Amateurs:
Development Costs, Market Structure and
Innovation on Digital Platforms*

Kevin Boudreau

January 12, 2017

Abstract

Digital products, developed on digital platforms, with digital tools and channels have become historically cheap to build. I exploit exogenous variation in costs faced by app developers to estimate causal effects of especially low development costs on market structure and products available. Initial cost declines behave as standard theory predicts, until finally causing highly nonlinear increases in numbers of lowest-quality developers. The massive influx of amateurs and associated low-quality products is nevertheless associated with an increase in absolute numbers of high-quality products. Patterns conform to a simple characterization of market selection and retention in an entrepreneurial marketplace.

Keywords: Amateurs, fixed costs, innovation, digital platforms
JEL codes: D4, E26, J4, L1, L8, 03

* Northeastern University. Dodge Hall, 360 Huntington Ave, Boston, MA 02115. kevin@neu.edu. I would like to thank executives at Priori Data for contributing data. The paper also benefitted from comments of seminar participants at Harvard University, New York University, the University of Southern California, and Warwick University. I wish to acknowledge funding and support provided by Harvard Business School and Google. All errors are my own.
1 Introduction

“We’ve learned about the internal combustion engine, we’ve learned about electrification, we’ve learned about how to build cities—and now we’re down to the Internet.” – Robert Solow (2014)

Online platform-based marketplaces are now well-established mainstays of the modern economy and include some of the largest and most iconic companies of our time (Hall, 2002). Online marketplaces selling physical “analog” goods, such as Amazon.com and Alibaba.com, are now well-understood to attract “long tails” of products and sellers by reducing search and transaction costs and perhaps offering a chance at achieving high growth and commercial success, at least for some fraction of sellers (e.g., Lee and Wu, 2009; Brynjolfsson, Hu, and Simester, 2011).

The long tail phenomenon extends too to more recent generations of online marketplaces on which traded goods and services are themselves digital, such as software, music, written works, podcasts, artwork, video production, images, and games on platforms such as the Apple AppStore, Youtube, Soundcloud, Twitch, Shutterstock, Valve and Amazon Kindle Self-Publishing.

However, in these cases of digital platforms we not only typically observe a long tail of companies, smaller merchants, developers-for-hire and entrepreneurs of various stripes; these market typically attract a great many amateurs—hobbyists, students, hackers, tinkerers, users and the like. This paper studies whether amateurs have a consequential impact on aggregate market structure and the basket of products that become available to consumers.

In many cases, such as Youtube or Soundcloud, platforms were initially launched with policies explicitly intended to attract amateur developers. Even in cases with explicit policies, amateurs have nonetheless appeared—and done so in large numbers. For example, roughly a year after releasing the iPhone, on June 29, 2007, Apple launched its AppStore on July 10, 2008 to distribute

\(^1\) Video disseminated by US White House of Robert Solow in association with his receipt of the Presidential Medal of Freedom Address (November 26, 2014), available at m.youtube.com/watch?feature=youtu.be&v=LNttoDiB5rc
third-party applications software for the iPhone, iPod and iPad. The software
development kit was downloaded more than 100,000 times within four days
of launch and more than one million times within the year. Fast Company
magazine heralded “...a revival of the hobbyist programmer. Not since the
days of the Commodore 64 and Atari 2600 has indie software been sold by such
tiny teams of programmers” (Stevens, 2011). Making these sorts of changes
possible, even “a high schooler with limited programming experience and a
laptop can build an app” and distribute it alongside those of professional firms
(Desai, 2015).

Despite a seeming “blurry line between professionals and amateurs” Chip
(2012), journalistic coverage describes amateurs as making up a large bulk
of supply. For example, Der Spiegel describes the AppStore, as “including a
lot of amateur developers, but also incorporates software companies” (Muller,
2009). Market researcher, VisionMobile, estimates that 24% of the hundreds
of thousands of developers of apps make zero revenues and 47% make less than
$100 per month. Therefore, rather than these just being amateurs in the sense
of being unpaid, these amateur developers are even willing to make negative
disbursements, paying annual platform access while devoting time, effort, and
investments in equipment to this activity.²

What leads amateurs to participate in some marketplaces in varying pro-
portions—and not others? How should we understand determinants of market
structure with amateurs along with more traditional firms? And, do amateurs
even matter much? How? On the face of it, answering these questions has
the potential to help explain development activity on platforms such as the
Appstore. New AppStore product releases averaged about 10 thousand per
month from initial launch in July 2008 until July 2010, and grew to more than
15 thousand per month the following year, and grew once more to 20 thousand

²Even probabilistic expected payoffs are near zero. The probability of transitioning from
the bottom half to among the top one percent is virtually zero, while entering into the
very top one percent only pays about $2MM in gross income, on average, in this best of
circumstances. (For example, if the entire $30BB of app revenues were paid out entirely
to the top 1 percent of apps, roughly 1.5 MM, this would imply the mean app in the top
1 percent would receive $2MM. With highly skewed outcomes, the 99th percentile is will
receive a great deal less than this.)
per month in 2012, and again by 2014 reached 45 thousand per month. So numerous were developers and apps that a great many similar products were supplied to quite narrow subcategories. For example, in 2013 data, six years after the launch of the Appstore, market analyst Priori Data counted 3,759 weather-related apps, 528 hockey-related apps, 1,643 fishing related apps, 166 zombie-themed game titles and 1,550 apps for playing Sudoku.

It is not immediately clear why a platform owner should necessarily want to grant access to large numbers of amateurs. Standard theory predicts that least productive developers at the bottom of the market should tend to rapidly fail and churn out (Jovanovic, 1982; Klepper, 1996; Manso, 2016). Even when operating in the market, least productive developers–only very distantly following market leaders–should have little or no reasonable expectation of commercial success and therefore should have similarly little economic incentive to make discretionary investments in quality-improving product development or to pursue particularly risky and innovative projects (e.g., Aghion, Bloom, Blundell, Griffith, and Howitt 2005; Anderson and Cabral, 2007). Moreover, the entry of large numbers of lowest quality suppliers risks crowding out incentives of professional firms if this produces noise, asymmetric information, competition for attention, and some degree of business stealing.

Of course, one possibility is just that what appears to be amateurs is just low-productivity entrepreneurs with overconfident, unrealistic expectations of commercial success (e.g., Landier and Thesmar, 2009). However the persistence and continuing growth of numbers of developers who have no reasonable chance of commercial success casts doubt on this as a complete explanation.

Another possibility is that nonpecuniary rewards should play some role in drawing unpaid amateurs (e.g., Raymond, 2001; Lakhani and Wolf, 2005). For example, Koch and Kerschbaum (2014) document that, apart from seeking income from operations, large numbers of developers–particularly individual developers–participate in the pursuit of intrinsic motivation, intellectual

\footnote{In comparison, the rate of new additions of entrepreneurs of the entire U.S. economy grew only an order of magnitude faster, at 530,000 new business owners each month according to the Kaufman Foundation.}
stimulation, learning, own use of the application, and building professional reputation. Market researcher Vision Mobile estimates, using sampled survey data, that 13% of developers are motivated simply as hobbyists. Nonetheless, the existence of nonpecuniary motivations per se is not an entirely satisfactory or certainly complete explanation. Compensating rewards from nonpecuniary motivations have been documented in a wide range of sectors without resulting in large numbers of amateurs and systematically negative returns (e.g., Hamilton, 2000; Stern, 2004).

Analysis in this paper proceeds by focusing first on a most basic feature of online digital platforms: the exceptionally low fixed development costs required to build products “on top” of them, as elaborated in Section 2. The guiding theoretical framework, in Section 3, analyzes low development costs in a model of selection and retention in an entrepreneurial marketplace. The main prediction is that incremental reductions in minimum fixed costs can produce discrete, nonlinear changes in market participation, granting access to amateurs. The framework also clarifies ways in which amateurs might plausibly improve products on offer, contrasting predictions of standard theory.

