n=1}^{NP} \binom{NP}{n} (\hat{q}(w, n)) (Pr_{cwt})^n (1 - Pr_{cwt})^{N-n} \tag{25}$$

where NP is the total number of potential entrants in the market.[65] Since the number of potential entrants is very large, and for computational simplicity, I approximate expected sales as follows: $E[\hat{q}(w, N_c t)] = \hat{q}(w, E[N_c t]) = \hat{q}(w, Pr_{cwt} NP)$. That is, I approximate the expectation of a function as a function of the expectation.[66]

Each firm has a belief about the probability Pr_{cwt} - the probability that any other firm enters the market - and it makes its own entry decisions based on that probability. In other words, this is a Bayesian Nash game, where in equilibrium, the beliefs of the players are consistent with equilibrium actions. The likelihood function, which describes the probability of observing the actual entry pattern in the data, looks as follows:

$$\mathcal{L} = \prod_c \prod_w \prod_t \left[Pr_{cwt}^{Napps_{cwt}} \right] \tag{26}$$

Taking the natural logarithm of that expression gives me the log likelihood function:

$$ln(\mathcal{L}) = \sum_c \sum_w \sum_t [Napps_{cwt} \; ln(Pr_{cwt}))] \tag{27}$$

where $Napps_{cwt}$ is the actual observed number of entrants of type w in category c at time t. The probability of entry is a function of the parameter vector as well as of all of the optimal predicted entry probabilities ($\hat{Pr}_{cwt}$). By taking first order conditions with respect to the parameter vector, it is possible to obtain parameter estimates that maximize the probability of obtaining the observed data.

The model can be estimated using an NFXP algorithm. However, it is probable that there are multiple equilibria in the data, meaning that the likelihood function is in fact a likelihood correspondence, and the estimated set of parameters is only one of a number of possible parameter values in the model. As is standard in the

[65] Which can be a very large number (e.g., 2 times the maximum number of observed firms).

[66] Although Jensen's inequality applies, preliminary simulations suggest that as the number of products increases, the approximation is close.

literature (Sweeting 2009, Bajari, Hong, Krainer and Nekipelov 2010), I assume that only one equilibrium is played in the data, and estimate the model using a two step algorithm following Hotz and Miller (1993).[67]

5.6 Estimates of Static Supply Parameters

Table 10: **Supply Parameter Estimates**

	(1)
Downloads (θ_1)	-0.008***
	(0.000)
Downloads Squared (θ_2)	1.7×10^{-6}***
	(1.61×10^{-7})
Low Quality App FC	2.9***
	(0.24)
Medium Quality App FC	4.0***
	(0.25)
High Quality App FC	4.3***
	(0.25)
Very High Quality App FC	5.9***
	(0.27)
Observations (months × categories × sizes)	1,302

Robust standard errors in parentheses
*** p<0.01, ** p<0.05, * p<0.1

Table 10 presents supply side parameter estimates. The fixed cost coefficients are monotonically increasing in the quality of the app, as expected. While the fixed costs of producing a low and medium quality app are pretty close, the fixed costs of producing a very high quality app are roughly double the fixed cost of a low quality

[67] de Paula and Tang (2012) and Aguirregabiria and Mira (2015) are among recent papers that drop the "single equilibrium" assumption. However, dropping this assumption forces the econometrician to make distributional assumptions about the potential equilibria in the model. This method would be highly computationally intensive, considering the large number of firms in the sample and the large number of potential equilibria.

app. A timing test suggests that these parameters do not significantly change over time.

5.7 Consumer Welfare Decomposition and Counterfactuals

The supply model allows for a more complete decomposition of changes in consumer welfare than the decomposition in Table 9. First, it is possible to fully decompose changes in search costs from the change in product variety.[68] Second, I can additionally decompose the welfare effects of the change in product variety into the welfare effects from the increase in product variety *while holding the quality distribution constant*, and the welfare effects from the change in the quality distribution. I do that by using the share of each size group of the total number of apps in the pre-split equilibrium as the "baseline" distribution of quality, and the share of each size group of the total number of apps in the post-split equilibrium as the new distribution of quality. I can then calculate consumer welfare measures where the total number of entrants in each category is the post-split number, but their size distribution is the pre-split distribution.

I also estimate two counterfactuals which change entry incentives for firms without changing search costs. Reports suggest that the mobile app economy created over a million jobs in the US.[69] If search frictions reduce incentives to invest in app quality, it may be in the public interest to incentivize developers to create more high quality apps. The first counterfactual generates a similar level of additional entry as the search cost reduction through a fixed/entry cost reduction of approximately 15% for all app types. The second counterfactual shows how a larger reduction of 30% in entry/fixed costs for all product types affects the market.

Table 11 shows the consumer surplus change decompositions. It is analogous to the welfare decompositions from Column (1) of Table 9.[70]

The results in Column (1) show that the increase in product variety had a small positive effect on consumer surplus - increasing it by 3% from the baseline equilibrium. This is an order of magnitude smaller than what was called the "variety" effect in Table 8, which suggests that changes in search costs generated most welfare effects in the model. As before, there is a large effect from the reduction in in marginal search costs for consumers. There is also a reduction in total search costs that comes

[68] Recall, Table 9 includes some measure of changing search costs in the product variety effect.

