

A/B-Test of Retention and Monetization Using the Cox Model

Markus Viljanen, Antti Airola, Jukka Heikkonen, Tapio Pahikkala

Department of Future Technologies, University of Turku, Finland

majuvi@utu.fi, ajairo@utu.fi, jukhei@utu.fi, aatapa@utu.fi

Abstract

A/B testing is a popular tool for guiding mobile game development. The developer releases different versions of a game to different test cohorts, and observes which version has the best player retention or monetization. Correctly determining whether the differences are statistically significant is however challenging. Typically the analysis needs to be done on small and heterogeneous player cohorts, with differing follow-up times and unknown player churn. In this paper, we show for the first time how these issues can be properly addressed using the Cox model for recurrent events. The method enables a multivariate A/B-test, that allows determining which game version has the highest player retention or purchase rate, with confidence intervals provided. We demonstrate the benefits of the approach in multiple game development problems, on real-world free-to-play mobile game data.

Introduction

Game analytics has become an important tool in developing successful games (El-Nasr, Drachen, and Canossa, 2013), with modern analytics platforms making gathering of player data remotely easy. For free-to-play mobile game developers, this has led to an iterative game development model based on A/B-testing. When testing new features for a game, two or more variants of the game are distributed to different user cohorts, acquired for example via targeted advertisements on social media. By comparing the player behavior in these cohorts, the developer aims to determine which variant of the game is the most engaging, guiding thus further development of the game. In this work, we consider two metrics of engagement, retention rate and the purchase rate, which characterize the amount of time and money spent by players on a game. Retention rate is one of the most popular industry statistics (Seufert, 2014) and thus a natural choice for analysis. Purchase rate directly measures the number of purchases made by users, but its usability is limited to game versions that already have working monetization mechanisms implemented.

In order to implement A/B-testing properly, game developers require methods that allow estimating these metrics reliably from player data, as well as determining whether differences in metrics between different game variants are sta-

tistically significant, or simply due to random chance. This is made challenging due to a number of characteristics of the player data:

1. **Sample size:** it is not feasible for small developers to obtain large test cohorts for each A/B-test, since each player obtained via targeted advertising costs money.
2. **Limited time window:** player activity data is available only until the date of analysis. If players are obtained over time, the length of data available for different players may range from weeks to only day or two.
3. **Churn uncertainty:** it is unknown whether any given player has churned (i.e. permanently quit the game) after the last known session, or will return in the future.
4. **Confounding variables:** differences in user cohorts in terms of variables such as operating system, country etc. may bias the analysis, if not controlled for.

While game analytics has become a subject of significant interest in recent years in the academic game research literature, we are not aware of a proposed method that would address all these issues. Recently, methods from the field of survival analysis, such as the classical Cox regression model, have been used in game analytics to analyze player churn (Allart et al., 2016; Perianez et al., 2016). However, the practical applicability of the standard Cox model is limited, since fitting the model requires knowing which users have already churned. Further, the classical version of the method is used to model time to a single event such as churn or first purchase made, rather than session or purchase rates.

In this work we propose using recurrent event Cox regression (Cook and Lawless, 2007) to model retention and purchase rates. The method can naturally handle the limited and differing follow-up times for players, is not affected by churn uncertainty, allows control for confounding variables, and provides confidence intervals for the analyses. The method allows determining whether one game version has significantly better retention or purchase rate than the other. Further, the method has a simple interpretation and is straightforward to apply, for example using the statistical packages available for the R-language (Moore, 2016). We provide numerous examples on how different types of A/B-tests may be performed using the method on data from a real in-development free-to-play mobile game.

Duration and rate based metrics

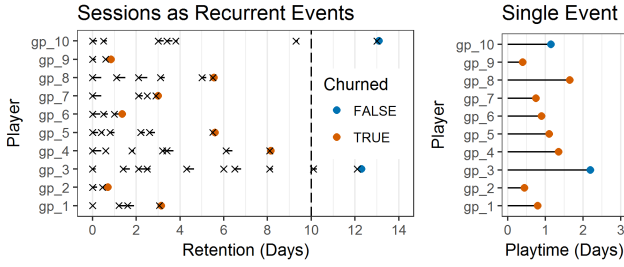


Figure 1: Time-lines for 10 hypothetical players followed for 14 days. Session starts are denoted with crosses, and lengths with line segments. A circle denotes the last session available in the data. On the right, we have summed together the session times of each player, resulting in the total playtime over this 14 days period.