Section 4 describes the empirical context, app developers on the Apple Appstore, and the data set. The empirical strategy exploits exogenous variation across 503 precisely-defined product subcategories, using minimum file size (in megabytes) as a proxy measure for minimum required development costs in each subcategory.

Section 5 presents results, demonstrating that rather tiny incremental variation in the proxy measure of minimum development costs causes more than doubling of developer numbers—almost entirely made up of very smallest developers, producing a flood of very lowest quality products. (These strong causal effects with minimum file size contrast with zero correlation with mean file sizes, despite mean values having orders of magnitude larger variation.) This radical transformation of market structure and associated descriptive facts is consistent with predictions of the analytical framework, and inconsistent with a range of alternative explanations.

As should be expected, the large influx of discretely lower quality ama-
teur developers coincides with a growing weight of the product distribution towards lower (and zero) prices, and nonlinear increase in numbers of product varieties. Most notable is the finding that the large influx of amateur developers and associated bulk of low quality products is associated with greater absolute numbers (although lower proportions) of highest quality products. The result is consistent across multiple measures and an exhaustive series of cuts of the data. These patterns are inconsistent with standard theories of economic incentives, and consistent with explanations anticipated in the analytical framework. In Section 6, I conclude.

2 Digital Platform-Based Marketplaces & Development Costs

Online marketplaces go as far back as the early 1970s when MIT and Stanford students sold goods to one another on the early ARPANET (Markoff, 2005). From these early beginnings, eventually public and private initiatives such as France’s government-sponsored Minitel system and Compuserve’s subscription-based “Electronic Mall” in the 1980s led to wider-spread use of electronic trade and intermediation on platforms. The mainstream acceptance of the world wide web and the National Science Foundation’s decision to lift its prohibition on commercial enterprise on the Internet finally propelled the creation of many more online marketplaces from the late 1990s, leading to many of today’s giants such as Alibaba, Amazon, and eBay.

More recently, a great many goods and services traded on online platforms are often themselves digital, ex: software programs (e.g., Apple App-Store), artwork (e.g., Curioos), studio production (Tongan), recorded media (Youtube), real-time performance streams (e.g., Twitch), images (e.g., Shutterstock), games (e.g., Valve), game development (Unity), and cloud-based programming microservices (e.g., MashApe).

As is well known, digital developers enjoy virtually zero variable costs of production, replication and distribution (e.g., Varian, Farrell and Shapiro,
What is particularly notable in cases of development on digital platforms relates to fixed development costs. Costs of developing new products and experimenting in a marketplace have fallen to historically low levels in cases of digital products, developed on digital platforms, using digital tools, and distributed via digital channels.

There are several underlying structural reasons for lower development costs of products built and commercialized “on top” of online digital platform-based marketplaces. To begin, marketing and commercialization costs can largely be erased or at least radically reduced where developers avail themselves to a platform’s transaction and institutional infrastructure (Levin, 2011) and ready access to buyers and distribution (Rochet and Tirole, 2003; Rysman, 2009).

The development task is itself radically simplified to one of reconfiguring existing general platform technologies (e.g., Varian, 2010; Bresnahan and Greenstein, 2014). Development tools, documentation and sample designs—themselves digital and easily distributed—further streamline this process.\(^4\)

The very nature of online digital products itself also eases creation of “minimum viable products” that can be trialed and later reconfigured, revised or even replaced with user updates upon receiving market feedback.

It can also be in the interest of a platform market designer to avoid levying charges (and perhaps even to provide subsidies) to developers working on their marketplaces (Rochet and Tirole, 2003). Collectively, these economies then allow projects to be self-funded and bootstrapped, allowing external financing costs to be circumvented altogether.

These structural factors contributing to especially low minimum costs of development are clear, for example, in the case of the Apple AppStore, an iconic modern online digital platform-based marketplace. Apple CEO Steve Jobs referred to the AppStore as "the best deal going" for software developers. It costs a developer $99 per year to register and place one app on the platform.

\(^4\)For example, developing an application software program (“app”) requires developing machine-readable instructions to direct an operating system platform to trigger activities across a computer’s subsystems, and perhaps a wider network of machines. Analogously, a digital media file or song is just another “app” in the sense of providing instructions for media player platforms to follow.
Beyond these charges, development requires a Mac computer, familiarity with Objective-C (or one of several other, often simpler compatible programming frameworks).

Market research firm Statista\(^5\) reports the average cost of developing an app for the Apple AppStore to be $27,463, in terms of labor costs or costs of outsourcing; but actual costs may be considerably higher or lower. Development budgets of leading developers are orders of magnitude higher. Investments made by app developers will also vary according to the nature of the software being developed and the level of quality and sophistication, the use of licensed content, the extent to which a developer wishes to engage in independent marketing and distribution efforts. At the same time, minimum development costs are considerably lower. For example, smaller apps are routinely outsourced for several thousand dollars. And, developing one’s own minimum viable app that will be successfully approved and certified by AppStore officials might take as little of weeks or days development effort. For example, the first version of “Yo,” a messaging app which would go on to raise $50MM of funding, was described by its founders to have taken just eight hours to program.

Upon developing an app, paying for AppStore access and having the app certified to meet minimum quality standards by Apple personnel, developers retail their apps on the AppStore. Developers set the retail price and pay to 30 percent of variable revenues to Apple. For this, they pay no credit card, marketing, or hosting fees.

3 Development Costs and Entrepreneurial Market Structure

Effects of fixed costs in markets have long been studied in research on market entry and selection into entrepreneurship. Reducing fixed costs should “lower the bar” to greater numbers of progressively lower quality marginal entrants,

\(^5\)Downloaded in December 2016 from http://www.statista.com/topics/1694/app-developers/
as lower costs reduce the corresponding expected income needed to recover these costs (e.g., Spence, 1976). Where lower fixed costs are associated with low exit costs, the bar is lowered farther from the option value of returning to an outside option if a project is unsuccessful (Manso, 2016).

The “bar,” or minimum quality threshold, is lowered farther still where motivated entrepreneurs are willing to accept lower returns for a compensating reward of “being one’s own boss” (Hamilton, 2000), or other sorts of nonpecuniary rewards and other sorts of deviations from standard risk-neutral preferences for market payoffs (Åstebro, Herz, Nanda, and Weber, 2014). Lowering the bar can also reshape market retention, as when low quality developers—i.e., those with low ex ante expectations of commercial success—turn out to be more successful than expected, ex post (e.g., Aguiar and Waldfogel, 2016).

It is useful to first integrate these arguments implying that lower costs (and higher nonpecuniary motivations) should each simply “lower the bar” to progressively larger numbers of lower quality marginal entrants—before proceeding to effects of especially low costs. Consider a platform marketplace that occurs over many periods. In each period, a new unit mass of developers joins the economy and chooses whether to join the platform or not. For simplicity, assume each developer lives two periods or “phases”. In a first phase, the developer chooses whether to join; if joining, then in the second phase the developer chooses whether to persist and continue to operate. The first phase reflects market selection; the second phase reflects market retention. Total number and types of market participants is the sum of those in the first and second phases.

To select onto the platform in the first phase, a developer’s expected payoffs over both phases must exceed opportunity costs. Let the probability of project success be denoted \( p \). Developers can only learn whether the project is successful by operating (experimenting) in the first phase. If successful, developer \( i \) are heterogeneous in their expectations of gross income flows, \( R_{i,1} \) and \( R_{i,2} \), over the two phases. With probability \( 1 - p \), the project is unsuccessful.

\[ ^6 \text{This follows, say, the completion of academic training, leaving other areas of employment, etc.} \]
yielding zero gross income. I assume zero discounting, for simplicity.