[69] http://www.asymco.com/2015/01/22/bigger-than-hollywood/

[70] To evaluate the other model specifications, I have to model changes not just in the app quality distribution, but also consider changes in prices and in lagged-downloads. To do that, I need price estimates that do not imply negative marginal costs.

Table 11: **Estimated Changes in Consumer Surplus**

	(1) Search Cost Policy	(2) -15% Fixed Cost Policy	(3) -30% Fixed Cost Policy
Pre-Split CS	100%	100%	100%
Increase in Variety (Pre-Split Quality Dist.)	+3%	+3%	+3.5%
Change in Quality Distribution	-1.5%	+1%	+2%
Increase in Number of Categories	+14%		
Reduction in Marginal Search Costs	+50%		
Overall Increase in CS	+65.5%	+4%	+5.5%

from the increase in the number of categories, and that also increases welfare. Again, overall, the net impact of the change in the number of game categories on consumer surplus is strongly positive.

The reduced form results raise a question about the relative welfare effects of changes in product variety and product quality. Counterfactual simulations using the structural model show that while the increase in the share of low quality products reduces consumer surplus by 1.5% relative to its pre-equilibrium baseline, it is smaller than the effect of the increase in product variety.

A useful comparison is with the second counterfactual (in Column 2), where the fixed costs of product development (or entry) fall by 15%. Here, the net increase in consumer welfare is much smaller - about 4% relative to the baseline. The contribution of the increase in variety - 3% - is the same as in Column (1), which makes sense since this counterfactual is designed to generate the same level of entry as in the search cost counterfactual (see Table 12 below). However, unlike Column (1), the contribution of the change in quality distribution is positive, an increase of about 1% relative to the pre-split consumer surplus. This occurs since the reduction in fixed costs allows additional more high quality apps to enter the market.

A large part of why the change in welfare in Column (2) is so small is because search costs do not change. This is important, since the number of products almost doubles. If the number of categories increase, the number of products per category in the search cost counterfactual mostly fall. This reduces the search costs that consumers expend per product (ψ_j in the demand model). In the fixed cost counterfactual, these search costs increase, which offsets the effect of larger product variety.

Column (3) in Table 11 shows the final counterfactual - reducing entry costs by double the previous amount, by 30%. Results show this counterfactual results in

higher consumer welfare - the overall gains are 5.5% in Column (3), as opposed to 4% in Column (2). The gains from variety do not increase substantially. Column (5) in Table 12 suggests that this may be in part because the share of very high quality apps does not increase by much. In addition, the number of products doubles relative to the baseline, but this also has a negative effect on welfare through search costs. If the number of products double, consumers find it harder to find certain products, and the gains from variety are relatively limited.

These counterfactuals are meaningful, despite their smaller welfare effects. There are two reasons for that: First, comparing Column (2) or Column (4) in Table 9 to Column (1) in Table 11 suggests that the effects of changes in marginal search costs may be overstated in Table 11. Second, there may be decreasing returns to additional changes in consumer search costs - consumer surplus will likely not increase by an additional 60% from tripling the number of categories again. At some point, browsing through a large number of categories may make it *harder* for consumers to find the products they are looking for. Consider the extreme case where each app is its own category, for example. Overall, the positive welfare effects of reducing fixed costs suggest that the two policies may be complementary - a reduction in search costs which incentivizes higher quality firms to enter the market can mitigate the effects of a reduction in search costs which allows smaller and lower quality firms to enter the market.

Table 12: **Equilibrium App Distribution**

	(1) Baseline	(2) Post-Split	(3) Post-Split (Orig. Dist)	(4) 15% Reduction Entry Cost	(5) 30% Reduction Entry Cost
Share of Low Quality Apps	0.64	0.75	0.64	0.59	0.54
Share of Medium Quality Apps	0.20	0.15	0.20	0.22	0.24
Share of High Quality Apps	0.15	0.08	0.15	0.17	0.20
Share of V. High Quality Apps	0.01	0.01	0.01	0.01	0.02
Total Number of Apps	111,261	162,472	162,472	157,582	206,382

6 Conclusion

This paper presents evidence regarding the effect of search on competition, entry, quality, and product design in online markets, using detailed data from the Google

Play mobile app store. It uses an exogenous shock whereby consumer search improved for some product types (games) in the Google Play store but not for others (non-games).

The data shows that lower search costs increase product entry and that niche products drive most of the entry effects. This is consistent with findings from IO theory (e.g., Bar-Isaac et al 2012), and it is the first evidence on this topic in the empirical literature. However, I find that the average quality of the affected products falls relative to the unaffected products following the split.