In order to illustrate different types of player retention related metrics and prediction problems, we present 10 player time-lines in Figure 1. In this example the considered events are player sessions, but purchases can be analyzed analogously. There are two ways to aggregate the data, resulting in *duration* and *rate* based metrics.

Duration based metrics

Duration based metrics measure the sum of the variable of interest, such as the number of sessions, until the player churns. The obvious problem with this approach is that excluding subscription based games users do not tell when they have churned. Rules based on player inactivity have been used to impute churn status to data (see e.g. Hadiji et al. (2014); Runge et al. (2014); Tamassia et al. (2016)), but these can introduce biases in analyses.

Playtime, measures the total time (e.g. hours) a player spends playing a game before churning. Previously, Sifa, Bauckhage, and Drachen (2014) have analyzed the distribution of playtimes on Steam, whereas Allart et al. (2016); Viljanen et al. (2017) used survival analysis methods for playtime analysis. *Days active*, denotes the number of days from installing the game until a player churns. Recently, Periañez et al. (2016) has proposed a survival ensemble model for predicting number of active days. *Churn classification*, refers to classifying whether a player will churn from the game within some specified time range in the future (see e.g. Chen, Huang, and Lei (2009); Hadiji et al. (2014); Runge et al. (2014); Drachen et al. (2016); Periañez et al. (2016); Sifa et al. (2016); Tamassia et al. (2016) for proposed classification methods). Classification and regression of customer purchases was considered by Sifa et al. (2015). Metrics such as session times (Chen, Huang, and Lei, 2009), total number of sessions (Weber, Mateas, and Jhala, 2011), gates cleared (Debeauvais and Lopes, 2015; Isaksen, Gopstein, and Nealen, 2015), and several other activity measures (Feng, Brandt, and Saha, 2007; Tarnag, Chen, and Huang, 2008) have also been considered.

Rate based metrics

Rate based metrics measure the rate at which a given event occurs over each unit of time. *Retention rate*, or simply retention, is a very popular metric in the game industry (El-Nasr, Drachen, and Canossa, 2013; Seufert, 2014), that is implemented in most major game analytics platforms. Given a cohort of users starting on day 0, the (discrete) retention rate on day k denotes the fraction of players returning to play on the k :th day. In Figure 1, the retention rate corresponds to summing the player sessions over the y-axis. For example, retention rate for day 7 is 20%, since two players play on the seventh day. The retention rate combines both longevity and frequency of play in one metric. Analogously to retention, we can analyze *purchase rate* denoting the fraction of players making a purchase over some time period.

The advantage of rate based metrics is that computing them does not require any information beyond the last observed date in the data, such as which players have already churned, or how far in the future current active players will churn. Despite its popularity in the industry, retention rate has not been much studied in academic game analytics literature, with some exceptions (Viljanen et al., 2016). Recurrent event survival analysis has however been used for example in medicine and reliability engineering to analyze rate based statistics such as frequencies of asthma attacks or machine faults (Cook and Lawless, 2007).

Recurrent event analysis of rate metrics

Session and Purchase Rates

Let $N(t)$ denote the number of events, i.e. sessions or purchases, in the interval $(0, t]$ for a player. Here, 0 denotes the time when a player first starts playing the game, and t can be measured for example in seconds, hours or days. Further, let $\Delta N(t)$ denote the number of events in a very small interval $[t, t + \Delta t)$. The rate function, that gives the instantaneous probability of an event at time t is defined as the limit
$$\rho(t) = \lim_{\Delta t \rightarrow 0} \frac{P[\Delta N(t)=1]}{\Delta t}.$$

By integrating the rate function on interval $(0, t]$, one obtains the expected cumulative number of events up to time t (Cook and Lawless, 2007), called the mean function of recurrent events $\mu(t) = E[N(t)] = \int_0^t \rho(u) du$. Given a real player cohort, we can estimate $\mu(t)$ non-parametrically as follows. Let $\{t_1, \dots, t_m\}$ denote the distinct times at which one or more events (i.e. session or purchase) happened. The follow-up times of players differ, as some may have been followed for weeks, while others may have only recently installed the game. The number of players who have been observed at least until time t_i is denoted as r_i . Finally, the number of events at time t_i (i.e. distinct players starting a session or making a purchase), is denoted as d_i . The Nelson-Aalen (NA) estimator $\mu_{NA}(t) = \sum_{t_i \leq t} \frac{d_i}{r_i}$ provides an unbiased estimate of the mean function (Cook and Lawless, 2007).