Developers are also heterogeneous in levels of non-pecuniary payoffs they derive, \( \beta_i \geq 0 \), from participating in the market (e.g., Hamilton, 2000; Astebro, et al., 2014). For example, it is well-documented that online developers often intrinsically enjoy the work, acquire new skills or reputation, or directly consume outputs (e.g., Raymond 2001; Lakhani and Wolf; Lerner, Pathak, and Tirole, 2006; Zhang and Zhu, 2011). (More generally, an additive \( \beta \) term in payoffs, captures any form of deviation from standard risk-neutral market income preferences and payoffs that are external to usual income from the market.\(^7\))

Opportunity costs for participating on the platform in each period are, \( W_{i,1} \) and \( W_{i,2}. \)^8 A developer choosing not to participate receives these opportunity costs instead as payoffs. Variable costs are zero. Therefore, the condition for joining the platform in the first phase is simply that the expected payoff stream exceed the expected opportunity costs:

\[
p (R_{i,1} + R_{i,2} + 2\beta_i) + (1-p) (\beta_i + W_{i,2}) - (W_{i,1} + W_{i,2}) > 0. 
\]

(Note: Where \( R \)’s and \( W \)’s are invariant and \( \beta_i = 0 \), expression (1) simplifies to \( p R_i > \frac{(1+p)}{2} W \), which is identical to Manso’s (2015) analysis.) The above expression implies that the marginal entrant will return to the outside option if unsuccessful in the first period; and, if successful in the first period, will continue operating in the second period, i.e., \( R_{i,2} + \beta_i - W_{i,2} > 0 \).

Developers choose their levels of fixed opportunity costs, \( W_{i,t} \). Let total developer cost, \( W_{i,t} \), be some minimal viable level of investment \( w_{min} \), plus any discretionary investment beyond this minimum level, \( w_{i,t} \), or \( W_{i,t} = w_{min} + w_{i,t} \). Greater investment increases expected income, conditional on a project being successful, \( R_{i,t} \equiv \rho_i w_{i,t}^\theta \), where \( \theta \epsilon (0,1] \) defines concavity in this relationship. The parameter \( \rho_i \geq 0 \) summarizes heterogeneity in \( ex \ ante \) income expecta-

\(^7\)This definition includes external payoffs in the form of payoffs from complementary markets.

\(^8\)This includes say foregoing salaried employment, leisure, or the pursuit of alternative projects. This opportunity cost also includes any material investments of time, effort, and capital.
tions, or commercial “quality.” Fixed investments choices are chosen to maximize per-period profits\(^9\) implying, \(w_{t,1}^* = (\rho \theta)^{\frac{1}{1-\varphi}}\) and \(w_{t,2}^* = (\rho \theta)^{\frac{1}{1-\varphi}}\).

### 3.1 Selection and Retention with Decreasing Costs: Empirical Predictions

Substituting the earlier expressions into earlier expression (1) implies the following minimum quality threshold in the first phase:

\[
\rho \geq \left( \frac{(1 + p)}{p + p^\frac{1}{1-\varphi}} \left( \theta^\frac{\varphi}{1-\varphi} - \theta^\frac{1}{1-\varphi} \right) \right)^{1-\varphi} (w_{\text{min}} - \beta_i) \tag{2}
\]

This minimum quality threshold is illustrated in Figure I and summarizes key insights of past theory. Developers without non-pecuniary motivations, \(\beta = 0\) (to the left of the \(\beta > 0\) line), must meet some minimum quality threshold (shown as a horizontal line). Developers with non-pecuniary pay-offs, \(\beta > 0\) (to the right of the \(\beta > 0\) line), are “motivated” entrepreneurs, whose minimum thresholds become progressively lower with higher compensating non-pecuniary rewards. (Where \(\beta > 0\), Figure I presumes a uniform distribution of \(\beta\), and an arbitrary fraction of developers for whom \(\beta = 0.\))

As regards market retention, fraction \(p\) of developers joining the platform are successful and will therefore persist in the second phase. The observe market structure will be the sum of developers in both phases.

<FIGURE I>

Expression 2 thus summarizes the conventionally understood effects of reducing development costs, described earlier in this section, that reductions

---

\(^9\)Optimizing per-period investments in this manner is consistent with needing to largely re-build the application in the second period if it is found to be successful and requires “scaling-up.”

\(^{10}\)The particular distribution of developers entering into the economy in a given period is, more generally, an empirical question. The distributional assumption here simply summarizes a case where there are some developers who do not care about non-pecuniary rewards, and others who do—where those who do are heterogeneous.
of bare minimum required fixed development costs, $\omega_{\text{min}}$, “lower the bar” to greater numbers of progressively lower quality developers.

However—crucial to the question of effects of especially low costs and not considered in past research on market selection and entry—inequality 2 is also true if the following condition holds:

$$w_{\text{min}} \leq \beta_i.$$  \hspace{1cm} (3)

Where inequality (3.1) holds, the right-hand-side of the entry condition (2) is always negative. The right-hand-side is therefore unconditionally less than the left-hand-side, which is necessarily positive by definition ($\rho \geq 0$).

Therefore, where costs—and particularly the bare minimum fixed development costs $w_{\text{min}}$ to develop a minimum viable product—fall lower than non-pecuniary rewards the consequence is that selection condition (2) no longer depends on attaining a minimum quality $\rho$ threshold.

Therefore, whereas initially minimum development cost declines “lowering the bar”, with progressive incremental cost declines—at some point—the “bottom falls out” of the market, in the sense of the minimum quality threshold disappearing (for at least a subset of developers), as illustrated by Figure II.

<FIGURE II>

Implications of the bottom falling out are even greater in the second phase of retention. Whereas under classical cost conditions, fraction $p$ developers are successful and are retained, with costs sufficiently low for the bottom to fall out, all developers meeting condition (3.1) persist in the marketplace.

This implies that this subgroup conforming to inequality (3.1) will be behaviorally distinct from professional companies or entrepreneurs (despite coming from the very same continuum of potential entrants). First, this group is willing to “pay to participate” in the marketplace in the sense of making negative disbursements, as is typical of say hobbyist amateurs. This differs from cases examined in previous literature in which economic actors simply accept
lower than competitive wages (e.g., Hamilton, 2000; Stern, 2004). These conditions also depart from cases of say developers turning out to have negative returns ex post, but ex ante formulating a positive expectation of returns (e.g., Rosen, 1981; Lee and Wu, 2009).

As a direct consequence, this group is not subject to usual market discipline; therefore, the bottom of the market in these cases can accumulate in size in relation to even larger and more productive professional firms. By contrast, standard theory instead predicts least productive competitors at the bottom of the market will tend to dwindle in importance and share of supply with age and time (Jovanovic, 1982; Klepper, 1996; Manso, 2016).

However, while amateurs will be unbounded from below in quality, as the bottom falls out, they will be bounded from above in quality. Even among developers with sufficiently high-\(\beta\), to right in Figure II, and meeting condition (3.1) will be those whose (endogenous) choice of development costs makes them sensitive to market outcomes. Thus, those with sufficiently high-\(\rho\) will choose investments levels (expression 3.1) exceeding non-pecuniary payoffs, \(W_{i,1} = (p\rho, \theta)^{\frac{1}{1-\beta}} + w_{\text{min}} > \beta_i\). As a consequence, these developers will be outwardly indistinguishable from traditional “motivated” entrepreneurs who must still pursue at least some level of income in order to sustain operations (Hamilton, 2000). In the first phase, the upper frontier (3.1) is therefore defined as follows:

\[
\rho_i < \frac{(\beta_i - w_{\text{min}})^{1-\theta}}{p\theta}.
\]

This concave frontier separating crowds from more traditional companies and entrepreneurs was traced in earlier Figure II.

Thus, the main empirical prediction is incremental reductions in (bare minimum required) development costs should begin with incremental increases in progressively lower quality marginal developers (“lowering the bar”); however, at the point at which minimum fixed costs drop to levels conforming to expression (3.1) for some developers, there will be an abrupt nonlinear increase
in overall developer numbers, from an influx of amateurs. These predictions are summarized in the following figure.

<FIGURE III>

It should be emphasized too that the number and distribution of developers observed in Figure III is the sum of developers selecting onto the platform in phase one, and those retained by the platform in the second phase. Within this simple framework, among professional companies who join in the first phase, only fraction $p$ are successful and persist in the second phase; fraction $1-p$ are unsuccessful. However, within this simple characterization an equal fraction $p$ of amateur developers are themselves successful in generating non-zero revenues and can potentially add to numbers of companies and crowds where they also exceed condition (3.1).