I set up a structural demand model which estimates consumer choice coefficients related to utility and to search in order to measure the overall welfare effects, and the contribution of different factors to welfare. I find that, consistently with the reduced form results, the search costs associated with game apps declined after the split in the game categories. I also set up an entry model with an incomplete information framework where apps choose to enter into different categories and invest in quality based on expected profits (and downloads).

Counterfactual simulations show that the policy change increased consumer welfare in the mobile app market; a decomposition of these effects shows that most of the increase came from a reduction in marginal search costs. There is also an increase in welfare due to the increase in greater product variety, and a smaller fall in welfare due to lower product quality. Further counterfactual simulations - reducing fixed costs - show that welfare gains are modest without changing search costs.

From a policy perspective, the results suggest two consumer welfare consequences to policies that *increase* search costs online: First, consumer welfare falls because of foreclosure, which is not fully offset by the higher quality of the firms that do enter the market. Second, consumer welfare falls due to higher consumer marginal search costs. Overall, the results of this paper suggest that policies which increase search costs in online markets may have strongly detrimental effects to consumer welfare.[71]

[71] Google's anti-trust case may provide an interesting exception. Although Google's policies in the search engine market increase consumer search costs, there are additional benefits to Google's search engine policies that my model does not capture. For example, the quality of Google's products is arguably higher than their competitors' qualities. Thus, by presenting the highest quality product first, Google minimizes the total search costs consumers pay.

References

Ackerberg, D. A. and M. Rysman (2005). Unobserved product differentiation in discrete-choice models: estimating price elasticities and welfare effects. *RAND Journal of Economics 36*(4), 771–789.

Aguiar, L. and J. Waldfogel (2016). Even the losers get lucky sometimes: New products and the evolution of music quality since napster. *Information Economics and Policy 34*, 1–15.

Aguirregabiria, V. and C.-Y. Ho (2012). A dynamic oligopoly game of the us airline industry: Estimation and policy experiments. *Journal of Econometrics 168*(1), 156–173.

Aguirregabiria, V. and P. Mira (2015). Identification of games of incomplete information with multiple equilibria and common unobserved heterogeneity. *University of Toronto Department of Economics Working Paper 474*.

Anderson, S. P. and R. Renault (1999). Pricing, product diversity, and search costs: A bertrand-chamberlin-diamond model. *The RAND Journal of Economics*, 719–735.

Arellano, M. and S. Bond (1991). Some tests of specification for panel data: Monte carlo evidence and an application to employment equations. *The review of economic studies 58*(2), 277–297.

Armstrong, M., J. Vickers, and J. Zhou (2009). Prominence and consumer search. *The RAND Journal of Economics 40*(2), 209–233.

Armstrong, T. (2016). Large market asymptotics for differentiated product demand estimators with economic models of supply. *Econometrica (Forthcoming)*.

Athey, S. and G. Imbens (2015). Machine learning methods for estimating heterogeneous causal effects.

Augereau, A., S. Greenstein, and M. Rysman (2006). Coordination versus differentiation in a standards war: 56k modems. *The RAND Journal of Economics 37*(4), 887–909.

Bajari, P., H. Hong, J. Krainer, and D. Nekipelov (2009). Estimating static games of incomplete information. *Journal of Business and Economic Studies*.

Bar-Isaac, H., G. Caruana, and V. Cuñat Martinez (2012). Search, design and market structure. *American Economic Review 102*(2), 1140–1160.

Berry, S. T. (1994). Estimating discrete-choice models of product differentiation. *The RAND Journal of Economics*, 242–262.

Berry, S. T. (1999). Free entry and social inefficiency in radio broadcasting. *The RAND Journal of Economics 30*(3), 397–420.

Blundell, R. and S. Bond (1998). Initial conditions and moment restrictions in dynamic panel data models. *Journal of econometrics 87*(1), 115–143.

Bronnenberg, B., J. B. Kim, and P. Albuquerque (2016). The probit choice model under sequential search with an application to online retailing. *Management Science*.

Brynjolfsson, E., Y. Hu, and D. Simester (2011). Goodbye pareto principle, hello long tail: The effect of search costs on the concentration of product sales. *Management Science 57*(8), 1373–1386.

Brynjolfsson, E. and M. D. Smith (2000). Frictionless commerce? a comparison of internet and conventional retailers. *Management science 46*(4), 563–585.

Cachon, G. P., C. Terwiesch, and Y. Xu (2008). On the effects of consumer search and firm entry in a multiproduct competitive market. *Marketing Science 27*(3), 461–473.

Chandra, A. and M. Tappata (2011). Consumer search and dynamic price dispersion: an application to gasoline markets. *The RAND Journal of Economics 42*(4), 681–704.

Chen, Y. and C. He (2011). Paid placement: Advertising and search on the internet. *The Economic Journal 121*(556), F309–F328.

Chevalier, J. and A. Goolsbee (2003). Measuring prices and price competition online: Amazon. com and barnesandnoble. com. *Quantitative Marketing and Economics 1*(2), 203–222.

Chevalier, J. A. and D. Mayzlin (2006). The effect of word of mouth on sales: Online book reviews. *Journal of marketing research 43*(3), 345–354.