In Figure 2, we plot empirical estimates of session rate $\rho(t)$ and mean cumulative number of sessions $\mu(t)$ for in-development mobile game, where an A/B-test comparing three variants of the game was performed. The session rate is high in the beginning when players have just started, but

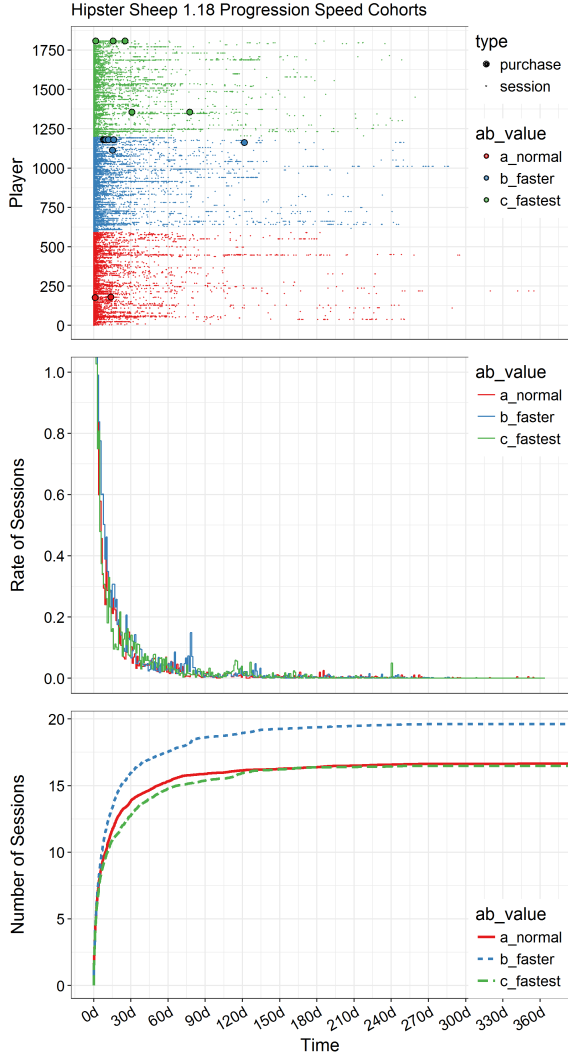


Figure 2: Session and purchase data from a mobile game, with three different randomly assigned progression speed cohorts. Player time-lines of events (top), session rates (middle), and mean number of sessions (bottom).

soon drops steeply as they lose interest and churn out. Correspondingly, $\mu(t)$ rises at first steeply, flattening out when most players have churned. The long tail corresponds to few dedicated players, who play up until a year. Comparing the cumulative session plots would suggest that out of the three game versions ‘b.faster’ has the highest player retention. Next, we introduce a statistical model that allows us to test this intuition.

The Cox Model as an A/B-test

The Cox proportional hazards model is a simple and robust regression model for rate functions (Cook and Lawless, 2007). It allows us to estimate from data the effect explanatory variables such as game version, or country of residence, have on the rate function. The model does not assume that the rate function would take any specific para-

metric form. Instead, it assumes that whatever form the rate function has, explanatory variables affect the rate by proportional changes.

This assumption can be illustrated as follows. Let $\rho_A(t)$ denote the (session or purchase) rate function for game version A. Then we assume that for the other version B, the rate function takes form $\rho_B(t) = \psi_B \rho_A(t)$, that is, the rate for version B is proportional to that of A. The coefficient ψ_B describes the relationship between the two rates. If $\psi_B = 1$, there is no difference. If $\psi_B > 1$, B has a higher rate, and if $\psi_B < 1$, A has the higher rate.

More generally, let $C = C_1 \times C_2 \times \dots \times C_k$ denote different categorical variables describing a group of players. For example, we could have $C_1 = \{A, B, C\}$, $C_2 = \{\text{android, ios}\}$, and $C_3 = \{\text{US, GB, AU...}\}$. Further, let $\mathbf{b} \in C$ denote a *baseline group*, whose (unknown) rate is defined as $\rho_0(t)$. The Cox model makes the following assumption about the relationship that holds between the baseline rate $\rho_0(t)$, and the rate $\rho(t|\mathbf{x})$ of any other group $\mathbf{x} \in C$:

$$\rho(t|\mathbf{x}) = \rho_0(t) \prod_{i=1}^k \psi_{x_i}, \quad (1)$$

where $\psi_{b_i} = 1$ for all values of i .

For example, assume $\mathbf{b} = (A, \text{android}, \text{US})$. Then, $\rho(t|(B, \text{android}, \text{GB})) = \psi_B \psi_{GB} \rho_0(t)$. Here, the rate of the baseline group is scaled by ψ_B since the players have game version B instead of A, and by ψ_{GB} since they are from Great Britain instead of US. The android platform does not affect the rate, since $\psi_{\text{android}} = 1$ because the baseline group consists of android users.

