Timely versus Quality Innovation:

the Case of Mobile Applications on Google Play and and iTunes

Yongdong Liu*, Denis Nekipelov* and Minjung Park*

[Very incomplete. Please, do not circulate]

This version: December 2013

Abstract

iTunes and *Google play* are dominant platforms where users of portable electronic devices operating under Android and Apple operating systems can purchase and download applications for those devices. The applications ('apps') are developed and brought to the platforms by a large number of independently operating developers. It is a highly competitive dynamic marketplace where the developers efforts are funded by the purchase activity of users and thus it becomes essential for the developers to keep innovating by both upgrading their existing apps and by introducing new apps to generate revenues. In this paper we use a unique dataset containing information regarding the user activity and introduction of apps on iTunes and Android platforms. Using a combination of techniques from the Computer Science literature, we were able to identify and validate the complete set of developers that operate on both platforms as well as the same apps that were introduced on both platforms. Using this dataset we study the relationship between the quality of an initially introduced app and the threat of competitor's entry. In particular, we find that the threat of competitor's entry can have a sizeable negative impact on the quality of the app which is being develop by forcing the developer to introduce an app prematurely, before it has been properly tested and debugged. Our reduced form analysis demonstrates varying effects of this phenomenon depending on both the size of the developer and its competitors and the sparcity of the product space in the app market on a given platform. Then we develop and estimate a structural strategic model of timing and quality decisions of the cross-patform app introduction. We use novel techniques from the Machine Learning to model the beliefs of developers in our semiparametric two-step estimator. The estimated structural model is then used to analyzed various counterfactual changes such as an increase in demand for certain app categories, transfers from the platform to the developers as well as 'A+B'-type contracts between the plaform and the developers on the resulting app quality across the markets.

Keywords: innovation; competing platforms; strategic entry; optimal stopping problem

---

*University of California, Berkeley. Generous supports from the National Science Foundation (SES 1024504) and the University of California at Berkeley are acknowledged. The usual disclaimer applies.

1 Introduction

The relationship between competition and innovation leads to a long debate and it can trace back to Schumpeter (1934) and Schumpeter (1942). The market for 'apps' – the programs that can be run under the operating systems that are installed on portable computer devices such as smartphones and tables – has been a phenomenon of a few recent years. Yet it represents a dramatic showcase of tradeoffs that arise for innovative firms at a highly competitive marketplace. The app market is represented by two dominant platforms that distribute apps for two competing operating systems: Android and Apple. The apps are not directly portable between the two platforms: without a substantial change in the source code an app developed for Apple cannot run on an Android device. In this paper we study the decisions of developers whose apps were successful on one platform to port those apps to another platform. In a competitive marketplace such a decision can be complex. If there are multiple competing developers who have apps that serve similar functions, the effort to introduce a quality app in th enew market can be undermined by competitors' entry which can divert the user base and reduce the potential profits from entry. In this case, a developer making such a decision faces a tradeoff between finishing the beta-testing and introducing a quality app but risking the loss of demand due to competitor's entry and introducing the app that has not been fully tested but then having to incur an increased cost fixing a running app.

We provide evidence that the timing and quality of entry in the app market is strategic and the firms making decisions about porting their apps to another platform make those decisions taking into account the actions of their competitors. In addition to providing a contribution to the literature on the role of competition onm the quality and timing of innovations, our paper also makes contributions to methodological literature on structural estimation of strategic competition models. Our structural model is based on the Bayes-Nash equilibrium where the developers make inferences regarding the app development processes of their competitors. Then, on the basis of this prediction they make the decision that trades off the quality of the app that they want to introduce on the new platform and the duration of the development process. To

provide a statistical procedure that imitates the prediction process for the developers, we use a range of tools from the Machine Learning to create a statistical model that incorporates all information available to developers in an efficient way leading to an accurate and parcimonious predictor. The developers also make predictions regarding the performance of their own app on a new platform based on its observed performance on the platform of the "first entry". We model this process by invoking a range of model selection procedures that allow us to detect which characteristics of app performance on one platform are facilitating the prediction of its performance on another platform. In particular, we use a novel approach and include the variable that classifies the pattern of app performance (i.e. the shape of the trend of the growth of demand indicators over time) into a finite set of categories. We then invoke a pseudo-likelihood approach to estimate the parameters of the structural model based on the first-stage estimates based on the described prediction models.

Our analysis sheds a new empirical light on the process of innovation that has been long looked at from the theoretical viewpoint. The empirical support for those is frequently weak because few datasets offer a clean homogeneous evidence regarding innovation mainly because industrial and BioTech innovation has a very long cycle making a construction of sufficiently large comprehensive datasets hard, if not impossible in some cases. Schumpeter suggests that market power might increase the rents of innovations and thus firm's incentive to innovate called "Schumpeterian effect" by Dasgupta and Stiglitz (1980). Besides Schumpeterian effect, several other theories also try to explain the relationship between innovation and competition, for instance, the replacement effect (Arrow (1962)). Innovation would replace monopoly profits with the profits from innovation and what the monopolist benefits from the innovation is the incremental profits. Therefore, a firm in a monopoly position has less incentive to innovate than a firm in a competitive industry. Gilbert and Newbery (1982) and, later, Aghion et al. (2009) show that the incentive to preempt may be stronger than Arrow's replacement effect. Even the well-established market power of a monopolist may still be challenged by the introduction of new products or cost saving innovation by its competitors. Then innovation can help the monopolist forestall competition.

Our paper also provides the empirical evidence on the role of strategic interactions in the innova-

tion decisions. The idea of interactions previously existed in theoretical literature. For instance, Loury (1979) discusses the strategic innovation competition among identical firms. Facing the uncertainties coming from innovation and market competition, firms are maximizing their expected profits. The major finding is that competition decreases individual firms innovation incentive. Dixit (1987) finds that in a game of two asymmetric plays, the player with higher odds to win tends to exerts more efforts and the underdog commits lower efforts. However, in a game with multiple plays, the absence of an odds-on player would sufficiently cause any one competitor to overcommit. The innovation and market structure are also studies in a more dynamic framework. Klepper (1996) summarizes six major regularities of product life cycle and models the market evolution from birth through maturity of an industry. The theoretical model shows that at the birth of an industry, entry and product innovation are very active. However, the advantage of size in process R&D makes firm invest more on process R&D and eventually the cost disadvantage prohibits new entries. After entries and the number of firms decline, the product R&D and the diversity of competing product are compromised.

Besides the monotonic relationship between competition and innovation, the inverted-U curve relationship also has a long history (Scherer (1967)). In a recent paper, Aghion et al. (2005) show that an inverted-U curve relationship between market competition and innovation theoretically and empirically. If a larger equilibrium fraction of sectors involves neck-and-neck competing incumbents, the competition tends to induce more innovation because of the dominant escape-competition effect. However, in sectors where innovations are made by laggard firms with already low initial profits, competition reduces innovation incentive.

Early empirical work on innovation and market structure mostly use industrial data (Levin et al. (1985), Geroski (1995), Nickell (1996), Blundell et al. (1999)). Recent papers start to use micro level data, such as Aghion et al. (2005) and Aghion et al. (2009). However, empirical findings do not draw very clear conclusions on market structure and innovation.

Besides some studies on platform and network effect (Katz and Shapiro (1985) and Farrell and Saloner (1986)), the burgeoning literature on two-sided market is still very recent. A well-studied

topic in this area is the pricing strategy in two-sided markets, such as Armstrong (2009), Rochet and Tirole (2006). Another stream of literature studies which sides are subsidized, but as Rochet and Tirole (2003) shows, in different markets either seller or buyer could be subsidized. Rysman (2009) is a good review of early literature on two-sided market. More recent literature on two-sided market covers wider topics beyond intra-platform analyses. For example, Hagui (2009) and Lee (2012) both study the inter-platform completion. Lee (2012) studies a bilateral contracting game among platforms and firms with externalities and results show that contingent contract may be less efficient than payment schema not made. Hagui (2009) studies the inter-platform competition, especially the pricing strategy in two-sided market. His model shows that user preference for product variety makes producer less substitutable. As a result, platform charges higher price on users in equilibrium. Lee (2013) empirically studies the market impacts of vertical integration occurring to platform.

This paper is organized as follows. In Section 2 we describe the raw data and the methodology for construction of the dataset of matched developers and apps across the platforms. In Section 2 we also provide the reduced-form evidence linking the quality of the app, the duration between the entries into two platforms, and the intensity of competition. In Section 3 we set up and estimate a structural model of competing entry of multiple developers across the platforms. To estimate the model we use the two-step approach. In the first step we use a class of model selection techniques from the Machine learning to construct a flexible model for expectations of app developers. Section 4 concludes.

2 Data

2.1 Basic descriptions

Our data comes from AppMonsta Inc., the company specializing in collection and analysis of the data from the app stores and consulting services to new and existing developers. The data comprise all iTunes apps and Android apps from October 2011 to October 2012. They were scraped from iTunes App Store and Google Play App Store on a daily basis. We have constructed

a daily panel data of all iTunes and Android apps. Table 1 lists the variables in our data set and brief descriptions. App characteristics available on Apple Store and Google Play are slightly different and table 1 marks the variables unavailable on each platform.

Although Google Play provides developer names, name cannot be used as the unique developer ID. We know apps by the same developers, and this information enables us to create a unique developer ID for Android apps. Some variables do not change very much over time, such as app name and developer name. Other variables are updated with every new version. App size, version description and Require OS are such variables. However, some other variables would change every day, for instance, app price and the number of ratings. Despite these different cases, we still organized our data as a daily panel date set.

In general the task of re-identifying observations from separate datasets is a routine necessity in many practical applications. In many cases there do not exist perfect cross-database identifiers of individual entries. There could be multiple reasons why that is the case. For instance, there could be errors in data entry and processing, wrong variable formatting, and duplicate data entry. The Computer Science literature has developed an extensive machinery for re-identification of such observations. The idea that has arisen in Newcombe, Kennedy, Axford, and James (1959) and was later formalized in Fellegi and Sunter (1969) was to treat the record linkage problem as a problem of classification of record subsets into matches, non-matches and uncertain cases. This classification is based on defining the similarity metric between each two records. Then given the similarity metric one can compute the probability of particular pair of records being a match or non-match. The classification of pairs is then performed by fixing the probability of erroneous identification of a non-matched pair of records as a match and a matched pair of records as a non-match by minimizing the total proportion of pairs that are uncertain. This matching technique is based on the underlying assumption of randomness of records being broken. As a result, using the sample of perfectly matched records one can recover the distribution of the similarity metric for the matched and unmatched pairs of records. Moreover, as in hypothesis testing, one needs to fix the probability of record mis-identification. Finally, the origin of the similarity metric remains arbitrary.

A large fraction of the further literature was devoted to, on one hand, development of classes of similarity metrics that accommodate non-numeric data and, on the other hand, development of fast and scalable record classification algorithms. For obvious reasons, measuring the similarity of string data turns out to be the most challenging. Edit distance (see, Gusfield (1997) for instance) is a metric that can be used to measure the string similarity. The distance between the two strings is determined as the minimum number of insert, delete and replace operations required to transform one string into another. Another measure developed in Jaro (1989) and elaborated in Winkler (1999) is based on the length of matched strings, the number of common characters and their position within the string. In its modification it also allows for the prefixes in the names and is mainly intended to linking relatively short strings such as individual names. Alternative metrics are based on splitting strings into individual "tokens" that are substrings of a particular length and then analyzing the power of sets of overlapping and non-overlapping tokens. For instance, Jaccard coefficient is based on the relative number of overlapping and overall tokens in two strings. More advanced metrics include the "TF/IDF" metric that is based on the term frequency, or the number of times the term (or token) appears in the document (or string) and the inverse document frequency, or the number of documents containing the given term. The structure of the TF/IDF-based metric construction is outlined in Salton and Harman (2003). The distance measures may include combination of the edit distance and the TF/IDF distance such as a fuzzy match similarity metric described in Chaudhuri, Ganjam, Ganti, and Motwani (2003).

[Table 1 about here.]

[Table 2 about here.]

Table1 and Table2 show the app distribution cross genres in Android and iTunes respectively. "Utilities & Tools", "Games" and "Entertainment" account for more than 40% Android apps. In iTunes app store, there are fewer "Utilities & Tools" apps, but more "Books & Reference", "Education" and "Lifestyle" apps. The third column in Table1 and Table2 reports the proportion of

cross-platform apps in each genre. 11.4% Android apps are cross-platform apps and that number is only 7.3% in iTunes. According to app launch time, we could identify the cross-platform entry order. Since the launch time of some Android apps are missing in our data, some entry orders cannot be identified. In accurately identified sample, about 44% cross-platform apps were launched on Android first, which are twice more than the apps first launched on iTunes. In terms of absolute number, "Game", "Entertainment", "Business", "Lifestyle" and "Education" have more cross-platform apps than other genres, but "Business", "Finance", "Social Networking", "Medical" and "Travel" have higher proportion of cross-platform apps.

[Table 3 about here.]

[Table 4 about here.]

Table3 and Table4 list the four major app characteristics cross genres. The number of ratings refers to the total number of ratings received by an app by the end of data period. Without the control for app tenure, app quality measured by this variable is usually misleading, since older apps tend to accumulate more ratings. Therefore, the weekly incremental number of ratings is provided as a standardized measure. We also list monthly major update frequency and monthly minor update frequency as other dimensions of app quality. In general, "Game", "Social Networking" and "Entertainment" apps harvest more ratings and accumulate ratings more quickly. However, apps in these genres do not necessarily update more often than apps in other genres.

[Table 5 about here.]

[Table 6 about here.]

We also report the app characteristics by different entrant groups according to the cross-platform entry order. Despite the fact that cross-platform apps are more originally from Android, Table5

and Table6 show that cross-platform apps originating from iTunes, in other words cross-platform apps launched on iTunes first, have more ratings. Even controlled for app tenure, weekly incremental ratings is still much higher on these apps. And this difference is specially great for ratings measured in iTunes App Store. However, this rating superiority does not hold for cross-platform apps originating from Android. Even more, in iTunes app store the cross-platform apps coming from Android have worse performance on ratings compared with single-platform iTunes apps. These two tables also show that cross-platform apps, in particular apps from iTunes, tend to update more frequently.