Ching, A., T. Erdem, and M. Keane (2009). The price consideration model of brand choice. *Journal of Applied Econometrics 24*(3), 393–420.

Chiou, L. (2015). Vertical integration and antitrust in search markets. *Available at SSRN 2663557*.

Crawford, G. S., R. S. Lee, M. D. Whinston, and A. Yurukoglu (2015). The welfare effects of vertical integration in multichannel television markets. Technical report, National Bureau of Economic Research.

De los Santos, B., A. Hortaçsu, and M. R. Wildenbeest (2012). Testing models of consumer search using data on web browsing and purchasing behavior. *The American Economic Review 102*(6), 2955–2980.

De Paula, A. and X. Tang (2012). Inference of signs of interaction effects in simultaneous games with incomplete information. *Econometrica 80*(1), 143–172.

Diamond, P. A. (1971). A model of price adjustment. *Journal of economic theory 3*(2), 156–168.

Dranove, D. and G. Z. Jin (2010). Quality disclosure and certification: Theory and practice. *Journal of Economic Literature 48*(4), 935–963.

Eeckhout, J. (2004). Gibrat's law for (all) cities. *American Economic Review*, 1429–1451.

Ellison, G. and S. F. Ellison (2005). Lessons about markets from the internet. *Journal of Economic Perspectives*, 139–158.

Ferreira, F. and J. Waldfogel (2013). Pop internationalism: has half a century of world music trade displaced local culture? *The Economic Journal 123*(569), 634–664.

Fishman, A. and N. Levy (2015). Search costs and investment in quality. *The Journal of Industrial Economics 63*(4), 625–641.

Gabaix, X. (2016). Power laws in economics: An introduction. *Journal of Economic Perspectives 30*(1), 185–206.

Garg, R. and R. Telang (2012). Inferring app demand from publicly available data. *MIS Quarterly, Forthcoming*.

Ghose, A., A. Goldfarb, and S. P. Han (2012). How is the mobile internet different? search costs and local activities. *Information Systems Research 24*(3), 613–631.

Goeree, M. S. (2008). Limited information and advertising in the us personal computer industry. *Econometrica 76*(5), 1017–1074.

Goldmanis, M., A. Hortaçsu, C. Syverson, and Ö. Emre (2010). E-commerce and the market structure of retail industries*. *The Economic Journal 120*(545), 651–682.

Greene, W. (2004). The behaviour of the maximum likelihood estimator of limited dependent variable models in the presence of fixed effects. *The Econometrics Journal 7*(1), 98–119.

Hendricks, K. and A. Sorensen (2009). Information and the skewness of music sales. *Journal of Political Economy 117*(2), 324–369.

Honka, E., A. Hortaçsu, and M. A. Vitorino (2015). Advertising, consumer awareness, and choice: Evidence from the us banking industry. *Consumer Awareness, and Choice: Evidence from the US Banking Industry (September 16, 2015).*

Hortaçsu, A. and C. Syverson (2004). Product differentiation, search costs, and competition in the mutual fund industry: A case study of s&p 500 index funds. *The Quarterly Journal of Economics 119*(2), 403–456.

Hotz, V. J. and R. A. Miller (1993). Conditional choice probabilities and the estimation of dynamic models. *The Review of Economic Studies 60*(3), 497–529.

Ipsos (2013). Mobile app marketing insights: How consumers really find and use your apps. Technical report, Ipsos and Google.

Kim, J. B., P. Albuquerque, and B. J. Bronnenberg (2010). Online demand under limited consumer search. *Marketing science 29*(6), 1001–1023.

Koulayev, S. (2014). Search for differentiated products: identification and estimation. *The RAND Journal of Economics 45*(3), 553–575.

Larson, N. (2013). Niche products, generic products, and consumer search. *Economic Theory 52*(2), 793–832.

Levin, J. D. (2011). The economics of internet markets. Technical report, National Bureau of Economic Research.

Lewis, G. (2011). Asymmetric information, adverse selection and online disclosure: The case of ebay motors. *The American Economic Review 101*(4), 1535–1546.

Li, X., T. Bresnahan, and P.-L. Yin (2016). Paying incumbents and customers to enter an industry: Buying downloads. *Available at SSRN*.

Liu, Y., D. Nekipelov, and M. Park (2014). Timely versus quality innovation: The case of mobile applications on itunes and google play. *NBER Working Paper*.

Luca, M. and G. Zervas (2016). Fake it till you make it: Reputation, competition, and yelp review fraud. *Management Science*.

Mankiw, N. G. and M. D. Whinston (1986). Free entry and social inefficiency. *The RAND Journal of Economics*, 48–58.

Moraga Gonzalez, J. L., Z. Sándor, and M. R. Wildenbeest (2015). Consumer search and prices in the automobile market.

Nelson, P. (1970). Information and consumer behavior. *Journal of political economy 78*(2), 311–329.