Therefore, while there are a range of arguments (as highlighted in the introduction) why a large influx of low-quality amateurs might have little or possibly negative effects on high quality products available in a marketplace, the model mechanics illustrate the possibility that some fraction of successful amateurs can transition to professional status, adding to numbers of high quality products that would have otherwise existed. The characterization also highlights that amateurs will differ in being dominated by different sources of (nonpecuniary) motivations, while also possibly enjoyed unusual long life, not being subject to usual market discipline. Each and any of these factors has the potential to alter product development patterns.

4 Empirical Context and Data

The remainder of this paper is devoted to empirically investigating whether incremental cost declines do in fact produce discrete, nonlinear changes in market participants—as predicted above—and to example implications for the basket of products that become available.

Data for this study comes from the Apple AppStore, a leading digital platform-based marketplace. A number of past research studies study data
from the Appstore. Thus far, most have focused on the demand side of the market. Carare (2012) finds that apps that make it on the top seller list (top 100 ranked) experience a discontinuous boost in demand. Garg and Telang (2012) use data on rank order in downloads, rank order in price, price charged and knowledge of whether an app offers in-app purchases or not to implement a method to estimate otherwise observed market shares of apps. (This method is exploited in this study.) Ghose and Han (2014) estimate a structural model of demand, using demand estimates to show that in-app purchases are associated with higher demand and in-app advertising is associated with lower app demand. Bresnahan, Orisini and Yin (2014), find that with demand for apps highly concentrated among most popular apps and these popular apps multihoming across competitor platforms (particularly Google’s Android), multiple platforms can coexist without market tipping. In a paper that analyzes the supply-side of the market, Yin, Davis and Muzyrya (2014) find that “hit” games are generated with higher probability by developers who release large numbers of new titles rather than iterating multiple versions of the same title; however among non-games, “hit” titles are more likely to be generated by those who release fewer titles, but who iterate on multiple versions. Bresnahan and Greenstein (2014) and Bresnahan, Davis, and Yin (2015) provide excellent review of a number of salient issues in this marketplace and how they may map to prior research on the economics of platforms, competition, firm tactics and technical change.

4.1 Data Set

The data set analyzed here is constructed from matched data from multiple sources and reflect the cross-section of 192,372 app developers supplying 693,541 apps from the mid of 2013. This comes well after launch—five years later—during a period of continuing steady growth. It comes before a change in the algorithms used to rank apps (which is relevant in being able to approximate revenues, as described below).

The first data source is the AppStore, itself, where all apps and app devel-
opers appear. Data were machine-collected with web-based crawlers to retrieve app title, developer name, version number and file size. These data include app name, version number, title, average user quality rating, number of ratings, file size.

The Apple AppStore also has developers to categorize their products in 32 app categories. However, this aggregates considerable aggregation of types. To achieve precise and most meaningful definitions of subcategories, I instead use category definitions created by Market analytics company Priori Data. Using machine learning algorithms and expert human judgement, 43 categories with 503 subcategories\textsuperscript{11} were defined by the company for all apps on the AppStore.

Priori Data also provided revenue share estimates for each app. Their estimates are based on the procedure developed by Garg and Telang (2012) that uses prices, download ranks and revenues ranks for each app to estimate the revenue share distribution (including app sales and in-app purchases). This procedure is combined with propriety data sets and research by the company to augment and better calibrate results.

5 Analysis & Results

The arguments of Section 3 predict that incremental cost reduction will–once low enough–produce nonlinear increases in numbers of developers of lowest quality. In this section, I confirm an abrupt nonlinear causal response in market participation (numbers of developers) to incremental changes in minimum fixed costs. (In the following section, I will confirm that products added in association with nonlinear change are of lowest quality.)

\textsuperscript{11}The breakdown of categories is proprietary. However, to provide some indication, major categories include such things as “chat and sms”, medical, children’s games, and board games. Subcategories include such things as wedding planners, voice recorders, vocabulary games, travel guides, timers, text messagers, and golf games.
5.1 Baseline Analysis and Causal Interpretation

A key assumption underpinning the empirical strategy is that variation across app subcategories in minimum development costs are exogenously determined and a basis for deriving causal inferences. The basic rationale here is that while many factors shape investment levels, generally speaking, bare minimum investments must adhere to basic features required by the product definition (and associated challenge and complexity) of associated development tasks, while also meeting bare minimum quality requirements for Apple certification. Before proceeding further the analysis, this basic assumption is more deeply investigated, using a simplest linear specification:

\[ \text{NumDevelopers}_i = \beta_0 + \beta_1 \text{MinFileSize}_i + \text{ControlVars}_i + \epsilon_i, \]

where \(i\) indexes the 503 distinct subcategories. The dependent variable, NumDevelopers, is a count of numbers of developers in each subcategory. The regressor of greatest interest is the proxy measure of minimum development costs, MinFileSize—the smallest file memory size of any app appearing in a subcategory, measured in megabytes (MB).

Table I reports OLS regression results. Standard errors are clustered by category. Model (1) reports results of a simple regression of NumDevelopers on MinFileSize and a constant. The estimated coefficient on MinFileSize is large and negative, indicating an average reduction of 3,861 developers per megabyte increase in minimum file size. This implies an average increase of 144 developers per reduction in each standard deviation increase in MinFileSize, or just 0.037 MB. This is 19.8% of the 728 developers, on average, in each subcategory.

A challenge in interpreting this relationship is the possibility spurious correlation with unobservable factors influencing both NumDevelopers and MinFileSize. As a coarse test to detect any such effect, model (2) adds a dummy variable for games-related subcategories. Differences between games and non-games might relate to any number of unobserved factors, from mar-
ket potential, to developers’ motivations, to ease of development, to size of
developer pool, and so on. There are, on average, 280.5 more developers in
games subcategories than in other categories. Adding this dummy has the ef-
fact of slightly increasing the magnitude of the earlier estimated coefficient on
$MinFileSize$ to 4,241 developers per MB (i.e., 157 developers per standard
devation change); but, the is not statistically significant. Taking this idea a
step further, model (3) adds dummy variables for each of the 43 main app
categories, thus re-estimating the model on the basis of within-category dif-
ferences alone. Again, there is no statistical change in the estimated coefficient
on $MinFileSize$.

$\begin{array}{|c|}
\hline
\text{TABLE I} \\
\hline
\end{array}$

Given the particular importance that market potential might play in simul-
taneously influencing both chosen investment levels and entry choices, model
(4) explicitly adds total revenues in the category, $Revenues$, as a regressor.
(Some caution needs to be taken here in interpreting results. Finding a lack of
change in estimates is more informative than the finding of any change, given
that realized revenues—distinct from underlying market potential—are endoge-
rous.) As should be expected, the relationship between number developers
and $Revenues$ is highly positively related to $NumDevelopers$. More crucially,
the estimated coefficient on $MinFileSize$ is statistically unchanged.

Another closely-related challenge in interpreting regressions is the possi-
bility of reverse causation. For example, greater developer numbers might
plausibly intensify competition and “crowd out” investment incentives. If the
observed minimum investment level did not reflect a natural limit, but instead
were responsive to incentives, we might then see large numbers of developers
cause smaller file size. However, if this were the case, we might then expect
small apps to be shaped in similar ways. However, replacing the minimum file
size with the first percentile file size produces a considerably smaller magni-
tude coefficient of opposite sign (0.74, s.e. = .32). This opposite (positive) sign
is consistent with endogeneity for measures of costs and investments greater
than the minimum.
The importance of bare minimum levels is underlined when comparing to the lack of correlation with average levels. Mean file sizes in subcategories are orders of magnitude larger than minimum file size, with mean file sizes varying 3,324 times more, in terms of relative size of standard deviation. And yet, while NumDevelopers is so very highly related to minimum file size, there is no statistical relationship with mean file size (-960.0, s.e.= 938.8), in a regression with category effects (and whether or not including the minimum file size at the same time).\textsuperscript{12}

Therefore, apart from the \textit{a priori} rationale, minimum file size appears to a meaningful proxy for minimum file size, with results of these tests indicating variation in this variable is exogenous and orthogonal to other determinants of numbers of developers participating.