We further compare the app characteristics by genres and entrant groups, by calculating the difference of app characters between entrant group and single-platform apps in that genre. If the dot representing the difference lies below zero in Figure3 and Figure4, it means that this entrant group has lower value than single-platform apps in that genre. Dots from the same entrant groups are connected by solid, dash or dotted lines in order to distinguish from each other. The relative position of three lines reveals the app performance rank of different entrant groups. First of all, the solid line representing cross-platform apps originating from iTunes seldom lies below zero. This implies that these apps have better rating performances and update more often. Rating performances are also very heterogeneous cross genres. In upper panels of Figure3 and Figure4, there are several spikes at "Games", "Photo & Video", "Utilities and Tools" and some other genres. On the contrary, other genres reside on zero or even below zero. Android and iTunes app markets share similar patterns, but these patterns are much clearly defined in iTunes.

[Figure 1 about here.]

[Figure 2 about here.]

We further compare the app characteristics by genres and entrant groups, by calculating the difference of app characters between entrant group and single-platform apps in that genre. If the dot representing the difference lies below zero in Figure3 and Figure4, it means that this entrant group has lower value than single-platform apps in that genre. Dots from the same entrant groups

are connected by solid, dash or dotted lines in order to distinguish from each other. The relative position of three lines reveals the app performance rank of different entrant groups. First of all, the solid line representing cross-platform apps originating from iTunes seldom lies below zero. This implies that these apps have better rating performances and update more often. Rating performances are also very heterogeneous cross genres. In upper panels of Figure3 and Figure4, there are several spikes at "Games", "Photo & Video", "Utilities and Tools" and some other genres. On the contrary, other genres reside on zero or even below zero. Android and iTunes app markets share similar patterns, but these patterns are much clearly defined in iTunes.

[Figure 3 about here.]

[Figure 4 about here.]

The competition level in app market is measured by Herfindahl index. Since the number of downloads is unobservable, the accumulative number of ratings is used to approximate the number of download. Apps market store is the ratio of its total ratings to the overall ratings in that genre. Each genre is treated as an independent market and the Herfindahl index is calculated by each genre. Figure5 0 shows the Herfindahl index in two platforms during our data period. The blue line represents Android and the red one is HHI of iTunes. In most genres, the competition level of Android app store is decreasing. On the contrary, the competition level of iTunes app keeps at a general stable level, though HHI in some genres are bumpier than others.

[Figure 5 about here.]

2.2 Reduced-form evidence

Developer is ranked by the number of its apps and is indicated whether it is the top50 or top100 developers in particular genre. In this section, we focus on the cross-platform entries following these top developers. Suppose one Android app by a top50 developer enters iTunes at data t, then the occurrence of entering iTunes by other developers is recorded. We separate these followers

into apps by other top developers and apps by non-top developers. We plot the frequency of cross-platform entries following a week, two weeks, and four weeks after the cross-platform entry of an app by top developer.

[Figure 6 about here.]

[Figure 7 about here.]

Figure6 shows the kernel density of cross-platform entry following the cross-platform entries of top Android developers. Figures7 shows the similar kernel density of iTunes apps. Top panels show the kernel density of entries of non-top developers and bottom panels show the kernel density of entries of top developers. The distribution of the number of following entries is very skew leftwards, mainly because there are not many following entries and high peaks appear on the left end. Therefore, there are on the left tail. The following entries by non-top developer have slightly evener distributions and the descending trend from the peak to the right tail gets less steep.

[Table 7 about here.]

[Table 8 about here.]

To compare the quality of followers, we calculate the deviation of mean and median of followers from the genre mean and median. App quality is measured on three aspects, weekly incremental number of ratings, monthly major updates and monthly minor updates. Weekly incremental number of ratings reflects how many ratings received by an app on weekly average. High quality apps usually get more ratings and therefore they should have higher weekly incremental ratings. Major update is usually related to some major changes or improvements, such as another function added or new interface. Minor update is mainly bug fixing. High quality apps involve fewer bugs and consequently minor updates should be less often. Weekly incremental number of ratings and

minor updates frequency are two strong indicators of app quality. Table7 and Table8 list the difference between mean (median) of followers and genre mean (median). Column I, II and III are weekly incremental number of ratings, major updates frequency and minor updates frequency respectively. No matter whether followers are apps by other top developers or apps by non-top developers, they are inferior to the genre average and median level.

[Figure 8 about here.]

[Figure 9 about here.]

[Figure 10 about here.]

[Figure 11 about here.]

[Figure 12 about here.]

[Figure 13 about here.]

[Figure 14 about here.]

[Figure 15 about here.]

[Figure 16 about here.]

[Figure 17 about here.]

[Figure 18 about here.]

[Figure 19 about here.]

Next we want to see the relationship between app quality and entry duration controlled for the Herfindahl index and the app quality. First, run the regression below

$$d_{ij} = c + a_1 HHI_{itunes} + a_2 HHI_{android} + a_3 q_{ij} + \zeta_j + \epsilon_{ij}$$

Subscript i represents app and j refers to developer. d_{ij} is the day gap between entries of top developers and entries of followers. HHI_{itunes} and $HHI_{android}$ are genre specific Herfindahl indies. q_{ij} is the quality measure of app i. In the regression above, we also control for the developer fixed effect, ζ_i. Estimate this equation on each genre and calculate

$$\tilde{d}_{ij} = \hat{c} + \hat{\zeta}_j$$

$\tilde{d}_{ij}$ is the duration controlled for the Herfindahl index and the app quality.

Figure8 to Figure19 plot the relation between entry duration and app quality. Each row plots two genres. The first and the third graph show the uncontrolled duration. The second and the fourth graph show the controlled duration. Figure8 to Figure13 are for Android apps. Figure14 to Figure19 plot iTunes apps. In graphs of uncontrolled duration, there is no clear pattern. However the controlled graphs show that followers entering at later time have higher qualities than haste followers.

3 Structural model

3.1 Theoretical setup

We now consider a structural model of the cross-platform behavior of the developers. According to our reduced-form evidence, we assume that the developers have a type space $\mathcal{T} = \{L, S\}$ corresponding to the large and small developers. We assume that the entire set of developers is fixed with $d^L = 1, \ldots, D^L$ indexing developers of type L and $d^S = 1, \ldots, D^S$ indexing the developers of type S. This entire set of players is considered to be common knowledge to all

the participating developers. The plaforms are indexed $\mathcal{P} = \{A, G\}$ where A stands for Apple iTunes and G stands for Google play. We assume that there is an exogenous two-dimensional process $(\xi_A^{d^\tau,\tau}(t), \xi_G^{d^\tau,\tau}(t))$ with values in $\{0,1\} \times \{0,1\}$ (where 1 stands for entry) and frequency $(\lambda_A^\tau(t), \lambda_G^\tau(t))$, $\tau \in \mathcal{T}$ which characterizes the "first entry" of developer of type τ on one of the plaform. To emphasize that this is the process of the first entry, we require that if for some t $\xi_A^{d^\tau,\tau}(t) \neq \xi_G^{d^\tau,\tau}(t)$, then $\xi_A^{d^\tau,\tau}(t') \neq \xi_G^{d^\tau,\tau}(t')$ for any $t' > t$. This characterization, however, allows for the first entry to be simultaneous to both plaforms. The process $(\xi_A^{d^\tau,\tau}(t), \xi_G^{d^\tau,\tau}(t))$ can be considered as a reduced-form interpretation of the "first entry" decisions by the developers. The full structural characterization of this process can be found in Liu (2013).

Consider a continuous-time game of cross-platform entry at time t. For each developer d^τ such that $\xi_A^{d^\tau,\tau}(t) + \xi_G^{d^\tau,\tau}(t) = 0$, the action space is $\mathcal{A}^{d^\tau} = \emptyset$, meaning that the developers who have not done the first entry cannot do the cross-platform entry. For each developer d^τ such that $\xi_A^{d^\tau,\tau}(t) + \xi_G^{d^\tau,\tau}(t) = 2$, the action space is $\mathcal{A}^{d^\tau} = \emptyset$, meaning that the developers who have entered both markets upon the first entry cannot do the cross-platform entry. Finally, for each developer d^τ such that $\xi_A^{d^\tau,\tau}(t) + \xi_G^{d^\tau,\tau}(t) = 1$, the action space at time t is $\mathcal{A}^{d^\tau}(t) = \{t',\, t' \geq t,\, t, t' \in \mathbb{R}\}$ intersected with an empty set. The entire action space $\mathcal{A}^{d^\tau} = \cup_{t\in[0,+\infty)}\mathcal{A}^{d^\tau}(t)$ In other words, for each developer can decide to make a cross platform entry in period t' in the future, unless the entry has occured in the past.

The "instant" payoff of each developer is characterized by the cost c indicating the cost of testing the app. The payoff from the cross-platform entry is characterized by the potential revenue R as well as the function $C(\Delta)$ where Δ is the interval of time between the first entry and the entry on another platform. We assume that $C(\Delta)$ is a decreasing function. The future payoff also has a component $\gamma\, N(t)$, where $N(t)$ is the number of entrants into the same category from competitors at time t. The time-varying uncertainty in the model is characterized by a pair of independent Brownian motions $B^d(\cdot)$ and $B^c(\cdot)$, where $B^d(\cdot)$ reflects the uncertainty associated with the developer-specific prediction for her revenues and $B^c(\cdot)$ reflects the market-level uncertainty. We denote the discount factor for the developer by ρ Thus, the payoff at the

time t if the cross-platform entry does not occur at t (using the time of the first entry as 0) is

$$U_t^{d,n} = \int_0^t e^{-\rho\,\tau}\left(-c\,d\tau + \sigma_R\,dB^d(\tau) + \sigma_M\,dB^c(\tau)\right).$$

The utility at the instant of cross-platform entry is

$$U_t^{d,e} = R - C(t) - \delta\,E^d[N(t)],$$

where the expectation $E^d[\cdot]$ is taken with respect to the distribution of uncertainty and beliefs of developer d at time t.

To characterize the best responses, we consider a set of threshold rules

$$a^d = t, \quad \text{if} \quad U_t^{d,n} \leq U_t^{d,e} \quad \text{and} \quad U_{t'}^{d,n} \geq U_{t'}^{d,e}. \tag{3.1}$$

LEMMA 1. *The decision rules (3.1) are unique best responses in the action space* $\mathcal{A}^{d^\tau}$

Next, we proceed with the characterization of an equilibrium.

3.2 Estimation of the structural model

We will use a linear probability model to predict the cross-platform entry. However, we have very rich app characteristics and market characteristics in our data. To avoid model over fitting, techniques of model selection is applied to select the "best" prediction model. Controlling for the propensity of entering the other market from the first stage, we estimate a Cox duration model on app's quality measures in the second stage and this regression will reveals a relative clear relation between app's performance and its cross-entry timing.

The first stage base model is a linear probability model of cross-entry on app characteristic and market characteristics.

$$crossentry = \alpha + \beta X + \epsilon$$

crossentry is an indicator of whether an app entered the other platform. X is a collection of market and app characteristics which may have impact on cross-entry. We applied backward, forward and both direction search based upon criteria AIC, BIC and adjusted R-square. We also exploited an exhaustive search for the best subsets of variables using an efficient branch-and-bound algorithm. Exhaustive search returns the best subset of variables for each model size. According to some penalty criteria on model size such as AIC and adjusted R-square, the best model size and the corresponding best subset of variables are determined. Backward, forward, both direction and exhaustive research return very similar sets of selected variables, so we follow the selection results by exhaustive search. Table9 lists the variables used for model selection and the variables selected for the prediction of cross-entry. This linear probability model is estimated on "big" and "small" developers separately to control for the developer size heterogeneity. A developer is considered to be "big", if the number of apps developed by this developers is on top 30% in that particular genre. Otherwise, this developer is treated as a "small" developer.

[Table 9 about here.]

In the second stage, a Cox proportional hazard model is estimated.

$$h(t|X) = h_0(t)exp(X\beta)$$

X contains app performances before entry, performance after entry and predicted entry probability. App performances are measured by the total number of ratings and weekly incremental number of ratings. Monthly major and minor updates are also used to approximate app quality. The performance before entry is measured at the instant of actual cross entries. For censored observations, they are taken at the last observed time period. Performance after entry are mainly measured by update frequency within one month after entry. Rating for new apps is not very reliable, since it takes time for new apps to diffuse and further accumulate ratings stably. For "censored" apps that do not cross-enter, these variables are unavailable. A dummy for censored apps is introduced to solve the issue. An interaction term of this dummy variable and update

frequency is added to the regression. Table10 and Table11 report the coefficients of Cox model for Android and iTunes apps respectively.

[Table 10 about here.]

[Table 11 about here.]

4 Conclusion

References

Chaudhuri, S., K. Ganjam, V. Ganti, and R. Motwani (2003): "Robust and efficient fuzzy match for online data cleaning," in *Proceedings of the 2003 ACM SIGMOD international conference on Management of data*, pp. 313–324. ACM.

Fellegi, I., and A. Sunter (1969): "A theory for record linkage," *Journal of the American Statistical Association*, pp. 1183–1210.

Gusfield, D. (1997): *Algorithms on strings, trees, and sequences: computer science and computational biology*. Cambridge University Press.

Jaro, M. (1989): "Advances in record-linkage methodology as applied to matching the 1985 census of Tampa, Florida," *Journal of the American Statistical Association*, pp. 414–420.

Newcombe, H., J. Kennedy, S. Axford, and A. James (1959): "Automatic linkage of vital and health records," *Science*, 130, 954–959.

Salton, G., and D. Harman (2003): *Information retrieval*. John Wiley and Sons Ltd.

Winkler, W. (1999): "The state of record linkage and current research problems," in *Statistical Research Division, US Census Bureau*. Citeseer.