Nickell, S. (1981). Biases in dynamic models with fixed effects. *Econometrica: Journal of the Econometric Society*, 1417–1426.

Nielsen (2011). State of the media: Mobile usage trends. Technical report.

Seim, K. (2006). An empirical model of firm entry with endogenous product-type choices. *RAND Journal of Economics*, 619–640.

Sorensen, A. T. (2007). Bestseller lists and product variety*. *The journal of industrial economics 55*(4), 715–738.

Stahl, D. O. (1989). Oligopolistic pricing with sequential consumer search. *The American Economic Review*, 700–712.

Stigler, G. J. (1961). The economics of information. *The journal of political economy*, 213–225.

Sweeting, A. (2009). The strategic timing incentives of commercial radio stations: An empirical analysis using multiple equilibria. *The RAND Journal of Economics 40*(4), 710–742.

Sweeting, A. (2013). Dynamic product positioning in differentiated product markets: The effect of fees for musical performance rights on the commercial radio industry. *Econometrica 81*(5), 1763–1803.

Varian, H. R. (2007). Position auctions. *international Journal of industrial Organization 25*(6), 1163–1178.

Waldfogel, J. (2011). Bye, bye, miss american pie? the supply of new recorded music since napster. Technical report, National Bureau of Economic Research.

Wildenbeest, M. R. (2011). An empirical model of search with vertically differentiated products. *The RAND Journal of Economics 42*(4), 729–757.

Yang, H. (2013). Targeted search and the long tail effect. *The RAND Journal of Economics 44*(4), 733–756.

Zentner, A., M. Smith, and C. Kaya (2013). How video rental patterns change as consumers move online. *Management Science 59*(11), 2622–2634.

Zhou, J. (2014). Multiproduct search and the joint search effect. *The American Economic Review 104*(9), 2918–2939.

7 Appendix A: List of Categories

Table 13: Google Play Game Categories Before and After March 2014

Before	After
Casual	Casual
Racing	Racing
Sports	Sports
Arcade & Action	Action
	Adventure
	Arcade
	Board
Cards & Casino	Card
	Casino
	Education
	Family
	Music
	Live Wallpaper
Brain & Puzzle	Puzzle
	Role Playing
	Simulation
	Strategy
	Trivia
	Word

Table 14: Google Play Non-Game Categories

Books & Reference	Libraries & Demo	Productivity
Business	Lifestyle	Shopping
Comics	Media & Video	Social
Communications	Medical	Sports
Education	Music & Audio	Tools
Entertainment	News & Magazines	Transportation
Finance	Personalization	Travel & Local
Health & Fitness	Photography	Weather

8 Appendix B

8.1 Download Ranges

Table 15: List of Cumulative Download Ranges

Lower Bound	Upper Bound
1	5
5	10
10	50
50	100
100	500
500	1,000
1,000	5,000
5,000	10,000
10,000	50,000
50,000	100,000
100,000	500,000
500,000	1 million
1 million	5 million
5 million	10 million
10 million	50 million
50 million	100 million
100 million	500 million
500 million	1 billion

8.2 Summary Statistics of New Apps

Table 16: **Summary Stats of New Apps at Weekly Level**

Variable	Mean	Std. Dev.	Min	Max	N Obs
Games					
Download Lower Bound	27,802	245,351	0	1 million	10,863
Rank	256	147	1	500	10,863
Non-Games					
Download Lower Bound	13,982	414,378	0	5 million	17,375
Rank	278	140	1	500	17,375

8.3 Regression Results Predicting Downloads for New Apps

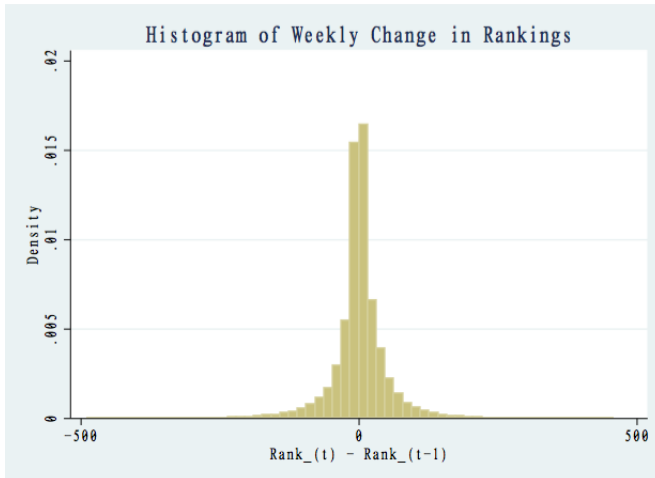
Table 17: **Regression Results on ln(Downloads)**

	(1) Games	(2) Non-Games
ln(Rank)	-1.168***	-0.926***
	(0.079)	(0.055)
Month FE	YES	YES
Category FE	YES	YES
ln(Rank) interactions with Month FE	YES	YES
Observations	10,020	16,738
R-squared	0.490	0.530