5.2 Development Costs and Market Participation

To estimate the relationship between between NumDevelopers and MinFileSize in a flexible manner, I use a nonparametric kernel technique. I use a second-order Epanechnikov kernel. The bandwidth is chosen according to "direct rule-of-thumb" local-linear method of Ruppert, Sheather and Wand (1995). The local linear estimates are essentially the weighted least-squares estimates calculated along the curve, where the weights are provided by the kernel.

As presented in Figure IV, cost reductions initially lead to incremental increases in NumDevelopers, proceeding from right to left in the figure. However, consistent with predictions, as minimum cost reductions proceed from already low levels to especially low levels, an highly nonlinear, even “kinked,” increase in developer numbers occurs. This shape of curve is notably similar to the theoretically predicted curve shown earlier in Figure III.

To better estimate the location of the “kink”, I estimate a two-part piece-wise linear model of the relationship between NumDevelopers and MinFileSize. The piece-wise specification allows the relationship to be modeled as two separate and independent linear models that are divided at some breakpoint level

\textsuperscript{12}Moreover, including various percentile statistics of file size and the mean value does not statistically change the estimated coefficient on MinFileSize.
of MinFileSize that is itself a parameter to be estimated (and placing no further constraints on the model, including that the two curves necessarily even intersect). Linear parameters, constants and breakpoint are estimated by maximum likelihood. The specification is therefore,

\[
\text{NumDevelopers}_i = \begin{cases} 
\beta_{0,\text{low}} + \beta_{1,\text{low}} \times \text{MinFileSize}_i + \epsilon_{i,\text{low}} & \text{if } \text{MinFileSize} \leq \delta \\
\beta_{0,\text{high}} + \beta_{1,\text{high}} \times \text{MinFileSize}_i + \epsilon_{i,\text{high}} & \text{if } \text{MinFileSize} > \delta 
\end{cases}
\]

\text{(5)}

Where \( \delta \) is the parameterized breakpoint to be estimated along with other model parameters by maximum likelihood.

As shown graphically in Figure IV, the piece-wise estimator shows similar incremental reductions to MinFileSize to the breakpoint of 0.063 MB. Important to note, despite the unrestricted estimation of both pieces of the model, the two linear models meet and are statistically equivalent at the breakpoint, consistent with a sudden change of slope at this point. The slope between NumDevelopers and MinFileSize on the right hand side segment segment of the piecewise model is negative 159 developers per MB of change in the minimum file size; at points below 0.063 MB, the slow rather dramatically changes to -21,073 developers per MB change in the minimum file size. This amounts to their being roughly two-and-a-half times greater number of developers in subcategories below the breakpoint. The mean number of developers in categories with MinFileSize greater than 0.063 is 362 developers (s.d. = 308) and this balloons to 856 (s.d. = 789) for MinFileSize less than 0.063.

<FIGURE IV>

As is typical of today’s observational data on online digital platforms, we are not able here to observe types of developers. However, analysis of the associated influx of products will confirm the discretely lower quality of added products with MinFileSize less than 0.063
5.3 Development Costs and the Basket of Products

This section documents how this nonlinear variation in numbers of developers above and below the breakpoint of MinFileSize equal to 0.063 MB relates to characteristics of products that become available, in terms of variety, price and quality.

Number of Product Varieties. As might be expected, where there are especially low bare minimum required development costs and hundreds of added developers, there are even greater numbers of added products. As reported in model (1) in Table II, there are 845.7 more products in each subcategory with low minimum development costs, or an average total of 1404 titles in contexts where MinFileSize < 0.063 (i.e., where indicator LowMinCost in Table II is switched to one). The estimate is invariant to including category dummies, where estimates are based only on within-category variation across subcategories, as in Model (2).

A simple regression of numbers of product varieties on numbers of developers, as in model (3), indicates there are 1.8 titles per marginal variation in developer counts. This estimate, however, is the average marginal effect across the entire population— including variation in high cost subcategories and variation not necessarily related to changes in minimum development costs (such as market potential).\textsuperscript{13} To better isolate product varieties added per added numbers of developers caused by changes in minimum fixed costs, I re-estimate the model, exploiting LowMinCost as an instrumental variable in model (4). The model also includes category effects for added controls.

The point estimate of 1.7 added product varieties per added developer under low minimum fixed development costs is notably especially high. First, it is remarkably similar to the simple cross-population average, where this earlier estimate of 1.8 for the entire population was already likely upwardly influenced by spurious factors. Second, standard theory emphasizes that the large influx of lowest quality developers under low costs (as later analysis confirms) should

\textsuperscript{13}The estimated constant in model (3) is negative, as the relationship between numbers of apps and numbers of developers is convex. Adding a
have little incentive to make discretionary investments in product development, given little chance of recuperating these costs (e.g., Aghion, et al. 2005; Anderson and Cabral, 2007). This suggestion of surprisingly high levels of effort and investment by low quality “crowd” developers raises questions for future research.14

**TABLE II**

**Prices.** Table III reports regression results for regressions of prices on LowMinCost, the indicator that is switched on for MinFileSize less than 0.063. Model (1) regresses price on this indicator, along with category effects (results change without including these controls). The estimated coefficient indicates that, on average, apps are about 20 cents (0.19, s.e. = 10) cheaper in subcategories with especially low minimum fixed costs. (NB. It is important to emphasize that this coefficient of negative 20 cents should be interpreted as reflecting changes in structure of supply rather than changes in prices paid by consumers. The effect of adding hundreds of developers to subcategories which would already have hundreds of developers and many zero priced products is likely a great deal smaller.)

To provide some indication of whether the 20 cent decline in the distribution of price across supply reflects lower costs per se, or changing composition of developers and apps, I re-estimate the model, including a continuous proxy measure of minimum development costs, MinFileSize (not reported). The statistically zero coefficient on this continuous measure suggests the change in

---

14 Although in contrast to standard theory, this outcome is plausibly explained by any number of factors highlighted in the theoretical analysis in Section 3. Most straightforwardly, for example, lower development costs might imply lower costs of generating investing in scope. In addition, unexpectedly higher investment incentives—sufficient to increase average scope choices well above one—might derive from non-pecuniary motivations. (Non-pecuniary motivations were approximated in Section 3 as an additive constant; however it remains possible that non-pecuniary rewards could respond to variation in effort.) The crowd of amateurs at the bottom of the market might also simply be longer-lived and less subject to market discipline, providing a greater time horizon over which multiple product might be released. Further, it will be seen below that—as predicted in Section 3—the bottom falling out of the market also leads to absolutely larger numbers of successful developers, who might have greater resources and inclination to invest in wide scope.
price is mostly related to the large shift in supply structure under especially low costs, rather than changes in costs per se.

In the earlier regression results, each product observation is weighted equally. This has the effect of over-weighting those categories with many more products. To restore equal weighting to observations from each subcategory, I re-estimate the model with observations inversely weighted according to numbers of product observations in each subcategory, as in model (2). This does not much change the point estimate.

Part of the drop in prices is certainly related to composition of free versus paid apps, as 57.2 percent of products are zero price where there are relatively low development costs ($MinFileSize < 0.63$) versus just 52.2 percent where there are relatively high development costs ($MinFileSize < 0.63$). To estimate, however, whether non-zero prices are also different, I re-estimate the model using only data on non-zero priced products, as in model (3). The point estimate is negative and not different from earlier estimates.

<TABLE III>

### Product Quality.

Among characteristics of the basket of products studied here, most important are measures of quality. First, lowering the bar might have greater scope for impacting quality. (In relation to the earlier studied characteristics of variety and price—there are already hundreds of varieties and many suppliers or zero price or near-zero priced apps even before the bottom falls out.) Analysis of quality here using might also be particularly informative given data using narrowly defined subcategories. Second, product quality here will be our most direct means within the available data to assess whether, as predicted, the large influx of developers under low costs are indeed of especially low quality.