Figure 1: App quality by genres and entrants(Android)

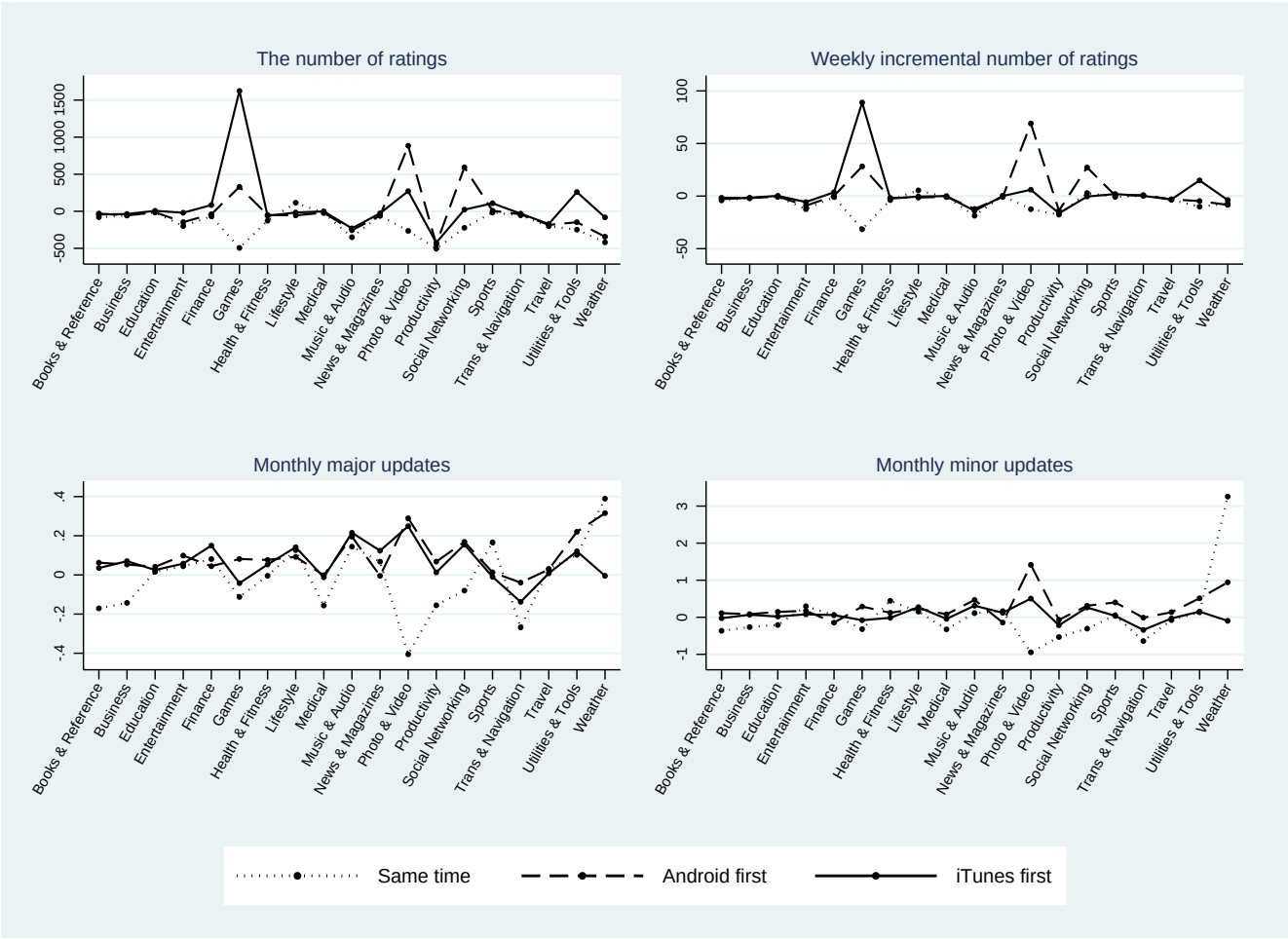


Figure 2: App quality by genres and entrants(iTunes)



Figure 3: App quality by genres and entrants(Android)

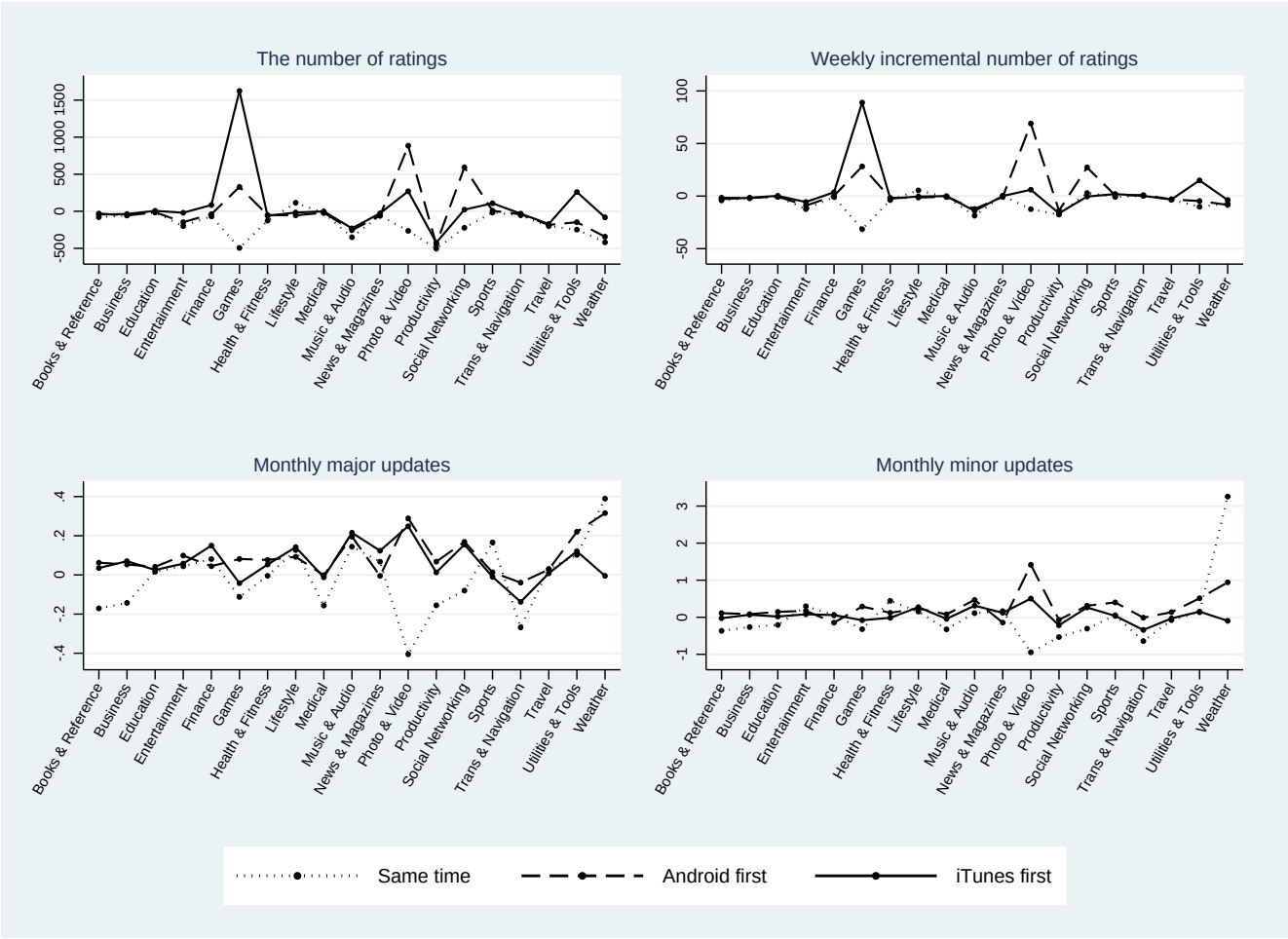


Figure 4: App quality by genres and entrants(iTunes)



Figure 5: Herfindahl index cross Genre and Platform

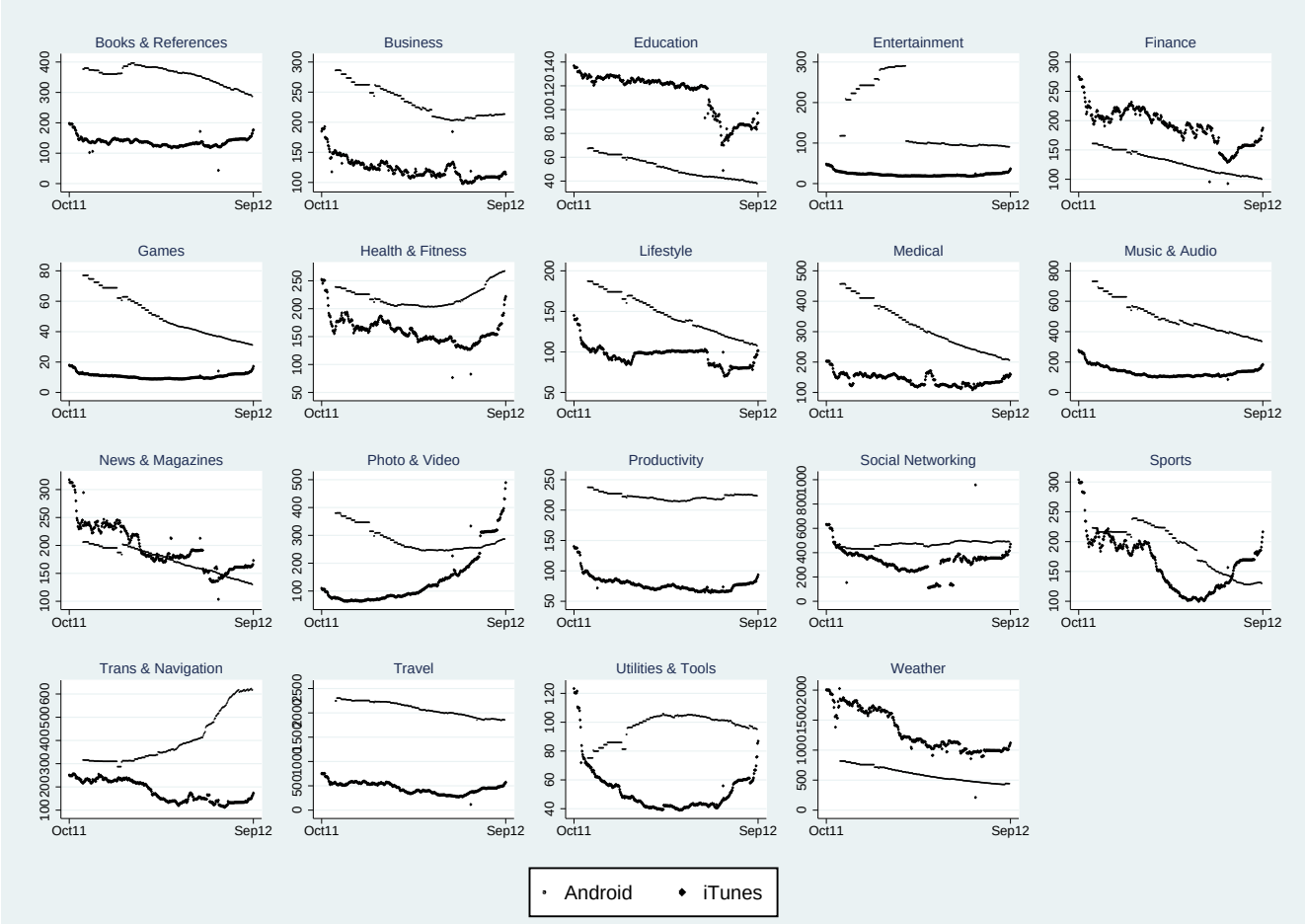


Figure 6: the frequency of entry following the entry of the top 50/100 developer on Android (Top panel is non-top follower and bottom panel is top follower)

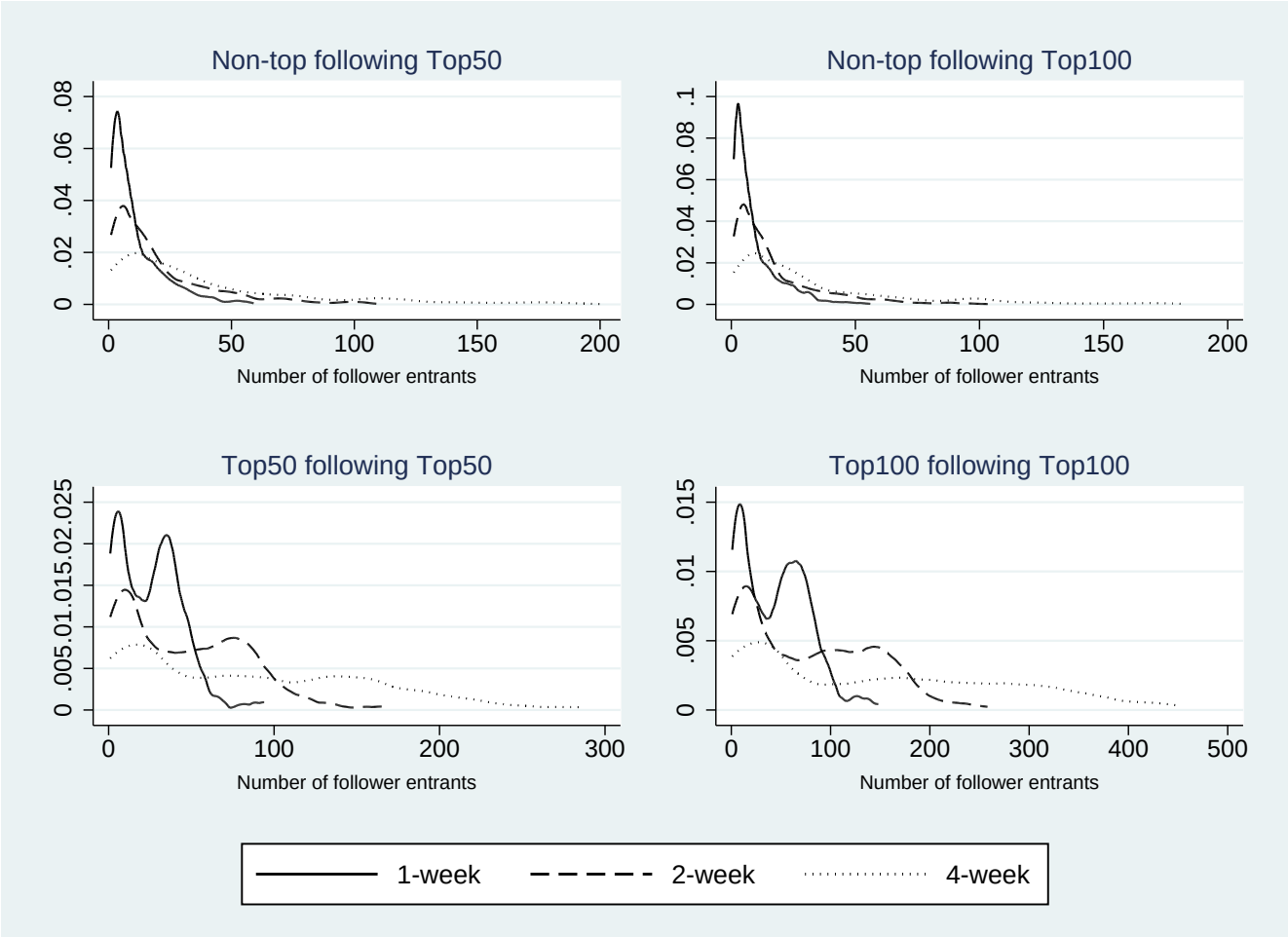


Figure 7: the frequency of entry following the entry of the top 50/100 developer on iTunes (Top panel is non-top follower and bottom panel is top follower)