Robust standard errors in parentheses
*** p<0.01, ** p<0.05, * p<0.1

8.4 Weekly Changes in App Rankings

Figure 14



9 Appendix C: Additional Reduced Form Results

9.1 Weekly Game and Non-Game App Entry

9.2 Test of the Timing of the Entry Treatment Effect

Since I have monthly data, I can test whether the treatment effect indeed occurred in March 2014, and is not something that hits in an earlier period (say, in mid 2013) but is still absorbed by the treatment dummy. To do that, I allow the treatment effect to vary by year by introducing interactions between time dummies and the treatment dummy, effectively estimating a potential treatment effect for every month (relative to some early baseline period). The estimating equation will then change slightly to become:

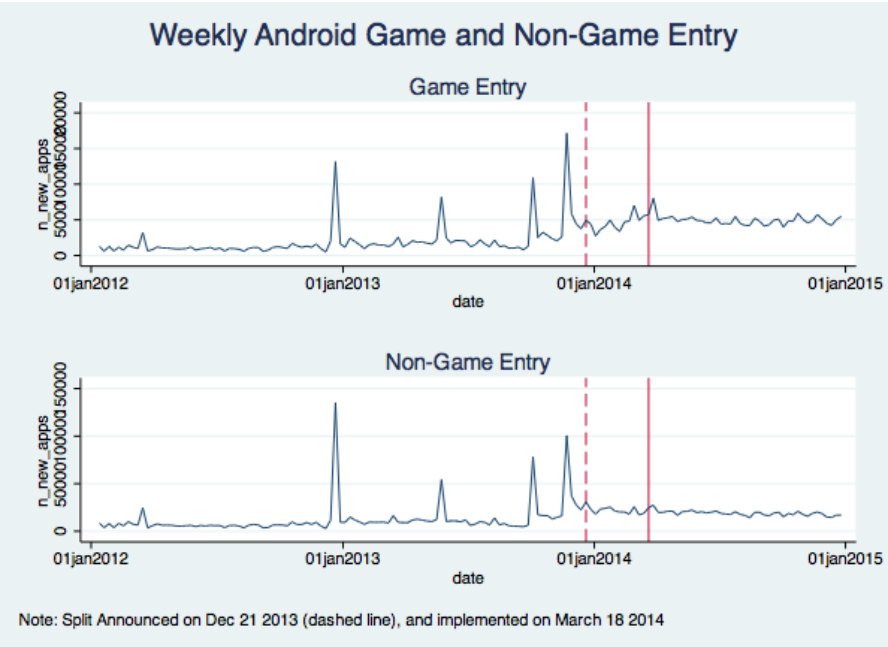
$$Y_{ct} = \sum_t \tau_t \ (Game_c \times \delta_t) + \delta_c + \delta_t + \epsilon_{ct} \tag{28}$$

where the τ_ts now capture period specific treatment effects, relative to a baseline period - which I consider to be March 2012, the first period of the sample.[72]

Figure 8 above below the period specific treatment effects. It suggests that the main entry effect hits right around the period of the increase in the number of categories. Indeed, there are two periods of pre-split positive effects (one of those statistically significant at the 95 percent level), but these can be explained since the

[72] I can use different baselines, which give similar results.

Figure 15

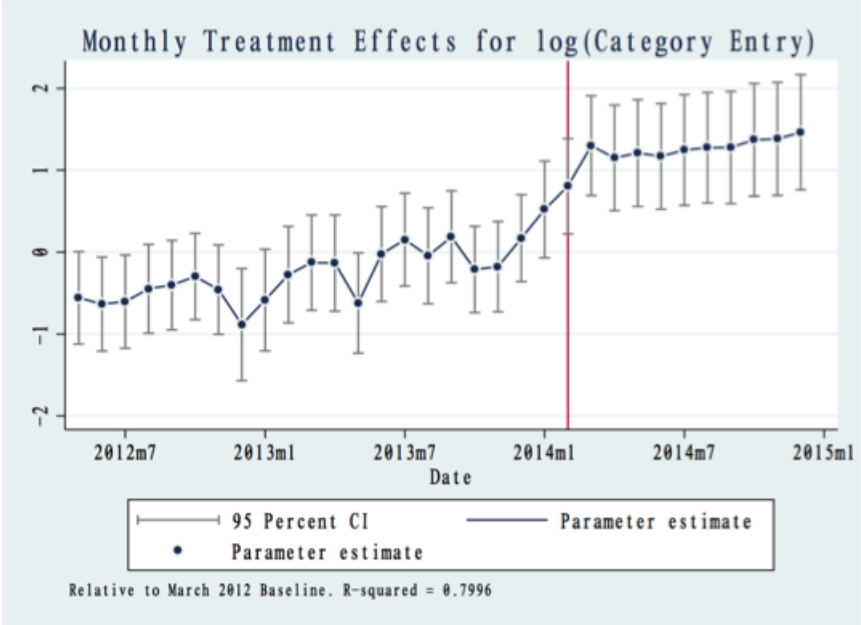


split was announced in December 2013, meaning that some firms may have entered the market early to be better positioned for when the split takes place.