In order to fit the Cox model to a data set of players, the information required about each player consists of the times at which they played a session (or made or purchase), the length of time the player was observed for, and the values of the categorical variables for the player. After fitting the Cox model, we recover the ψ coefficients, as well as confidence intervals around them.

When estimating confidence intervals for rates it is common to assume that the events follow a Poisson process, meaning that they happen independent of process history. This is clearly an unrealistic assumption for player data. A player who has not returned for 90 days has a much smaller probability of playing today than a player who has played yesterday. Since inactivity predicts future inactivity, the probability of playing is not in fact independent of player history. We stress the importance of this difference; incorrect use of the independence assumption will lead to narrow confidence intervals with high optimistic bias. Instead, we use the general (robust) confidence intervals that do not rely on this assumption (Cook and Lawless, 2007).

The proportional hazards model is the most efficient test when the underlying assumption applies, but it is also robust to departures as long as the rates are uniformly greater or smaller. If one game version is better initially and worse on a later date, so that the rates cross each other, the difference is not reported in the coefficient estimates which in a sense measure the average difference (Cook and Lawless, 2007). From a gaming perspective, this interpretation is acceptable.

Model applications

Hipster Sheep Data Set

Our data set consists of users acquired to the Hipster Sheep, a puzzle-based free-to-play mobile game, where the player guides a sheep through a maze, while gathering collectibles and avoiding dangers. Current version of the game can be downloaded in the App Store and the Google Play store.

Several different player cohorts were acquired over the development time of the game through paid marketing. This was done both for testing development progress (i.e. how high retention the current version has), as well as to perform A/B test comparisons in order to decide among alternative designs. There was also some organic user acquisition, such as users inviting friends or discovering the game in Google Play. This unplanned acquisition allows us to later compare acquisition countries and versions more broadly. We filtered the data to players in United States (US), Great Britain (GB), Australia (AU) and Netherlands (NL) where acquisition was performed. This was done to exclude developer devices, and ensure that the users are comparable by acquisition method. User acquisition is visualized in Figure 3, where early-2016 versions were pooled together to obtain a significant user cohort.

We apply the Cox Model to three different ABC-test scenarios in the following three sections. This allows us to evaluate real-world game development suitability of the model:

1. Cohort comparison: Players were randomly assigned to three game versions with different progression speeds to evaluate the optimal choice in terms of retention.
2. Development progress: User acquisition was performed over game versions to test retention and monetization. We evaluate the degree of improvement.
3. Optimal user cohort: Since user acquisitions consisted of different countries, platforms and game versions, we evaluate who are the best users to market the game to.

We finally discuss the importance of stratification for correct interpretation. The Cox model analyses are performed using the R survival package.

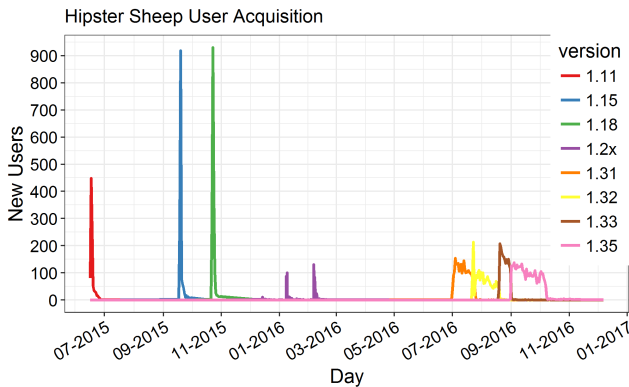


Figure 3: User acquisition was performed during Hipster Sheep development to test and improve game quality. Multiple versions were tested over 1.5 years.

Comparison by A/B/C-group

To evaluate optimal progression speed, in Hipster Sheep development version 1.18 the users were randomly assigned into different game versions when they installed the game. Progression speed was altered by adding or removing levels and disabling or unlocking earlier certain game features. This setting is an example of an ABC-test with 3 separate player cohorts: a_normal (596 players, 10035 sessions), b_faster (605 players, 11872 sessions) and c_fastest (610 players, 10050 sessions). Since monetization mechanics were not yet fully implemented in version 1.18 we skip purchase analysis.