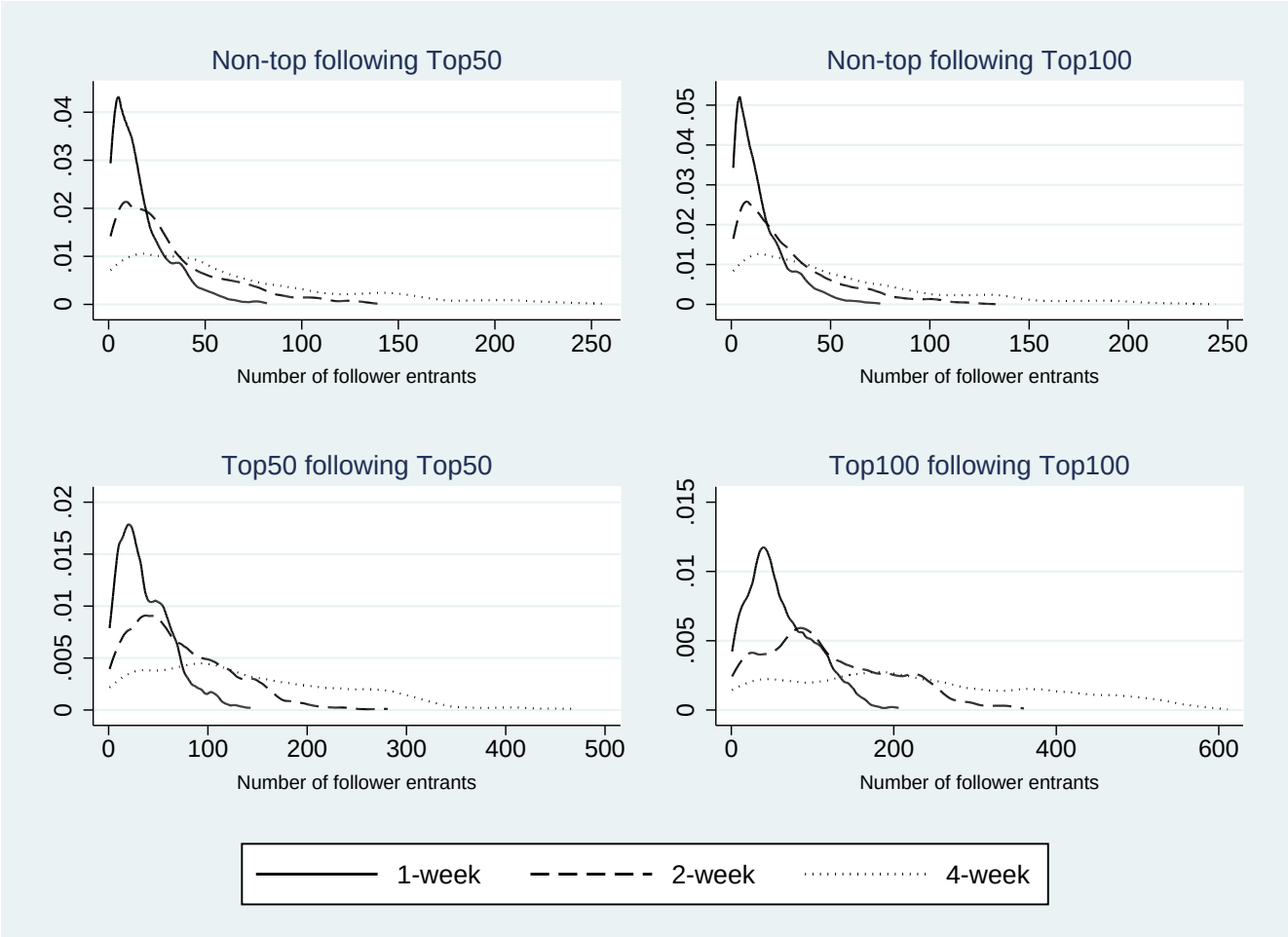


Figure 8: Weekly Incremental Ratings of Apps from Non Top50 Developers Following Top50 Developers within 2 weeks (Android)

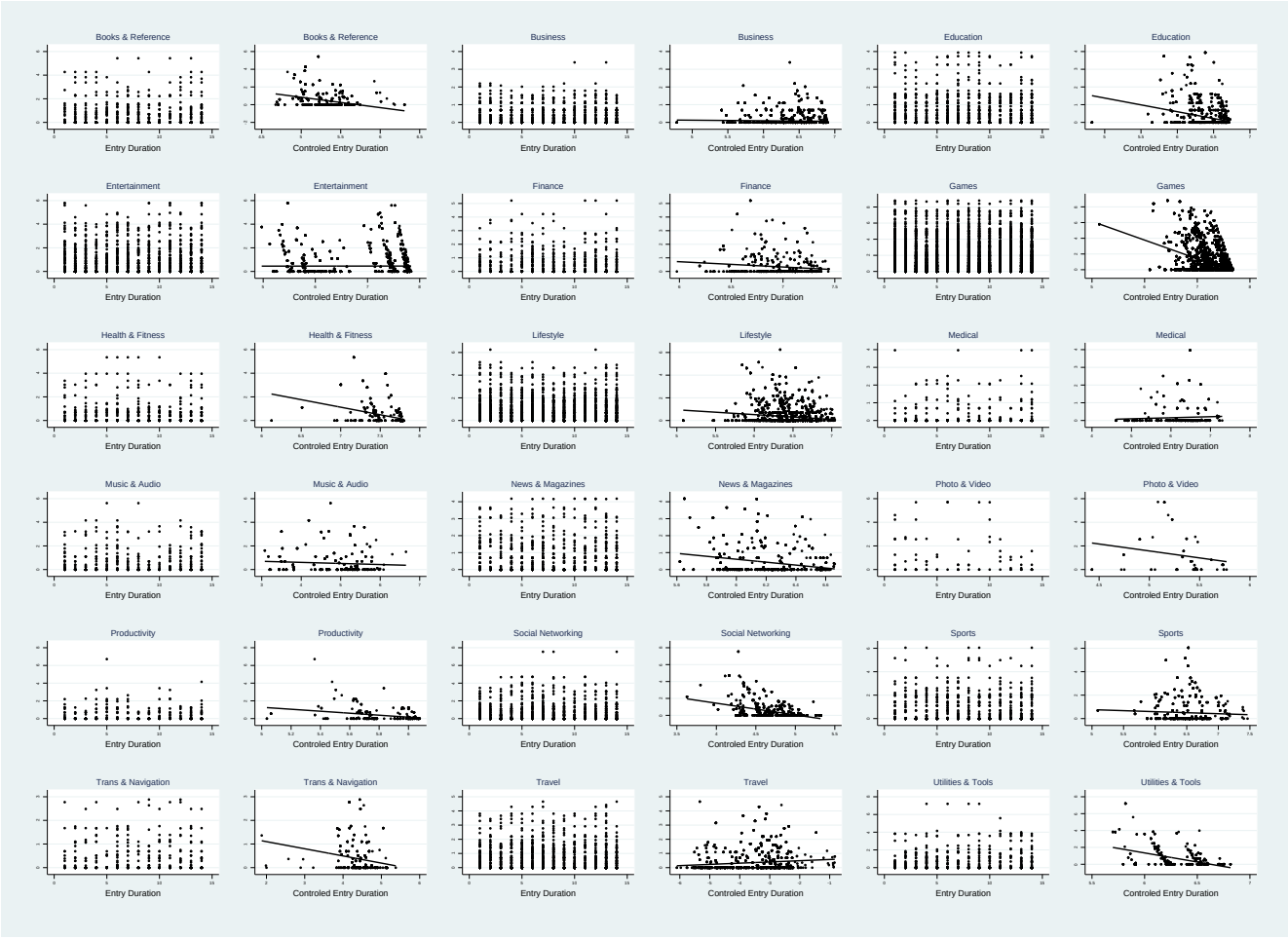


Figure 9: Monthly Major Updates of Apps from Non Top50 Developers Following Top50 Developers within 2 weeks (Android)

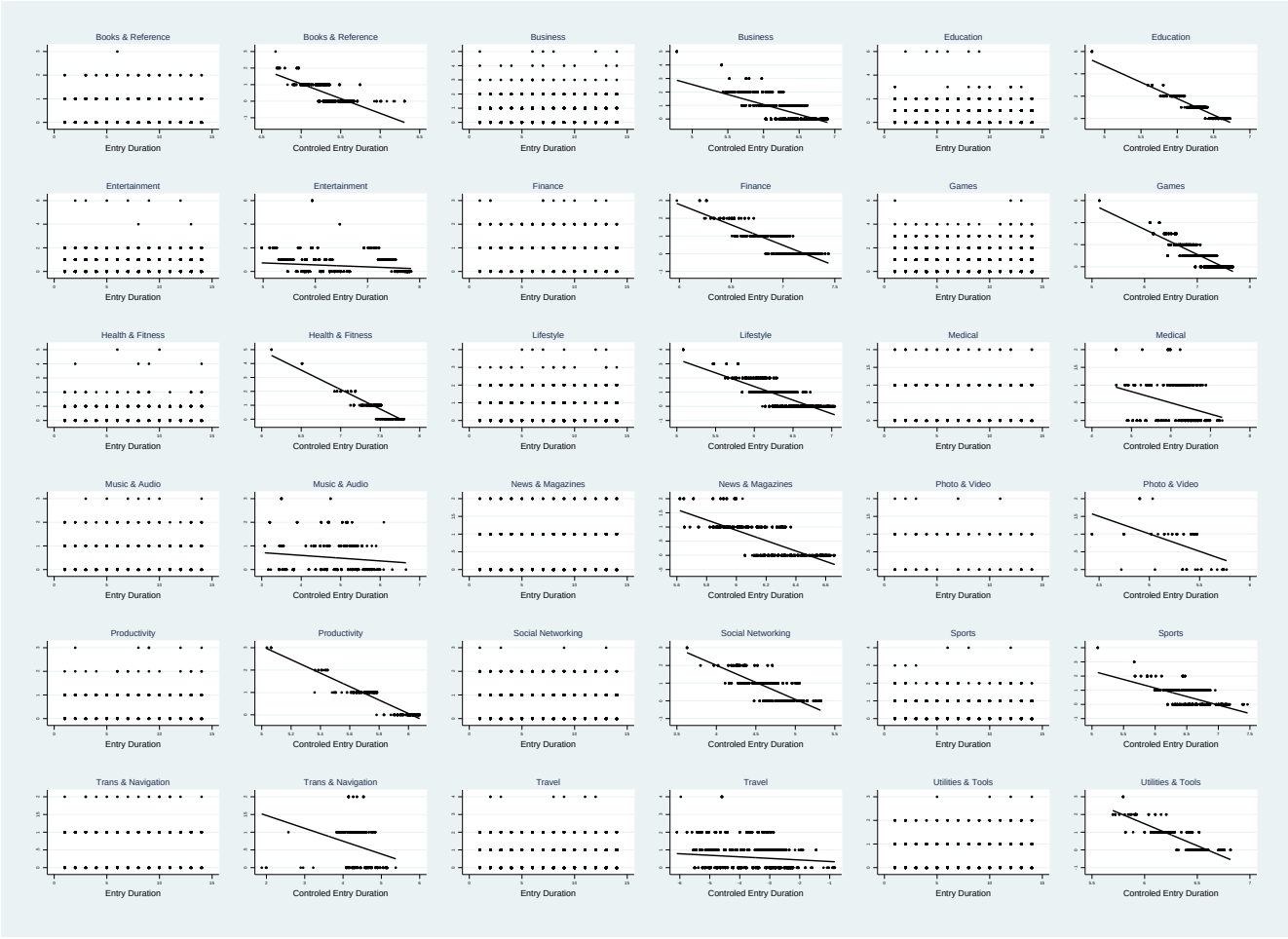


Figure 10: Monthly Minor Updates of Apps from Non Top50 Developers Following Top50 Developers within 2 weeks (Android)