9.3 Category Specific Entry Treatment Coefficient Estimates

9.4 Category Specific Size Treatment Coefficient Estimates

Figure 16



9.5 Placebo Tests for Quality

While the reduced form results are suggestive, some of the category specific results in the previous sections, particularly those that rely on category size, could be driven by mean reversion or the size of the categories rather than the treatment effect. To disprove this I run a placebo test.

I split my three year sample into two - the first part from January 2012 to December 2013, and the second part from January 2013 to December 2014. The split in the categories happens during the second part of the sample (in March 2014).

Consequently, I run a category specific regression (analogous Column 2 in Tables 4 and 5) for both parts of the split sample. In the 2012/3 sample, I set up a placebo treatment to start in March 2013 (exactly one year before the true treatment starts). I expect that the placebo treatment will generate zero treatment effects, whereas the true treatment in the second part of the sample should generate non-zero treatment effects.

The following table shows the regression results for the share of 1 star ratings, and for the mean size of the app for the two periods. Columns (1) and (3) show the 2012/3 regressions, whereas Columns (2) and (4) show the 2013/4 regressions.

Both placebo treatment effects in Columns (1) and (3) are not statistically sig-

Table 18: **Category Specific Treatment Effects on log(N Entrants)**

Category Name	τ_g	Category Name	τ_g
Arcade	0.040	Educational	2.383***
	(0.106)		(0.137)
Card	-0.405**	Strategy	1.985***
	(0.163)		(0.123)
Casual	0.011	Trivia	2.241***
	(0.100)		(0.132)
Puzzle	-0.209*	Word	1.474***
	(0.113)		(0.132)
Action	2.061***	Adventure	2.702***
	(0.112)		(0.143)
Board	1.883***	Family	2.923***
	(0.116)		(0.168)
Casino	1.536***	Music	2.532***
	(0.151)		(0.204)
Racing	0.395***	Role Playing	1.804***
	(0.117)		(0.119)
Sports Games	0.010	Simulation	3.140***
	(0.143)		(0.160)
Observations	1,469		
R-squared	0.829		

Robust standard errors in parentheses
*** p<0.01, ** p<0.05, * p<0.1

nificant. In particular, the placebo treatment for the 1 star rating regression is a full order of magnitude lower than the "true" treatment, and the placebo treatment for the mean size regression is going in the opposite direction of the "true" treatment effect (although not statistically significant).

I also run the category specific treatment effects for the placebo period for the share of 1 star ratings and for the mean size of the apps. These results are pictured in the two figures below. For the placebo regression, the category specific treatment effects are plotted against the mean number of apps in the category in 2012, and for the "true" regression, the category specific treatment effects are plotted against

Table 19: **Category Specific Treatment Effects on App Size (in MB)**

Category Name	τ_g	Category Name	τ_g
Arcade	-3.58***	Educational	-4.68***
	(1.21)		(1.36)
Card	-0.825	Strategy	-6.18***
	(2.82)		(2.0)
Casual	-3.02***	Trivia	-1.27
	(0.853)		(1.75)
Puzzle	-1.55	Word	-4.66**
	(1.08)		(1.82)
Action	-27.2***	Adventure	-30.4***
	(5.4)		(8.5)
Board	-4.84***	Family	-7.72***
	(1.52)		(1.14)
Casino	-48.7***	Music	-2.67
	(7.1)		(4.42)
Racing	-9.94***	Role Playing	-11.84**
	(1.36)		(4.4)
Sports Games	-3.8*	Simulation	-6.22***
	(2.0)		(1.66)
Observations	1,469		
R-squared	0.560		

Robust standard errors in parentheses
*** p<0.01, ** p<0.05, * p<0.1

the mean number of apps in the category in 2013. The ordering of the categories does not change between 2012 and 2013, meaning that the relationship between the treatment coefficients and category size should be comparable between the two sets of regressions.