Several measures of quality are available in the data: user ratings, number of user ratings, size of application, and versions. Each is an imperfect signal of quality, but patterns in relation to exogenous shifts in minimum development costs studied here are broadly similar. Each measure indicates patterns consistent with the bottom falling out of the market, with incrementally lower
minimum development costs leading to a large mass—hundreds—of added lowest quality products in each subcategory—dramatically increasing the overall mass at the bottom of the market. At the same, there is evidence that the bottom falling out also increases absolute numbers of high and highest quality apps.

Key patterns are readily observable even in descriptive plots of the data. For example, Panel I of Figure V plots the distribution of user ratings of products (averaged across subcategories), stratified by those with relatively low or relatively high minimum development costs. Among the 845 products that are added under relatively low minimum development costs, there are a staggering 530 (63 percent of all added products) that receive zero ratings. (The count of 530 relates to the exposed portion of the far left pink bar in Panel I of Figure V, whose total height is 770—as bars for low and high cost subcategories are overlapping, not stacked.).

Consistent with such a large influx of much lower quality products, the chances of a product being of high quality drops under relatively low development costs. For example, comparing the relative densities (proportions) of products in low versus high quality as a ratio (the line joining the small x’s in Panel I of Figure V), this ratio descends well below one (on the right axis, logged), at ratings of two or higher. Crucially, however, despite the lower probability of high quality under low costs, the absolute number of higher rated apps under low costs is higher. This can be seen as the exposed pink

\[15\]

\[16\]

\[<\text{FIGURE V}>\]

\[15\]Affirming this skew to low quality in user ratings and the interpretation of unrated products as lower quality, the relative number of products available under low costs is highest for lowest rated products, even among the subset of products that are given at least one rating. The ratio of relative number of products is shown descending from left to right in the diagram (i.e., the dashed line connected by small circles).

\[16\]Note too that these comparisons are on the basis of total numbers of apps. If we approximate the “added” influx of apps as differences with a downward shift in minimum fixed costs, the ratio of densities and relative probability of attaining high quality is even lower. For example, the share of products with ratings of at least 3.5 is 43 percent among all products under high costs, 31 percent under low minimum costs, and just 23 percent among the “marginal” numbers who are added under low of 845 developers who join with lower costs.
area of low cost bars sticking out above high cost bars in Panel I of Figure V. Accordingly, the ratio of absolute number of products, shown as the line joining circles, is above one.

The other measures of quality in Panel II, III and IV in Figure V reveal consistent patterns. For example, in cases of numbers of ratings (Panel II), file size (Panel III) and version number (Panel IV), the far leftward bar of lowest quality products represents the vast bulk of all products added with low development costs. For example, in the case of file size in Panel III, the single greatest mass of developers is concentrated on very smallest apps, consistent with the bottom falling out of the market and disproportionately adding large numbers of low quality developers and apps. Of the 1404 apps overall in each subcategory, on average, in lower cost subcategories, over a thousand are in the very smallest category in the graph—less than 2 MB.\(^{17}\) Similarly, around a thousand products per category are in lowest quality category in other panels.

Again, the probability of attaining high levels of quality are considerably lower with low minimum development costs, for numbers of ratings (Panel II) and file size (Panel III) (as shown with the relative density line less than one). And again, absolute numbers of high quality products are higher with low minimum development costs (as shown with the relative absolute numbers line greater than one). An exception is the case of version numbers, which offer a possibly still more optimistic picture of product development under low minimum development costs, in Panel IV. In this case, not only is the ratio of absolute numbers of products greater than one, but so is the ratio of densities or probabilities of higher versioning.\(^{18}\) Broadly speaking, each of these patterns is consistent with a massive influx of low quality products (and

---

\(^{17}\)By comparison, the most basic free version of Angry Birds a popular and rather simple game by Rovio was about 30MB; the Star Wars version of Angry Birds was 140 MB. More intricate apps are considerably larger.

\(^{18}\)This could plausibly relate to any number of mechanisms, including the possibility that versioning is lower cost in cases of lower minimum development costs, or possibly that each “inventive step” with new versions is smaller. This pattern is also consistent with earlier evidence and discussion related to product scope choices (Section 5.3), where the nonlinear addition of low quality developers have higher product development incentives than would be predicted by standard theory.
developers) as the bottom falls out with relatively low minimum development costs; but adding much larger numbers is also associated with greater numbers of high quality products.

To provide more systematic and statistical assessment of whether indeed the “bottom falling out of the market” leads to greater numbers of high quality products, along with the large influx of low quality, I proceed to test differences within a regression framework. The most direct and relevant econometric comparison\(^{19}\) is simply a comparison of the absolute counts of numbers of products across the distribution, where I focus here on very lowest and highest quality categories.

Table IV presents regressions of counts of total number of unrated products in a subcategory on \(\text{LowMinCost}\) and a constant. Therefore the constant should be interpreted as mean number under high costs, and the coefficient on \(\text{LowMinCosts}\) is the difference between low and high cost contexts. The results reported in model (1) simply repeat the facts of the far left bar in Panel I of Table IV, that the mean number of unrated products is 240 under high costs, and 530 more (or 770) under low costs. Crucially, adding category effects and re-estimating coefficients on the basis of within category, across-subcategory variation produces statistically identical results, as in model (2).

\(<\text{TABLE IV}\>\)

Models (3) and (4) repeat the above analysis, but replacing the dependent variable with the count of numbers of highest rated products, or an average rating of greater than 4.5 out of 5. Model (3) effectively re-reports the information contained in the far right bar of Panel I, that there are 111 apps in the highest rated category under high costs and 124 more under low costs, or

\(^{19}\)Usual approaches to analyzing differences in distribution here that involve direct comparison of percentiles or quantiles (e.g., quantile regression) or direct comparisons of density distributions (e.g., kernel density estimate comparisons or Kolmogorov–Smirnov tests) do not provide most useful comparisons. Under the hypothesis of a large addition of mass to the bottom of a distribution—a large and additive uncentered change to the distribution—calculation of higher quantiles will be “dragged down”; and densities above the domain where this mass is added will be calculated as having declined, given that densities are a calculation of relative rather than absolute frequency.
Again, these estimates do not change when adding category effects, as in model (4).

A chief concern in interpreting user ratings is that ratings are reported by self-selected buyers (i.e., those who have chosen to report a rating and who have chosen to download and use the product). This could lead to thinly rated apps who are rated by lone enthusiasts or by the developer. Even simply the greater variance introduced by thin ratings can itself lead to over reporting of low quality products as more frequently high rated, given ratings are bounded from below.\footnote{In principle, analyzing patterns across large numbers of subcategories might “average away” the variance problem. (There is averaging across individual products in creating the dependent variable of total numbers of a products in a given category across an entire subcategory; there is also effectively averaging in the regression analysis, itself.) However, ratings of poor quality products are bounded from below, creating this possibility of systematically overstating numbers of higher quality products (and understating numbers of low quality products). While ratings of high quality products are themselves bounded from above, the likely higher demand and larger numbers of ratings for high quality products may make this a smaller problem.}

To examine whether possible bias or thin reporting accounts for the finding of higher numbers of high quality products, Table V presents results in which coefficients are re-estimated for counts of highest quality category (ratings greater than 4.5) when this variable is re-defined by counting just rating with some minimum number of ratings. Not counting products with few user ratings has at least two effects on the interpretation of results. First, it reduces possible bias from thinly rated apps, as above. Second, it focuses analysis on apps that should truly be of higher quality (presuming that apps with more ratings are those that are more highly demanded).