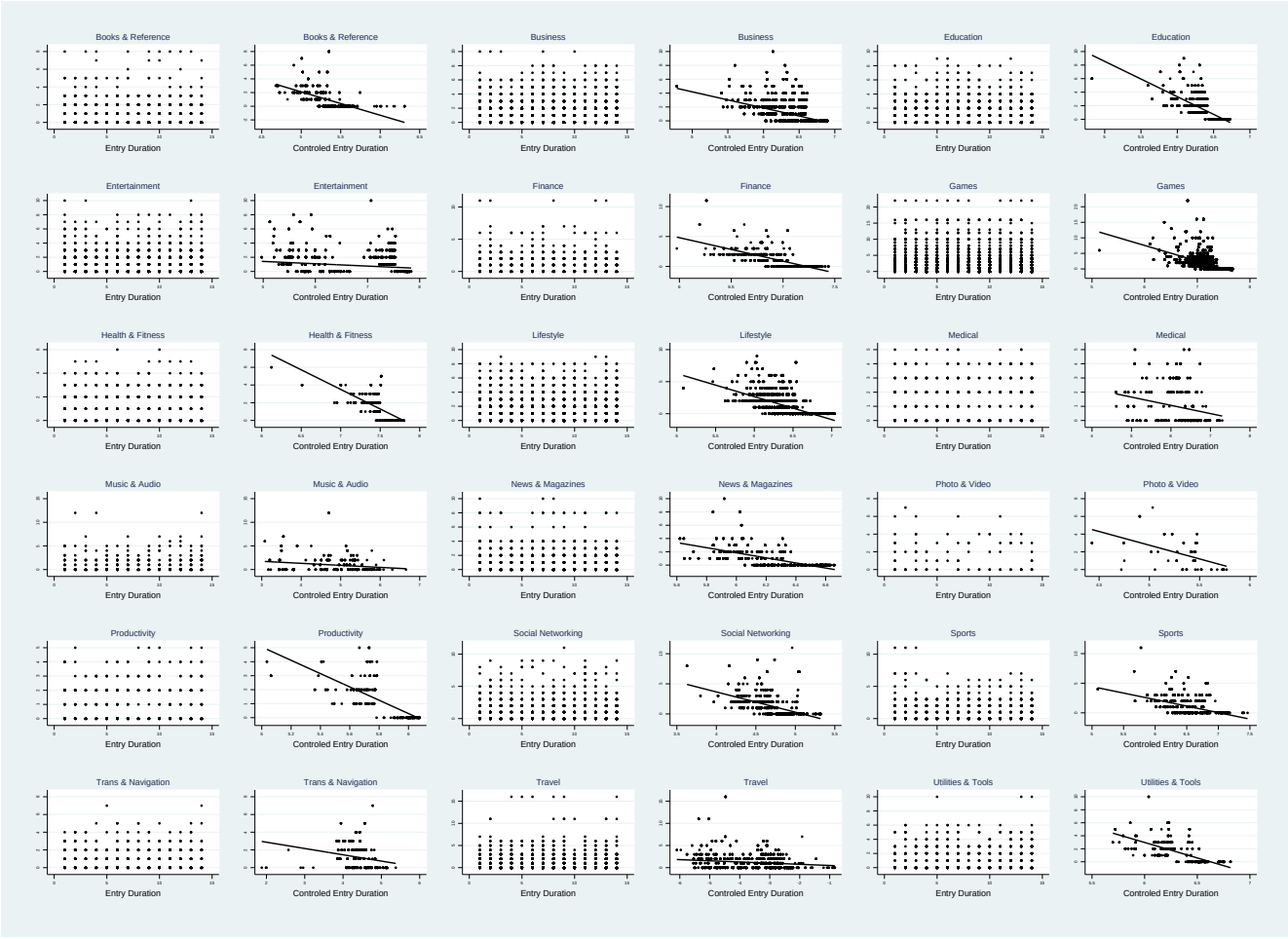


Figure 11: Weekly Incremental Ratings of Apps from Other Top50 Developers Following Top50 Developers within 2 weeks (Android)

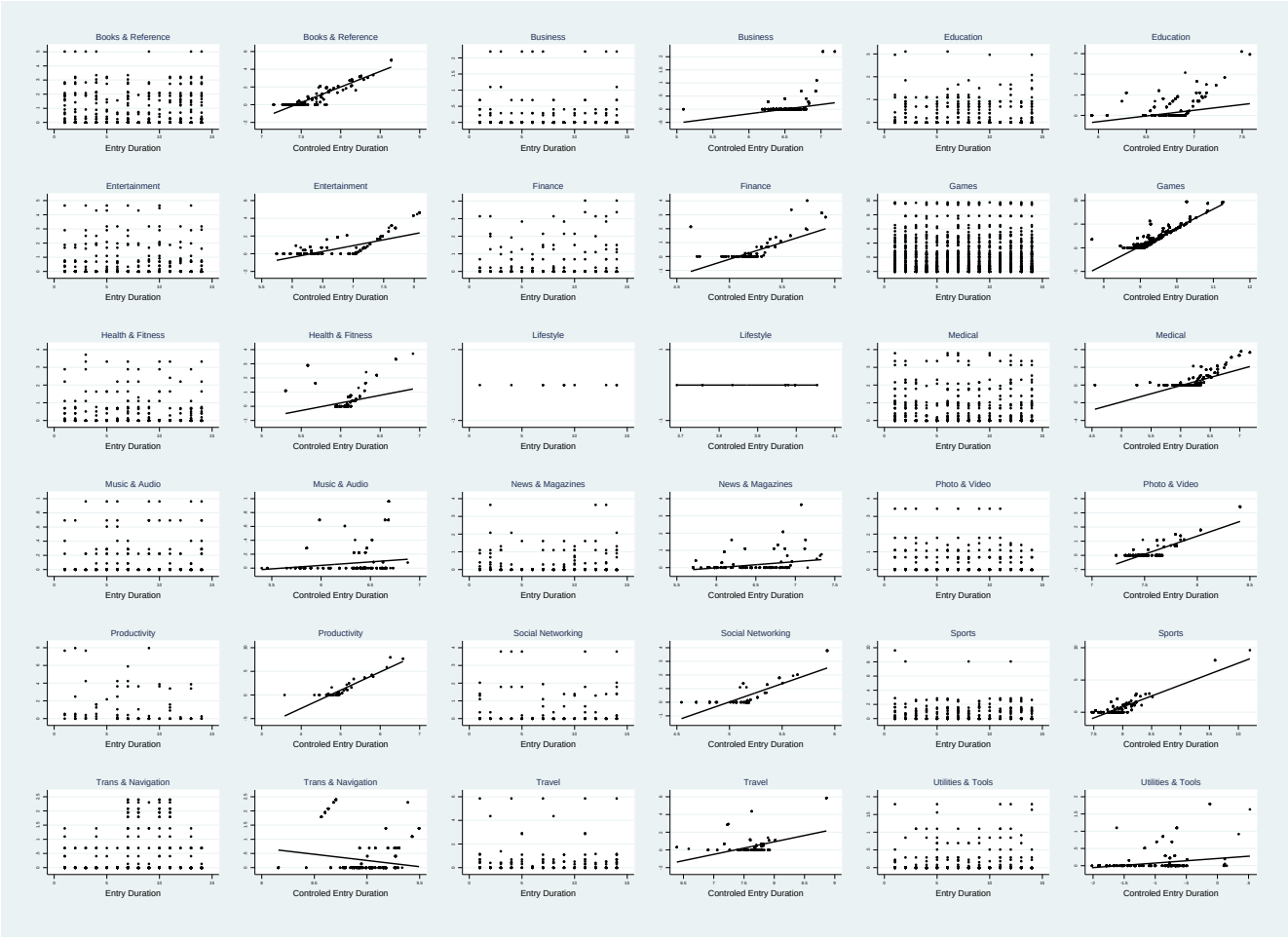


Figure 12: Monthly Major Updates of Apps from Other Top50 Developers Following Top50 Developers within 2 weeks (Android)

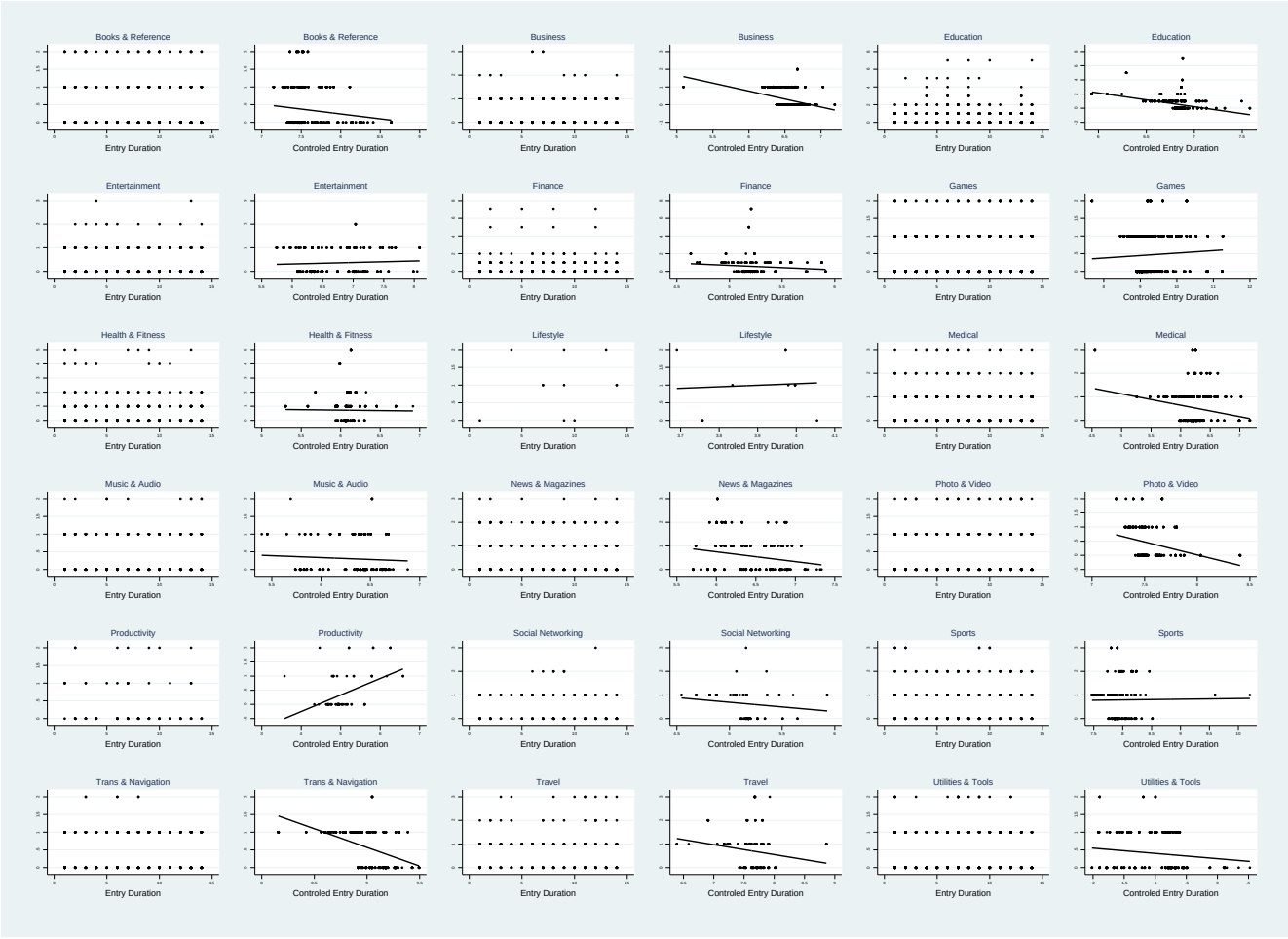


Figure 13: Monthly Minor Updates of Apps from Other Top50 Developers Following Top50 Developers within 2 weeks (Android)

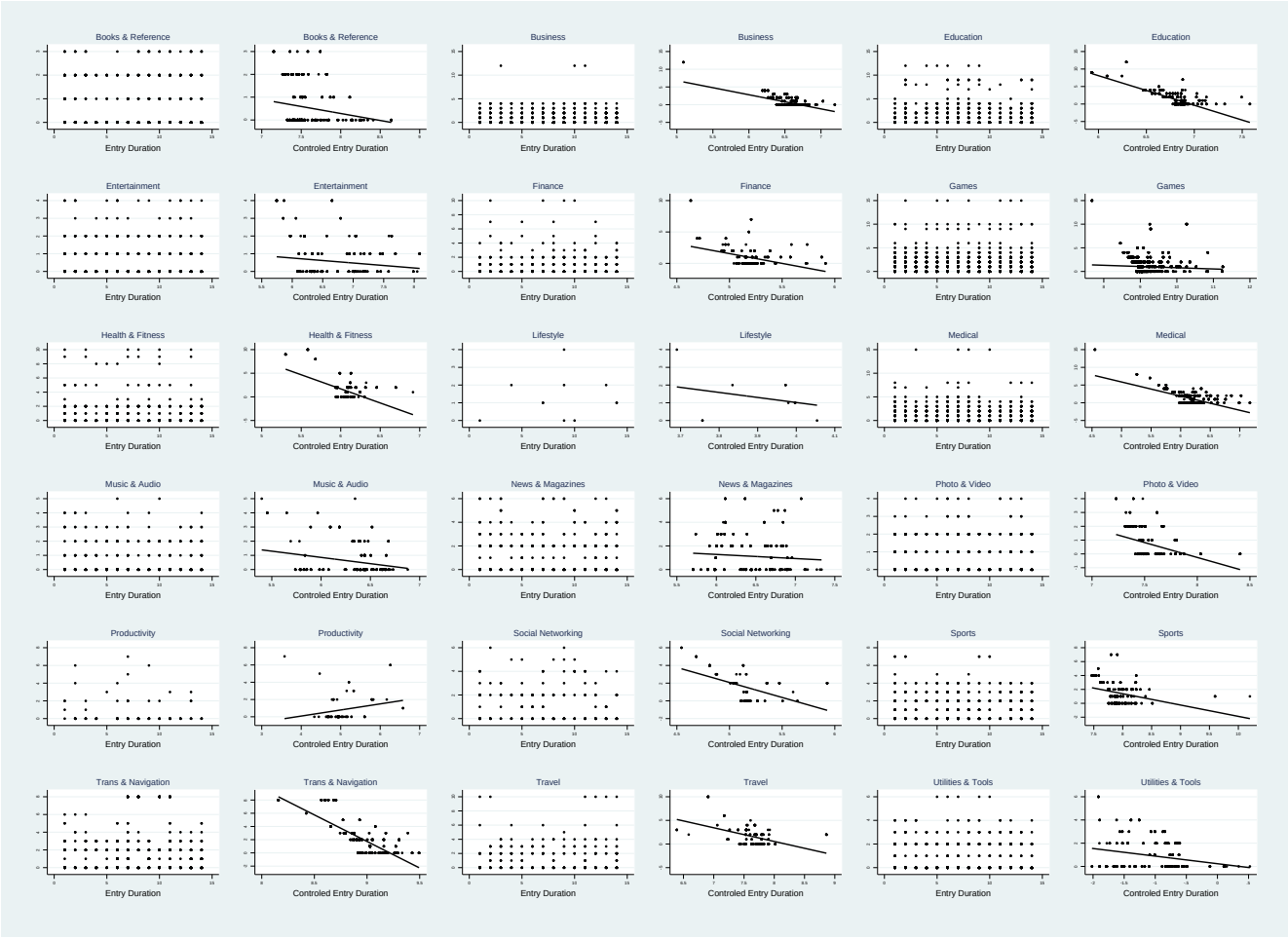


Figure 14: Weekly Incremental Ratings of Apps from Non Top50 Developers Following Top50 Developers within 2 weeks (iTunes)