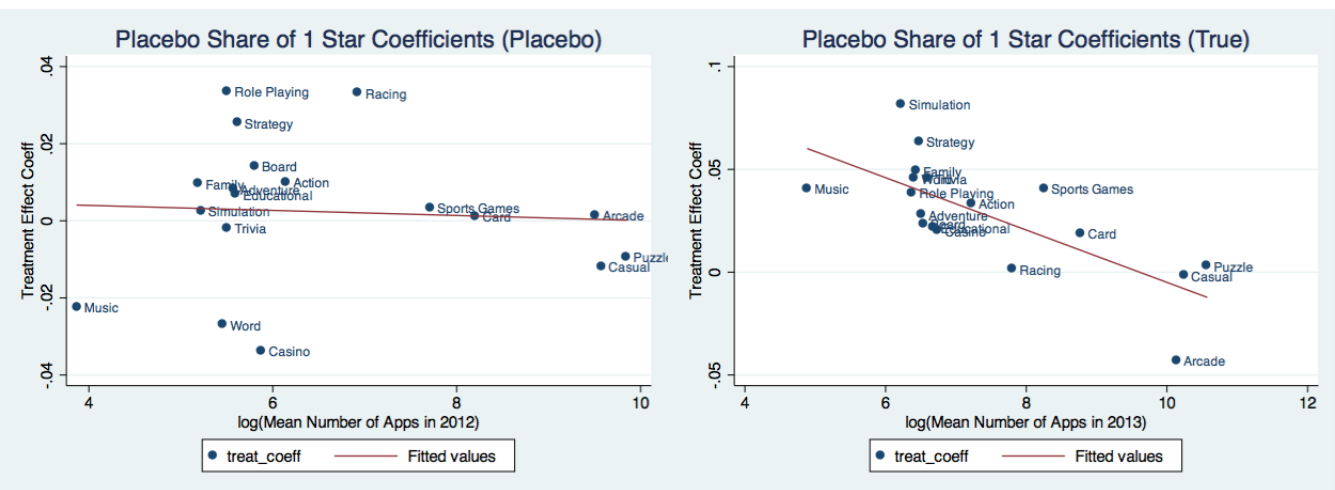
In the 1 star share regressions, the placebo category treatment coefficients have essentially a zero correlation with the size of the categories (in terms of the number of apps). This, together with their very small magnitude, suggests that the placebo treatment essentially had no effect on the share of 1 star reviews, as expected. By contrast, the magnitudes are much larger (often twice as big) in the "true" regression,

Table 20: **Placebo Test Regressions Results**

	Share of 1 Star Ratings		**Mean App Size (MB)**	
	(1)	(2)	(3)	(4)
	2012/3	2013/4	2012/3	2013/4
Placebo Treatment	0.002		2.077	
	(0.004)		(1.780)	
True Treatment		0.028***		-5.562***
		(0.003)		(1.022)
Time FE	YES	YES	YES	YES
Category FE	YES	YES	YES	YES
Observations	881	966	881	966
R-squared	0.350	0.468	0.602	0.646

Robust standard errors in parentheses
*** p<0.01, ** p<0.05, * p<0.1

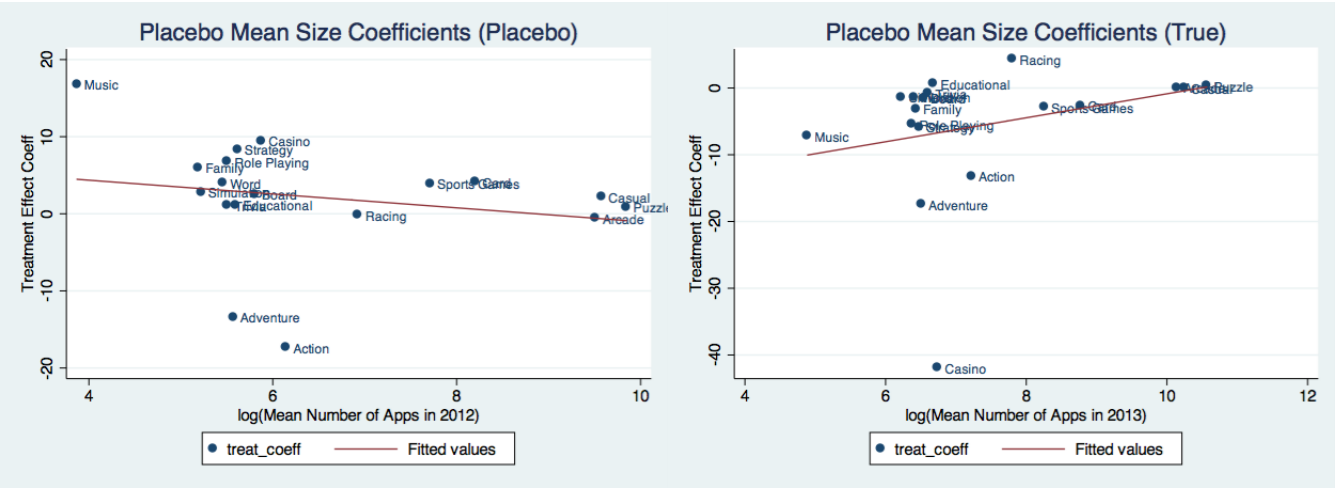
Figure 17



and there is a clearly negative correlation between category size and the treatment effects - as expected.

The mean size regressions are similar. In the placebo regressions, there is a negative relationship between category size and the placebo treatment effects, which

Figure 18



are mostly positive and not statistically significant. This suggests that in the pre-split period, the smallest categories experienced a larger increase in mean size (increase in quality) relative to the larger categories. This is consistent with the story that apps in the small categories (or types) had to be "better" to be successful. By contrast, the relationship between the treatment coefficients and category size is positive in the "true" period, and the coefficients themselves are mostly negative for the small categories. Again, this is consistent with the story of the split in categories either reducing competition or otherwise reducing the incentives of developers in small categories to produce "high quality" apps.

10 Appendix D

10.1 Additional Monthly Coefficient Estimates

Figure 19