Model (1) reports results when regressing the count of apps with ratings of 4.5 or more on $\text{LowMinCost}$ and category effects—without excluding any products. These results therefore repeat those of model (4) of Table IV. To explicitly present a basis of comparison, the mean number of equally highly rated apps under relatively high minimum costs ($\text{MinFileSize} > 0.063$) is presented as the constant term.\footnote{This notation is kept to maintain consistency with earlier results; however, this constant or mean number of high products for high cost subcategories is estimated from the}
As reported in model (2), even when dropping all products with fewer than ten ratings, the coefficient on *LowMinCost* remains large and highly statistically significant. The drop in magnitude *per se* is not itself indicative of lower importance; as the number of counted products that go into the dependent variable declines, so too must the coefficients. It is perhaps more notable still that as the number of counted products per subcategory in the highest rated category drops from 203 to just 41 (see table), as all products with fewer than ten ratings are discarded, the results remain so very large and statistically significant. Given the necessarily change in magnitude of coefficients, I also report in the table the ratio of number of high quality apps, as defined, under low minimum costs versus those in high minimum costs. This ratio begins at 242 percent in model (1), meaning there are almost two and half times as many high quality apps under low costs where all observations are included in constructing the dependent variable; this ratio remains high, just under two times or 196 percent, when counting just products with at least 10 ratings in model (2).

Similar patterns are found despite the continual whittling down of the definition of the dependent variable to try to increasingly isolate products that are systematically of high quality (or at least to exclude as many false positives, as possible). Remarkably, even when excluding all apps but those with greater than 500 ratings, as in model (6), the estimated coefficient on *LowMinCost* remains marginally statistically significant at \( p = 10\% \). This is especially notable given mean numbers of high quality products under this stringent criterion are just 4.2 per subcategory; and where the point estimate remains large at 2.4 (i.e., greater numbers of highest quality products, defined as at least 4.5 rating with at least 500 ratings), and the ratio of apps in low versus high costs itself remains high at 149 percent.

In models (7) through (10), as the hurdle for being counted as part of the dependent variable becomes much higher, we see similar patterns across products with a thousand or more reviews. The thresholds of 82,149 and model without category effects, given these category effects complicates interpretation of the constant.
150,644 were included as these represent 95th and 99th percentile products in the entire population. Most remarkable of all, despite exceedingly low numbers of such products, with fewer than one such product per subcategory for higher thresholds, there continue to be systematically positive point estimates on LowMinCost, and relatively stable high ratios between counts of these hit products under low relative to high development costs.\textsuperscript{22} At the 99th percentile of numbers of ratings (150,644 ratings), there are on average only 0.01 products per subcategory receiving ratings greater than 4.5, under relatively high minimum development costs. Under relatively low minimum development costs, there are 0.03 more than this (or 437 percent more). Otherwise stated, where there is only one in ten subcategories with such a hit product, on average, in relatively high minimum cost contexts; in low minimum cost contexts there about four in ten.

\textless TABLE V \textgreater

\textless FIGURE V \textgreater

6 Summary and Conclusion

In this paper, I estimated the causal impact of especially low development costs on market participation (market structure) and product development in the context of app developers on the Apple AppStore. The empirical strategy was to exploit exogenous variation across 503 precisely-defined app subcategories, using minimum observed file size as a meaningful indication of variation in the required costs and complexity of developing different sorts of applications.

\textbf{Empirical Results.} I found that incremental cost declines result in incremental increases in numbers of market participants. However, once the proxy measure for minimum development costs, minimum file size, reaches 0.063

\textsuperscript{22}These consistently positive results depend on category effects being included in the model.
megabytes, subsequent cost declines led to an abrupt (even “kinked”) nonlinear response—more than doubling numbers of developers and adding a massive influx of very lowest quality products. For example, on average in each narrowly-defined subcategory, shifting below 0.063 MB added 845 more products, of which 530 did not even receive a single user rating. This large influx of hundreds of developers supplying very lowest quality products is nonetheless associated with greater absolute numbers of high quality products becoming available, by multiple measures and in a range of cuts of the data.

A Transition from “Lowering the Bar” to the “Bottom Falling Out”.

I interpret the nonlinear influx of lowest quality developers in response to incremental cost changes as consistent with a transition from “lowering the bar” (i.e., progressive lowering of the minimum quality threshold) to the “bottom falling out” of the market (i.e., the minimum quality threshold falling away altogether), as theorized earlier. This occurs where minimum costs fall below the value of nonpecuniary rewards for some share of developers.

Econometric results and descriptive patterns are each consistent with this transition creating a distinct subclass of developers on the platform. Despite all entering developers (in the theory) coming from the same continuum of quality and nonpecuniary rewards, this transformation of market selection and retention processes creates a distinct subgroup of developers who differ from professional firms and entrepreneurs in not being subject to usual market discipline. Thus, developers in this group—amateurs—are willing to persist on the platform despite facing a lack of commercial success, and even to make negative disbursements to continue doing so. (Here, this consisted in paying $100 annual platform fees and the opportunity cost of time and equipment spent in development.)

In principle, this nonlinear response to rather small cost declines could also reflect a highly nonuniform distribution of potential entrants. In this case, it remains plausible that there is no transition to the bottom falling out, but rather the standard prediction of continued lowering the bar. The underlying distribution is, of course, unobservable—and therefore it is not
possible to entirely rule this out. However, this would not account for the persistent negative (expected) returns and widespread negative disbursements documented here. This account would also require that a large–hundreds of thousands–of participating developers are uniformly and extraordinarily risk-seeking or irrational (and growing in number over time).\footnote{Under standard explanations, low quality developers—or even low quality developers with nonpecuniary motivations—would not make net negative disbursements in order to operate in the marketplace without having at least some reasonable probabilistic expectation of upside.}

**Interpretation of Increased Numbers of High Quality Products.** There are several plausible explanations for greater numbers of high quality products under especially low minimum development costs. For example, some fraction of low-quality amateurs might turn into high-quality professional suppliers, *ex post*. This interpretation of some amateur developers turning out to be successful *ex post*, in its simplest form, relates to prior research arguing the benefits of adding more independent experiments or “draws from the urn” (e.g., Nelson 1961; Abernathy and Rosenbloom, 1968; Aguiar and Waldfogel, 2015, 2016). The simplest form of this argument is that greater numbers of draws increases the expected extreme value or maximum—an order statistic argument—without changing the underlying distribution. Here, however, there is a profound transformation, fattening the very bottom of the distribution from where additional “draws” are made. This explanation therefore requires a context with especially high uncertainty and high variance in *ex post* product development outcomes.\footnote{This explanation comes closest Aguiar and Waldfogel’s (2015) interpretation of how cost declines in the pop music industry expanded the number of titles, and high quality titles with it. (Their analysis does not consider amateurs or a discrete and dramatic drop in the wider distribution of quality, as here.)} I speculate, too, that other conditions surrounding amateurs—nonpecuniary motivations and lack of market discipline—might have also contributed useful variance.

Another possible contributing explanation for greater numbers of high quality products is simply that lows bare minimum costs are related to low costs of experimentation and wider scope, more generally—in which case the added
high quality products could have come from professional firms.

**A Longer Long Tail.** To my knowledge there have been no prior systematic studies of the sources and importance of amateurs on market structure and dynamics. However, it should be noted that many examples previously understood as cases of superstars and skewed outcomes could possibly be better characterized as markets with many amateurs, instead—unpaid and without even any expectations (if not to say hopes) of “making it big.” Garage bands and rockstars are one such example. The results here suggest the possibility that it might be precisely this willingness for people to engage in the development that allows a market to be as successful as it is, in the end. Therefore, while it might not be privately optimal for individual amateurs with no reasonable expectation of commercial success to engage in development; as a collective, there remains a benefit to enlisting these efforts. The problem here is then solved by the interaction of especially low costs with nonpecuniary motivations. Thus, the long tail of developers on digital platforms, the long tail of garage bands, and the “long tail” of hobbyist entrepreneurs and hackers in the economy might be a good deal “longer” than standard theory would otherwise predict.
REFERENCES


Crecente, Brian. “Apple: App Store's success couldn't have happened without game developers” Polygon (Feb 6, 2015) downloaded at http://wwwpolygoncom/2015/2/6/7991031apple-app-stores-success-couldnt-have-happened-without-game-developers