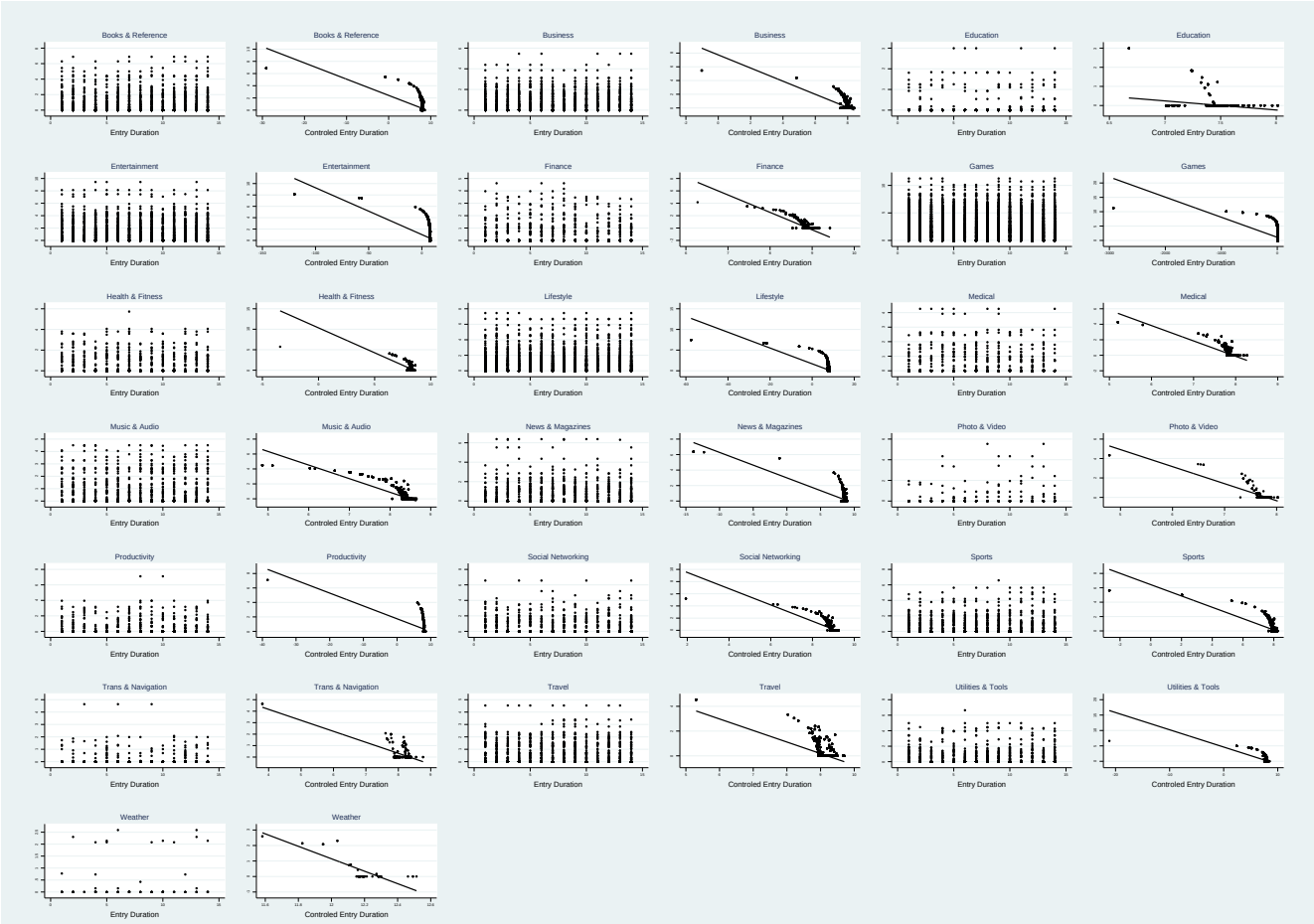


Figure 15: Monthly Major Updates of Apps from Non Top50 Developers Following Top50 Developers within 2 weeks (iTunes)



Figure 16: Monthly Minor Updates of Apps from Non Top50 Developers Following Top50 Developers within 2 weeks (iTunes)

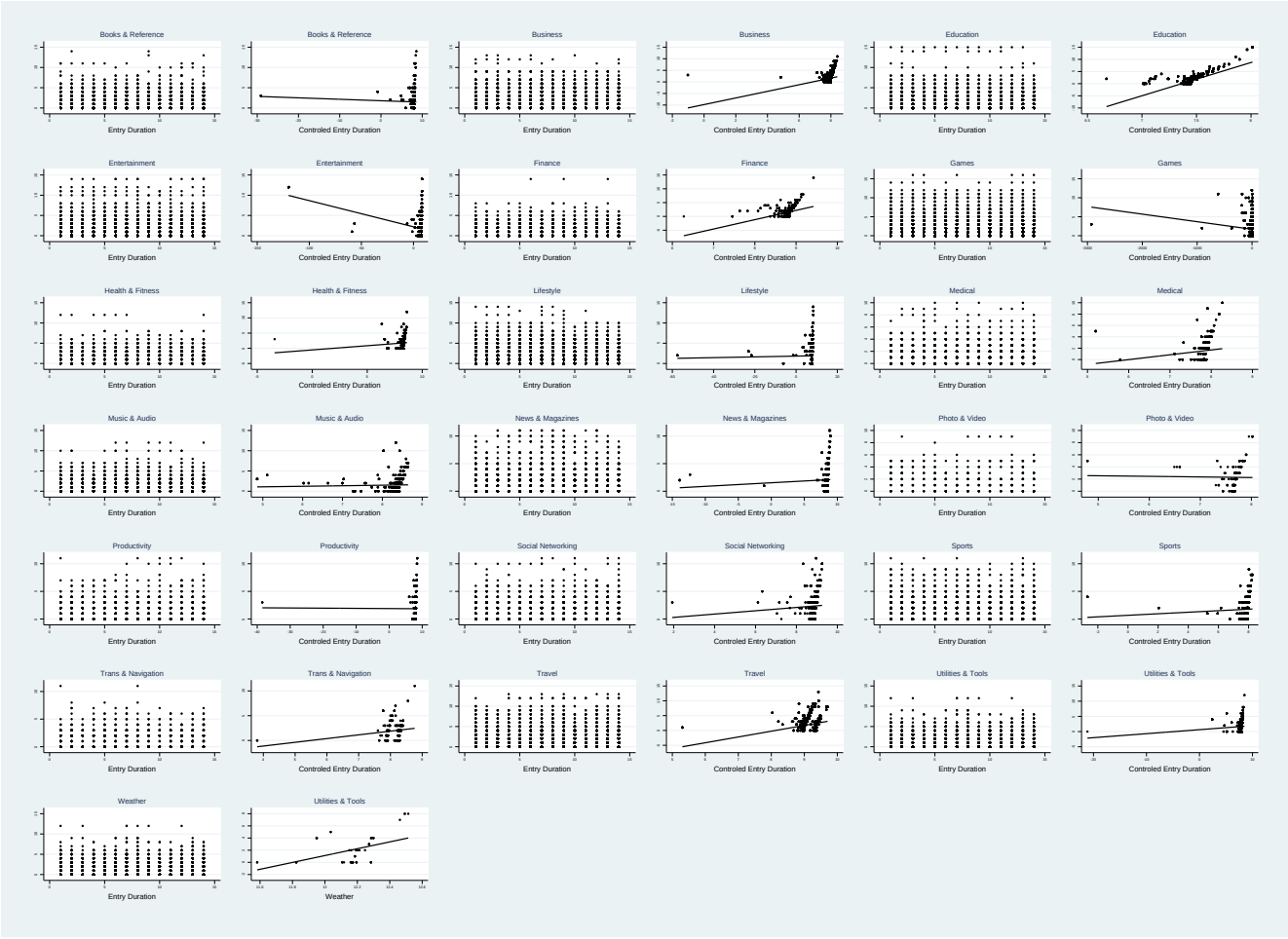


Figure 17: Weekly Incremental Ratings of Apps from Other Top50 Developers Following Top50 Developers within 2 weeks (iTunes)

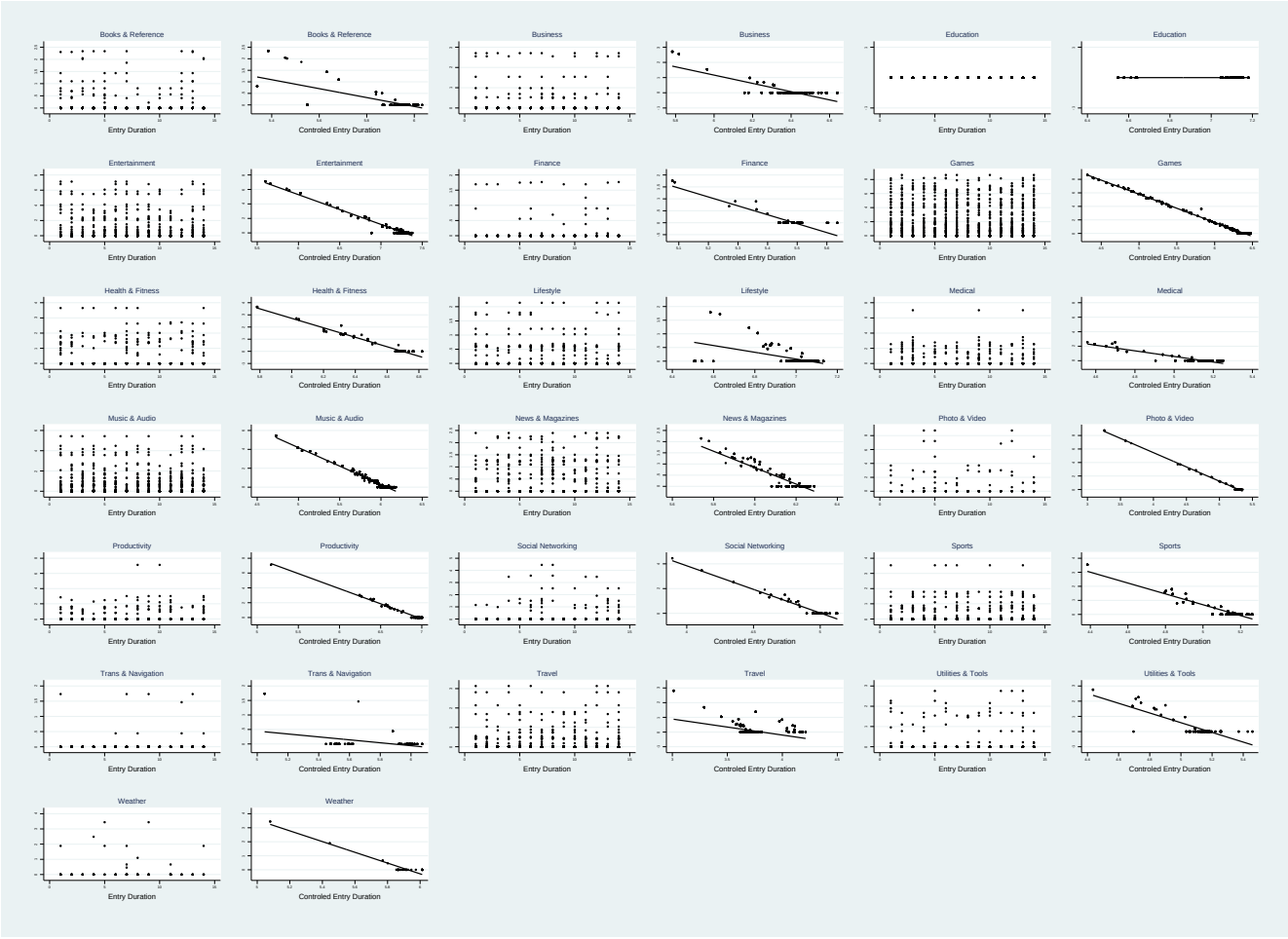


Figure 18: Monthly Major Updates of Apps from Other Top50 Developers Following Top50 Developers within 2 weeks (iTunes)

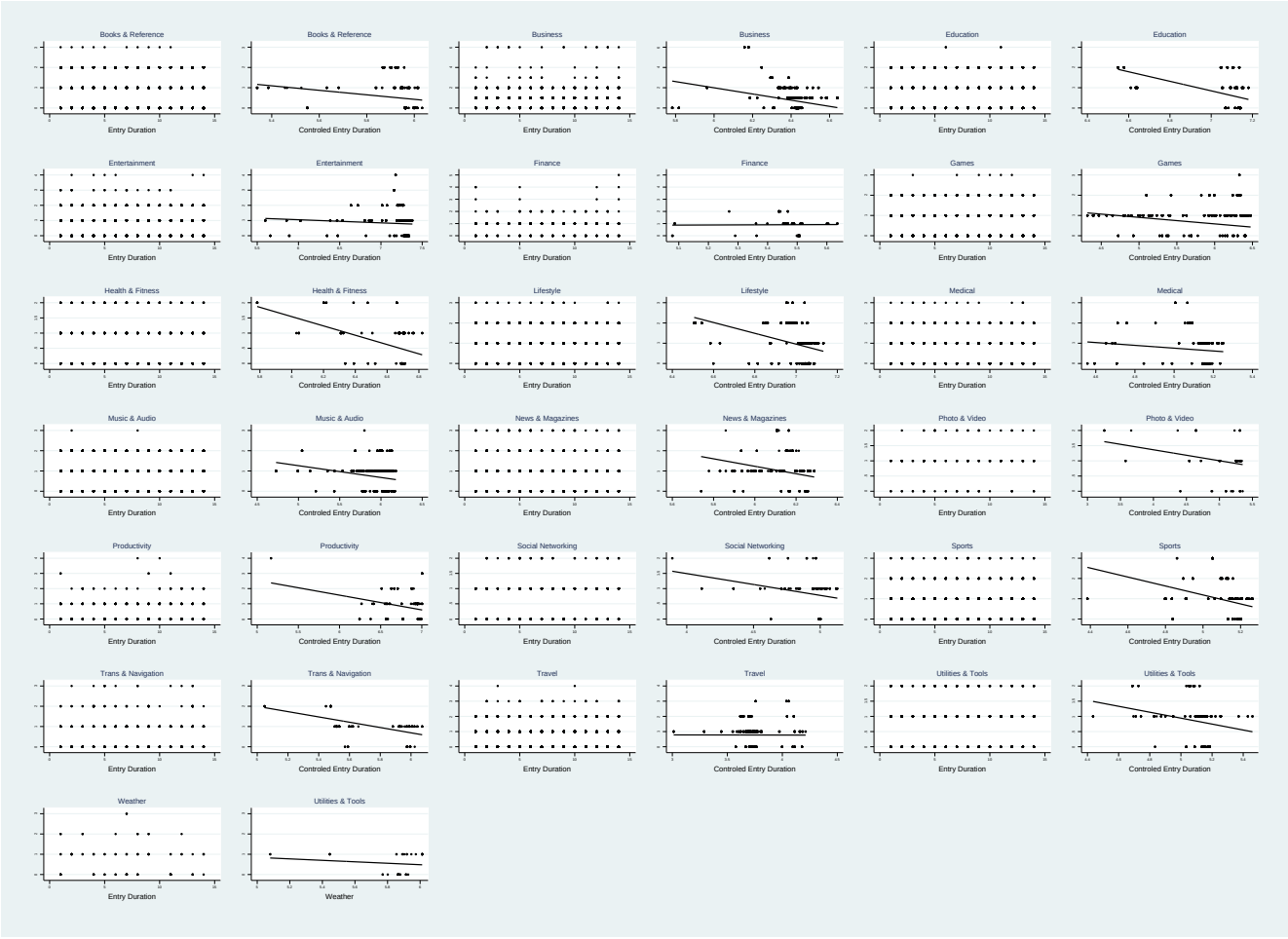


Figure 19: Monthly Minor Updates of Apps from Other Top50 Developers Following Top50 Developers within 2 weeks (iTunes)

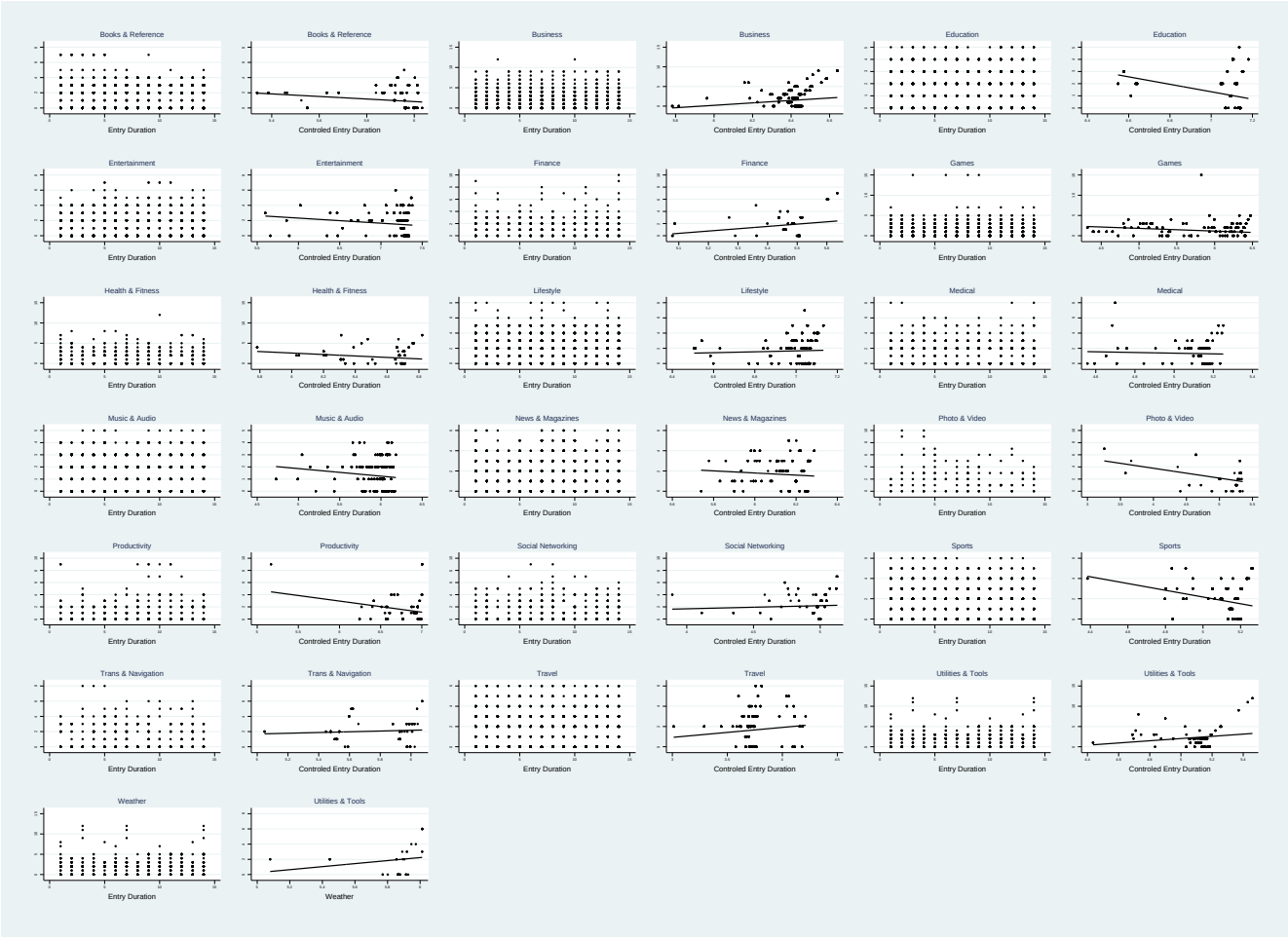


Table 1: Genre and Cross-Platform Apps Distribution(Android)

Genre	Genre Distribution		Percent of Cross-platform	Entry Order (%)			
	Number	Percent (%)	Apps (%)	Same Time	iTunes First	Android First	Unidentified
Books & Reference	44,660	7.86	8.88	1.74	17.20	59.52	21.54
Business	21,697	3.82	24.89	2.28	24.11	39.79	33.83
Education	29,951	5.27	15.80	1.58	22.59	45.20	30.62
Entertainment	72,468	12.76	7.27	1.63	21.29	47.83	29.25
Finance	12,450	2.19	22.85	1.69	19.93	40.91	37.47
Games	81,974	14.43	10.40	1.58	25.99	39.27	33.17
Health & Fitness	14,622	2.57	14.67	1.77	20.42	47.37	30.44
Lifestyle	44,460	7.83	17.26	1.33	21.04	44.57	33.06
Medical	7,472	1.32	24.20	1.71	20.91	45.80	31.58
Music & Audio	19,385	3.41	9.97	1.09	16.82	49.48	32.61
News & Magazines	17,865	3.15	19.91	1.49	16.00	45.09	37.42
Photo & Video	9,910	1.75	5.62	0.36	19.39	43.09	37.16
Productivity	15,152	2.67	10.64	1.30	19.85	45.78	33.06
Social Networking	30,159	5.31	10.68	1.27	22.10	39.35	37.27
Sports	21,293	3.75	13.60	1.90	20.66	43.32	34.13
Trans & Navigation	8,143	1.43	14.05	1.84	22.20	36.89	39.07
Travel	24,448	4.30	20.89	1.59	18.58	47.05	32.78
Utilities & Tools	88,552	15.59	2.34	1.16	21.87	47.32	29.65
Weather	3,244	0.57	11.53	0.80	14.44	36.63	48.13
Total	567,905	100.00	11.42	1.59	21.18	44.65	32.59

Table 2: Genre and Cross-platform App Distribution(iTunes)

Genre	Genre Distribution		Percent of Cross-platform	Entry Order (%)			
	Number	Percent (%)	Apps (%)	Same Time	iTunes First	Android First	Unidentified
Books & Reference	119,164	14.48	3.77	1.93	18.21	56.77	23.08
Business	47,694	5.79	12.70	1.92	23.34	39.99	34.75
Education	86,540	10.51	6.26	1.57	24.09	45.75	28.59
Entertainment	88,581	10.76	5.65	1.60	23.00	46.36	29.04
Finance	19,542	2.37	13.84	1.63	20.16	40.61	37.61
Games	142,332	17.29	5.70	1.65	25.74	39.08	33.53
Health & Fitness	21,675	2.63	9.29	1.94	22.35	44.66	31.05
Lifestyle	82,487	10.02	9.12	1.46	20.86	44.62	33.05
Medical	17,767	2.16	10.77	1.52	19.85	48.33	30.30
Music & Audio	33,661	4.09	6.84	1.39	15.51	52.04	31.06
News & Magazines	23,616	2.87	13.24	1.47	15.99	45.36	37.17
Photo & Video	19,753	2.40	3.32	0.61	23.32	39.79	36.28
Productivity	24,539	2.98	5.29	1.23	21.03	42.91	34.82
Social Networking	18,172	2.21	11.58	1.24	19.86	39.57	39.33
Sports	27,151	3.30	11.12	1.72	20.43	43.41	34.44
Trans & Navigation	14,146	1.72	7.66	1.29	16.97	43.54	38.19
Travel	42,618	5.18	10.20	1.61	18.20	46.25	33.94
Utilities & Tools	54,717	6.65	5.97	1.29	20.01	46.54	32.17
Weather	3,748	0.46	10.38	0.77	15.94	36.76	46.53
Total	823,068	100.00	7.30	1.59	21.18	44.65	32.59

Table 3: App Quality Cross Genre(Android)

Genre	Number of Ratings	Weekly Incrememtal number of Ratings	Monthly Major Updates	Monthly Minor Updates
Books & Reference	114.48	4.97	0.31	0.60
Business	54.63	1.88	0.41	0.84
Education	39.28	1.86	0.44	0.90
Entertainment	279.11	15.00	0.35	0.75
Finance	132.18	3.51	0.55	1.20
Games	842.17	52.40	0.46	1.05
Health & Fitness	167.49	5.78	0.39	0.88
Lifestyle	139.05	4.57	0.41	0.87
Medical	31.89	1.06	0.43	0.86
Music & Audio	408.12	19.53	0.41	0.89
News & Magazines	116.45	3.03	0.46	1.02
Photo & Video	327.25	14.74	0.42	1.00
Productivity	560.75	19.08	0.50	1.22
Social Networking	956.07	31.06	0.43	1.01
Sports	85.19	3.77	0.52	1.01
Trans & Navigation	179.49	5.23	0.70	1.50
Travel	224.73	4.61	0.48	0.94
Utilities & Tools	278.60	11.22	0.36	0.83
Weather	542.78	13.74	0.65	1.48
Total	339.88	16.06	0.42	0.91

Table 4: App Quality Cross Genre(iTunes)

Genre	Number of ratings	Weekly incremental number of Ratings	Monthly major updates	Monthly minor updates
Books & Reference	50.85	1.07	0.31	0.62
Business	30.72	0.72	0.58	1.27
Education	32.36	0.39	0.49	1.04
Entertainment	275.63	4.93	0.41	0.90
Finance	93.27	1.75	0.68	1.53
Games	718.98	20.60	0.43	0.98
Health & Fitness	149.90	3.74	0.52	1.17
Lifestyle	122.27	1.88	0.54	1.22
Medical	45.87	0.78	0.52	1.14
Music & Audio	206.51	4.02	0.58	1.19
News & Magazines	115.89	2.02	0.67	1.44
Photo & Video	203.94	4.78	0.56	1.43
Productivity	171.52	3.23	0.65	1.58
Social Networking	593.02	24.63	0.64	1.56
Sports	123.43	1.73	0.61	1.33
Trans & Navigation	115.66	1.48	0.61	1.39
Travel	49.69	0.75	0.58	1.25
Utilities & Tools	193.89	3.32	0.53	1.21
Weather	327.29	5.75	0.76	1.76
Total	226.02	5.58	0.49	1.10

Table 5: App Quality of Cross-platfrom Apps(Android)

Cross Entrants	Number of Ratings	Weekly Incrememtal number of Ratings	Monthly Major Updates	Monthly Minor Updates
Non Cross Entrants	273.42	13.67	0.40	0.87
Enter Simultaneously	69.54	3.97	0.40	0.80
Enter Android First	265.22	16.40	0.49	1.08
Enter iTunes First	382.72	19.56	0.47	0.93

Table 6: App Quality of Cross-platfrom Apps(iTunes)

Cross Entrants	Number of Ratings	Weekly incremental number of ratings	Monthly major updates	Monthly minor updates
Non Cross Entrants	185.37	4.22	0.47	1.03
Enter Simulatneously	15.21	0.46	0.61	1.23
Enter Android First	40.46	3.11	0.54	1.10
Entier iTunes First	701.49	17.39	0.90	2.05

Table 7: Apps from Non Top50 Developers Following Top50 Developers within 2 weeks

Genre	Android						iTunes					
	Median			Mean			Median			Mean		
	I	II	III	I	II	III	I	II	III	I	II	III
Books & Reference	-0.19	0.00	0.00	-4.47	0.01	0.08	0.00	1.00	1.00	2.14	0.40	0.73
Business	-0.06	0.00	0.00	-3.39	0.01	-0.04	0.00	1.00	1.00	-0.26	0.25	0.40
Education	-0.17	0.00	0.00	-1.26	-0.03	-0.06	0.00	1.00	1.00	-0.37	0.22	0.42
Entertainment	-0.42	0.00	0.00	-11.15	0.01	-0.08	0.00	1.00	1.00	10.37	0.34	0.67
Finance	-0.20	0.00	0.00	-1.14	0.00	-0.25	0.00	0.00	1.00	-0.44	0.24	0.21
Games	-0.39	0.00	0.00	-11.55	-0.01	0.00	0.00	1.00	1.00	106.65	0.26	0.56
Health & Fitness	-0.18	0.00	0.00	-4.37	0.00	-0.16	0.00	1.00	2.00	-1.81	0.25	0.50
Lifestyle	-0.22	0.00	0.00	-4.32	0.15	0.15	0.00	1.00	2.00	1.61	0.32	0.63
Medical	-0.22	0.00	0.00	-4.51	0.00	0.01	0.00	1.00	1.00	-0.12	0.22	0.38
Music & Audio	-0.09	0.00	0.00	2.11	0.04	0.06	0.00	1.00	1.00	-1.42	0.19	0.31
News & Magazines	-0.43	0.00	0.00	-25.29	-0.01	-0.10	0.00	1.00	2.00	0.50	0.27	0.65
Photo & Video	0.25	1.00	1.00	21.21	0.19	0.64	0.00	1.00	2.00	-1.19	0.47	0.77
Productivity	-0.48	0.00	0.00	-14.29	0.07	-0.08	0.00	0.00	0.00	0.29	0.14	0.24
Social Networking	-0.27	0.00	0.00	-11.86	0.08	-0.10	0.00	1.00	2.00	-0.45	0.31	0.56
Sports	-0.26	0.00	0.00	-20.13	0.04	-0.08	0.00	1.00	2.00	0.38	0.22	0.52
Trans & Navigation	-0.20	1.00	1.00	-2.75	-0.02	-0.02	0.00	1.00	2.00	-0.49	0.27	0.50
Travel	-0.30	-1.00	-1.00	-2.64	-0.27	-0.68	0.00	1.00	2.00	-0.14	0.28	0.64
Utilities & Tools	-0.13	0.00	0.00	3.62	0.01	0.11	0.00	1.00	2.00	-1.81	0.25	0.48
Weather	-	-	-	-	-	-	0.00	0.00	1.00	-4.55	0.22	0.38