TABLE I
BASELINE OLS REGRESSION RESULTS DESCRIBING THE RELATIONSHIP BETWEEN NUMBERS OF DEVELOPER FIRMS AND MINIMUM DEVELOPMENT COSTS

<table>
<thead>
<tr>
<th>Depvar:</th>
<th>No. Developer Firms per Subcategory [000s]</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(1)</td>
<td>(2)</td>
<td>(3)</td>
<td>(4)</td>
</tr>
<tr>
<td>MinFileSize</td>
<td>-3.86</td>
<td>-4.24</td>
<td>-4.29</td>
<td>-4.57</td>
</tr>
<tr>
<td></td>
<td>(.80)</td>
<td>(.86)</td>
<td>(.79)</td>
<td>(.81)</td>
</tr>
<tr>
<td>Games</td>
<td>.28</td>
<td>.28</td>
<td>.28</td>
<td>.28</td>
</tr>
<tr>
<td></td>
<td>(.12)</td>
<td>(.12)</td>
<td>(.12)</td>
<td>(.12)</td>
</tr>
<tr>
<td>Revenues</td>
<td>.04</td>
<td>.04</td>
<td>(.02)</td>
<td>(.02)</td>
</tr>
<tr>
<td>Constant</td>
<td>1.0</td>
<td>.94</td>
<td>.94</td>
<td>.94</td>
</tr>
<tr>
<td></td>
<td>(.06)</td>
<td>(.07)</td>
<td>(.07)</td>
<td>(.07)</td>
</tr>
<tr>
<td>Categories Effects</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
</tbody>
</table>

Notes. Standard errors clustered by app categories are reported in parentheses. MinFileSize is in megabytes. Number of observations = 503 subcategories.
### TABLE II
LOW MINIMUM DEVELOPMENT COSTS AND PRODUCT VARIETIES PER SUBCATEGORY

<table>
<thead>
<tr>
<th></th>
<th>Number of Product Varieties per Subcategory</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Depvar: Low Min Development Costs and Low vs High Cost Per Developer IV</td>
</tr>
<tr>
<td></td>
<td>(1)</td>
</tr>
<tr>
<td>LowMinCost (l[MinFileSize&lt;.063])</td>
<td>845.7</td>
</tr>
<tr>
<td>NumDevelopers</td>
<td>1.8</td>
</tr>
<tr>
<td>Constant</td>
<td>558.3</td>
</tr>
<tr>
<td>Categories Effects</td>
<td>Y</td>
</tr>
<tr>
<td>Adj-R^2</td>
<td>.07</td>
</tr>
</tbody>
</table>

Notes. Standard errors clustered by app categories are reported in parentheses. MinFileSize is in megabytes. Number of observations = 503 subcategories.
TABLE III  
LOW MINIMUM DEVELOPMENT COSTS AND PRODUCT PRICES

<table>
<thead>
<tr>
<th>Depvar:</th>
<th>Price</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>High vs Low Cost FE Per Developer Weighted Price &gt; 0</td>
<td>(1)</td>
<td>(2)</td>
<td>(3)</td>
<td>(4)</td>
</tr>
<tr>
<td><strong>LowMinCost</strong></td>
<td></td>
<td>.19</td>
<td>-0.19*</td>
<td>.04</td>
<td>-.21**</td>
</tr>
<tr>
<td>(MinFileSize&lt;.063)</td>
<td></td>
<td>(.15)</td>
<td>(.10)</td>
<td>(.18)</td>
<td>(.10)</td>
</tr>
<tr>
<td><strong>MinFileSize</strong></td>
<td></td>
<td></td>
<td></td>
<td>3.0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>(2.2)</td>
<td></td>
</tr>
<tr>
<td><strong>Constant</strong></td>
<td></td>
<td>1.3***</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>(.19)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Categories Effects</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>Inv. Freq. Weight</td>
<td></td>
<td></td>
<td>Y</td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>Adj-R^2</td>
<td>.00</td>
<td>.01</td>
<td>.01</td>
<td>.01</td>
<td>.01</td>
</tr>
</tbody>
</table>

Notes. Standard errors clustered by app categories are reported in parentheses. MinFileSize is in megabytes. Number of observations = 693,541 products.
TABLE VI
LOW MINIMUM DEVELOPMENT COSTS AND THE TAILS OF QUALITY

<table>
<thead>
<tr>
<th>Depvar:</th>
<th>Count of Lowest Quality (Unrated)</th>
<th>Count of Highest Quality (&gt;4.5 Rating)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>High vs Low Cost</td>
<td>High vs Low Cost</td>
</tr>
<tr>
<td></td>
<td>FE</td>
<td>FE</td>
</tr>
<tr>
<td></td>
<td>(1)</td>
<td>(2)</td>
</tr>
<tr>
<td></td>
<td>(3)</td>
<td>(4)</td>
</tr>
<tr>
<td>LowMinCost</td>
<td>530</td>
<td>124</td>
</tr>
<tr>
<td>(1[MinFileSize&lt;.063])</td>
<td>494</td>
<td>158</td>
</tr>
<tr>
<td>Constant</td>
<td>240</td>
<td>111</td>
</tr>
<tr>
<td></td>
<td>(66)</td>
<td>(20)</td>
</tr>
<tr>
<td></td>
<td>(68)</td>
<td>(22)</td>
</tr>
<tr>
<td></td>
<td>(25)</td>
<td>(17)</td>
</tr>
<tr>
<td>Categories Effects</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>Adj-R^2</td>
<td>.07</td>
<td>.12</td>
</tr>
<tr>
<td></td>
<td>.04</td>
<td>.14</td>
</tr>
</tbody>
</table>

Notes. Standard errors clustered by app categories are reported in parentheses. MinFileSize is in megabytes. Number of observations = 693,541 products.
**TABLE IV**

**ESTIMATES OF DIFFERENCES IN ABSOLUTE NUMBERS OF VERY HIGHEST QUALITY PRODUCTS BETWEEN SUBCATEGORIES WITH RELATIVELY LOW VERSUS HIGH MINIMUM DEVELOPMENT COSTS**

<table>
<thead>
<tr>
<th>Depvar:</th>
<th>Number of Highest Rated Products, &gt;4.5 out of 5 Rating</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apps Counted in</td>
<td>All counted</td>
</tr>
<tr>
<td>Per subcategory:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(1)</td>
</tr>
<tr>
<td>LowMinCost</td>
<td>157.8</td>
</tr>
<tr>
<td>(I{MinFileSize&lt;.063})</td>
<td>(22.1)</td>
</tr>
<tr>
<td>Constant</td>
<td>111.1</td>
</tr>
<tr>
<td></td>
<td>(17.2)</td>
</tr>
<tr>
<td>Est. Ratio No. Apps</td>
<td>2.42%</td>
</tr>
<tr>
<td>(LowMinCosts +</td>
<td></td>
</tr>
<tr>
<td>Const)/Const</td>
<td></td>
</tr>
<tr>
<td>Adj-R²</td>
<td>.14</td>
</tr>
</tbody>
</table>

Notes. Standard errors clustered by app categories are reported in parentheses. MinFileSize is in megabytes. Number of observations = 503 subcategories.
FIGURES

FIGURE I
TYPES OF DEVELOPERS IN THE FIRST PHASE SELECTED INTO THE MARKETPLACE, UNDER CLASSICAL MINIMUM QUALITY THRESHOLD

FIGURE II
TYPES OF DEVELOPERS IN THE FIRST PHASE SELECTED INTO THE MARKETPLACE, WITH LOW DEVELOPMENT COSTS
FIGURE III
PREDICTED NUMBERS (AND TYPES) OF DEVELOPERS VS. MINIMUM DEVELOPMENT COSTS

FIGURE IV
NONLINEAR CAUSAL ESTIMATES OF RELATIONSHIP BETWEEN NUMBERS OF DEVELOPERS AND MINIMUM DEVELOPMENT COSTS (PER SUBCATEGORY)
FIGURE V
AVERAGE PRODUCT QUALITY DISTRIBUTION (PER SUBCATEGORY)