Table 8: Apps from other Top50 Developers Following Top50 Developers within 2 weeks

Genre	Android						iTunes					
	Median			Mean			Median			Mean		
	I	II	III	I	II	III	I	II	III	I	II	III
Books & Reference	-0.19	0.00	0.00	-4.03	-0.02	-0.18	0.00	0.00	0.00	-0.72	0.37	0.41
Business	-0.06	0.00	0.00	-3.46	-0.21	-0.44	0.00	1.00	1.00	-0.55	0.15	0.23
Education	-0.17	0.00	0.00	-1.79	0.06	0.16	0.00	0.00	0.00	-0.40	0.17	0.07
Entertainment	-0.42	0.00	0.00	-12.31	-0.09	-0.36	0.00	0.00	0.00	6.06	0.27	0.37
Finance	-0.20	0.00	0.00	-1.66	0.02	-0.33	0.00	0.00	1.00	-1.08	0.23	0.23
Games	-0.39	0.00	0.00	215.93	-0.05	-0.19	0.00	0.00	0.00	32.32	0.11	0.02
Health & Fitness	-0.18	0.00	0.00	-5.82	0.24	0.26	0.00	0.00	0.00	-1.87	0.08	0.22
Libraries & Demo	-0.22	1.00	1.00	-5.97	0.67	0.49	0.00	1.00	2.00	-1.51	0.39	0.59
Lifestyle	-0.22	0.00	0.00	-4.11	0.07	0.14	0.00	1.00	2.00	3.70	0.28	0.45
Medical	-0.09	0.00	0.00	-1.35	-0.14	-0.31	0.00	1.00	2.00	-0.78	0.38	0.45
Music & Audio	-0.43	0.00	0.00	-26.64	0.11	0.08	0.00	1.00	2.00	-1.49	0.29	0.55
News & Magazines	-0.25	0.00	0.00	-2.60	-0.07	-0.42	0.00	1.00	2.00	189.32	0.41	1.02
Photo & Video	-0.31	0.00	0.00	115.83	-0.11	-0.30	0.00	0.00	0.00	6.27	0.13	-0.02
Productivity	-0.27	0.00	0.00	-16.18	0.02	-0.02	0.00	1.00	2.00	-2.79	0.19	0.37
Social Networking	-0.26	1.00	1.00	39.16	0.31	0.35	0.00	1.00	2.00	-1.13	0.20	0.33
Sports	-0.20	1.00	1.00	-2.99	-0.02	0.61	0.00	1.00	2.00	-1.04	0.25	0.27
Trans & Navigation	-0.30	0.00	0.00	5.28	-0.12	-0.19	0.00	0.00	0.00	-0.33	-0.11	-0.20
Travel	-0.13	0.00	0.00	-6.01	-0.13	-0.20	0.00	1.00	2.00	-2.75	0.35	0.88
Utilities & Tools	-0.30	0.00	0.00	-11.08	0.14	0.18	0.00	0.00	1.00	-4.58	0.01	0.20
Weather	-0.60	1.00	1.00	-14.86	-0.11	-0.21	0.00	0.00	0.00	-0.72	0.37	0.41

Table 9: Model selection of the prediction of cross-entry

Variables	Android	iTunes
Duration between Two Entries	x	x
The number of Rating	x	x
Weekly Incremental Number of Rating		x
Rating Star	x	x
Average Price	x	x
Frequency of Major Updates		x
Frequency of Minor Updates	x	
Average Weekly Number of Rating	x	x
Variance of Weekly Rating Number	x	x
Average of 2-week Rating Incremental	x	
Variance of 2-week Rating Incremental	x	x
Variation Coefficient of Daily Rating Number	x	x
Variation Coefficient of 2-week Rating Incremental	x	x
Thu Number of Sibling Apps	x	x
Max Number of Rating of Sibling Apps		x
Average Number of Rating of Sibling Apps		
If Have Other Cross-Platform Apps	x	x
The Total Number of Apps in Specific Android Genre		x
The Total Number of Apps in Specific iTunes Genre	x	
The Total Numberof Apps in Android		x
The Total Numberof Apps in iTunes	x	
HHI in Android Genre		x
HHI in iTunes Genre	x	

Table 10: Android Regression of Cox proportional hazard model

Variable	Entire Apps	Books & Reference	Business	Education	Entertainment	Finance	Games	Health & Fitness	Lifestyle	Medical	Music & Audio	News & Magazines	Photo & Video	Productivity	Social Networking	Sports	Trans & Navigation	Travel	Utilities & Tools	Weather
Ln(number of ratings)	-0.439	-0.442	-0.66	-0.483	-0.485	-0.682	-0.36	-0.613	-0.523	-0.594	-0.601	-0.527	-0.37	-0.847	-0.483	-0.403	-0.523	-0.445	-0.416	-0.292
	(0.008)***	(0.058)***	(0.053)***	(0.040)***	(0.033)***	(0.048)***	(0.016)***	(0.073)***	(0.030)***	(0.092)***	(0.086)***	(0.049)***	(0.116)***	(0.069)***	(0.038)***	(0.046)***	(0.083)***	(0.042)***	(0.047)***	(0.129)**
Ln(weekly incremental number of ratings)	0.552	1.012	1.124	0.742	0.742	0.663	0.452	0.926	0.729	0.912	0.942	0.618	0.611	1.239	0.639	0.49	0.837	0.652	0.72	0.22
	(0.010)***	(0.088)***	(0.092)***	(0.071)***	(0.045)***	(0.064)***	(0.017)***	(0.120)***	(0.043)***	(0.167)***	(0.131)***	(0.065)***	(0.167)***	(0.103)***	(0.052)***	(0.056)***	(0.134)***	(0.067)***	(0.064)***	-0.204
Monthly Major Updates	-1.456	-3.09	-2.031	-1.145	-2.021	-1.056	-1.922	-1.213	-1.417	-2.629	-0.372	-2.929	-0.64	-1.298	-1.216	-1.887	-2.332	-1.722	-2.254	-2.56
	(0.070)***	(1.018)***	(0.291)***	(0.191)***	(0.338)***	(0.304)***	(0.184)***	(0.380)***	(0.209)***	(1.018)***	-0.346	(0.571)***	-0.781	(0.556)**	(0.275)***	(0.442)***	(0.592)***	(0.352)***	(0.545)***	(1.145)**
Monthly Minor Updates	-0.226	-0.38	-0.169	-0.178	-0.358	-0.239	-0.187	0.028	-0.09	-0.516	-0.862	0.062	-0.614	-0.455	-0.213	-0.287	-0.008	-0.244	-0.354	-0.015
	(0.017)***	(0.130)***	(0.055)***	(0.069)***	(0.095)***	(0.104)**	(0.034)***	-0.101	(0.054)*	(0.175)***	(0.136)***	-0.103	(0.234)***	(0.221)**	(0.074)***	(0.050)***	-0.154	(0.102)**	(0.128)***	-0.253
Predicted cross entry probability	4.073	4.788	3.797	3.848	4.339	3.761	3.879	4.388	3.791	3.733	4.444	3.363	5.106	3.952	3.895	3.814	3.544	3.766	5.565	4.337
	(0.016)***	(0.080)***	(0.060)***	(0.055)***	(0.059)***	(0.080)***	(0.041)***	(0.104)***	(0.050)***	(0.104)***	(0.108)***	(0.078)***	(0.202)***	(0.108)***	(0.075)***	(0.081)***	(0.109)***	(0.068)***	(0.102)***	(0.270)***
n_major_postentry_cross	2.068	4.261	2.365	1.308	3.075	1.605	2.653	1.997	2.291	2.766	0.118	4.463	2.299	0.715	2.214	3.14	2.851	2.255	2.984	4.843
	(0.079)***	(1.068)***	(0.323)***	(0.234)***	(0.408)***	(0.386)***	(0.202)***	(0.466)***	(0.262)***	(1.047)***	-0.437	(0.696)***	(1.031)**	-0.577	(0.304)***	(0.482)***	(0.655)***	(0.360)***	(0.591)***	(1.603)***
n_minor_postentry_cross	0.559	0.327	0.58	0.734	0.724	0.896	0.439	-0.092	0.261	1.08	1.759	-0.422	0.516	1.44	0.574	0.4	0.187	0.536	1.01	-0.662
	(0.032)***	-0.258	(0.104)***	(0.134)***	(0.141)***	(0.203)***	(0.062)***	-0.218	(0.116)**	(0.235)***	(0.289)***	-0.258	-0.469	(0.255)***	(0.133)***	(0.118)***	-0.236	(0.141)***	(0.211)***	-0.634
N	402,276	30,533	13,224	20,791	51,935	8,248	58,691	10,014	30,405	4,763	13,437	11,524	7,185	11,646	21,392	13,828	5,657	16,146	70,536	2,321

* p¡0.1; ** p¡0.05; *** p¡0.01

Table 11: iTunes Regression of Cox proportional hazard model

Variable	Entire Apps	Books & Reference	Business	Education	Entertainment	Finance	Games	Health & Fitness	Lifestyle	Medical	Music & Audio	News & Magazines	Photo & Video	Productivity	Social Networking	Sports	Trans & Navigation	Travel	Utilities & Tools	Weather
Ln(number of ratings)	-0.47	-0.697	-18.829	-1.275	-0.518	-25.341	-0.139	-19.466	-18.933	-18.856	-0.998	-0.996	-0.371	-0.766	-0.913	-0.84	-16.904	-19.725	-0.842	-24.524
	(0.018)***	(0.189)***	-3,679,631.01	(0.204)***	(0.065)***	-1689763747	(0.017)***	-9,108,248.63	-7,177,634.02	-6,279,357.73	(0.196)***	(0.177)***	(0.131)***	(0.140)***	(0.168)***	(0.134)***	-1,098,767.23	-9,063,743.44	(0.142)***	-1725479487
Ln(weekly incremental number of ratings)	-0.164	-0.142	-0.649	0.395	-0.168	-0.391	-0.104	-0.222	-0.307	-0.23	-0.272	-0.267	-0.109	-0.37	-0.124	-0.253	-0.672	-0.463	-0.212	0.28
	(0.008)***	(0.042)***	(0.062)***	(0.067)***	(0.026)***	(0.060)***	(0.013)***	(0.052)***	(0.094)***	(0.080)***	(0.047)***	(0.058)***	(0.059)*	(0.064)***	(0.049)**	(0.048)***	(0.103)***	(0.067)***	(0.051)***	(0.086)***
Monthly Major Updates	-1.21	-0.832	-1.423	-1.032	-1.105	-1.321	-1.515	-1.278	-1.264	-2.386	-2.001	-1.596	-0.872	-1.543	-0.949	-1.111	-1.046	-1.497	-1.564	-1
	(0.042)***	(0.172)***	(0.127)***	(0.163)***	(0.153)***	(0.205)***	(0.122)***	(0.257)***	(0.255)***	(0.376)***	(0.238)***	(0.183)***	(0.274)***	(0.242)***	(0.177)***	(0.177)***	(0.249)***	(0.150)***	(0.196)***	(0.531)*
Monthly Minor Updates	-0.159	-0.058	-0.198	-0.4	-0.106	-0.39	-0.107	-0.28	-0.203	-0.16	0.21	-0.07	-0.031	0.114	-0.186	-0.284	-0.244	-0.022	0.041	-0.572
	(0.011)***	-0.045	(0.030)***	(0.061)***	(0.041)***	(0.069)***	(0.023)***	(0.068)***	(0.085)**	(0.067)**	(0.047)***	-0.048	-0.075	(0.056)**	(0.056)***	(0.054)***	(0.070)***	-0.04	-0.038	(0.146)***
Predicted cross entry probability	4.143	5.979	3.645	3.971	4.522	3.492	4.134	3.407	3.844	3.288	4.588	3.452	5.605	3.923	3.398	3.623	4.022	4.262	4.787	3.048
	(0.013)***	(0.058)***	(0.040)***	(0.040)***	(0.049)***	(0.063)***	(0.039)***	(0.067)***	(0.069)***	(0.066)***	(0.075)***	(0.061)***	(0.164)***	(0.095)***	(0.075)***	(0.059)***	(0.111)***	(0.055)***	(0.062)***	(0.164)***
n_major_postentry_cross	1.09	0.771	1.202	0.04	0.898	1.471	1.679	1.258	1.065	3.125	2.054	1.447	0.944	2.489	1.737	1.263	1.328	1.276	1.536	-0.443
	(0.051)***	(0.231)***	(0.155)***	-0.186	(0.181)***	(0.275)***	(0.140)***	(0.298)***	(0.287)***	(0.417)***	(0.292)***	(0.233)***	(0.382)**	(0.310)***	(0.219)***	(0.232)***	(0.333)***	(0.172)***	(0.237)***	-0.671
n_minor_postentry_cross	0.703	0.506	0.746	1.469	0.649	0.711	0.642	0.867	0.857	0.28	-0.017	0.62	0.41	-0.177	0.411	0.786	0.821	0.365	0.4	2.3
	(0.023)***	(0.115)***	(0.067)***	(0.096)***	(0.077)***	(0.154)***	(0.053)***	(0.122)***	(0.142)***	(0.139)**	-0.123	(0.117)***	(0.183)**	-0.152	(0.105)***	(0.113)***	(0.169)***	(0.076)***	(0.099)***	(0.318)***
N	792,374	115,958	44,088	83,152	85,313	17,837	137,398	20,458	23,626	16,569	32,150	21,619	19,349	23,686	16,772	25,255	13,388	39,697	52,585	3,474

* p|0.1; ** p|0.05; *** p|0.01