In Figure 2, we see that the Nelson-Aalen estimate of the expected cumulative number of sessions $\mu(t)$ suggests that b_faster has the highest retention, whereas a_normal and c_fastest are roughly equal. To test this assumption, we consider A as baseline group, and find the coefficients ψ_B and ψ_C , with 95% confidence intervals, by fitting the Cox model. The estimated coefficients in Figure 4 support the visual observation; b_faster is 18% better than a_normal in terms of session rate ψ_B , whereas c_fastest is marginally (1%) worse. However, the confidence intervals around both b_faster and c_fastest both contain value 1.0, meaning that we cannot reject the null hypothesis that the rates do not differ from a_faster. To conclude, there is not enough evidence to conclude at $p = 0.05$ confidence level that b_faster really corresponds to a substantial improvement over the other versions.

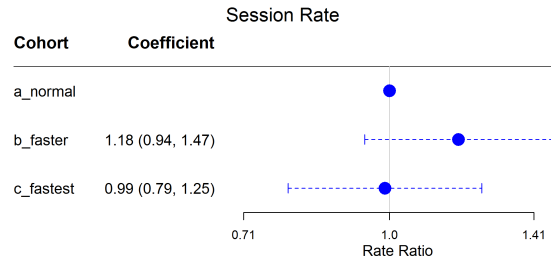


Figure 4: Session rate ratio of b.faster and c.fastest cohorts relative to the baseline a.normal with robust 95% confidence intervals.

Comparison by Version

	1.11	1.15	1.18	1.2x
Android				
Players.	983	1463	1811	570
Sessions	12353	25340	31847	11283
Purchases	1	3	14	4
iOS				
Players.	2518	1760	1894	3751
Sessions	19984	12871	14603	29600
Purchases	62	56	79	174

Table 1: Cohort Statistics for Android and iOS Versions

Development progress may be evaluated through improvements in retention and monetization over consecutive

Hipster Sheep Android&iOS Development Tests

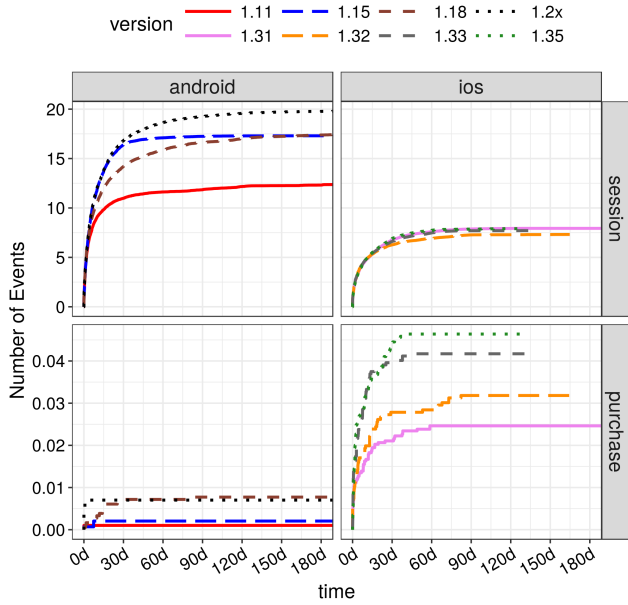


Figure 5: Cumulative number of sessions and purchases for four Android versions and four iOS versions.

game versions (Seufert, 2014). Table 1 lists four user acquisitions both for iOS (AU/NL) and Android (US/GB). Cox model is fitted separately to the iOS and Android cohorts.

The cumulative rate plots (Figure 5) suggest that for Android the changes 1.11 → 1.15 and 1.18 → 1.2x uniformly increased retention rates, whereas 1.15 → 1.18 was a paradigm shift with decreased early retention (0d 30d lower slope) balanced by higher retention (30d 180d steeper slope) over several subsequent months. For the iOS versions retention remained stable, whereas monetization appears to improve after each release. The plots are in line with the Cox rate ratio estimates (Figure 6). Relative to the 1.18 retention rate, the first version 1.11 is 29% lower, 1.15 is only 2% lower, and 1.2x finally improved 13%. Version 1.18 is statistically significantly better than 1.11, whereas other differences are not significant. Compared to iOS 1.31 version, for later versions monetization was improved by 29%, 69% and 88%, but the improvements are not statistically significant.

Comparison by Multiple Features

The targeted platform was not the only feature that varied in versions, but the acquisition source also varied. The early acquisitions for android were located in United States (US) and Great Britain (GB), whereas the new iOS acquisitions were located in Australia (AU) and Netherlands (NL). There was small-scale acquisition and organic discovery in the new 1.31, 1.32, 1.33 and 1.35 versions for android, located primarily in US and GB. These statistics are listed in Table 2, and they allow us to perform AB-tests with multiple features simultaneously across the categories.

It is possible to obtain an estimate for each segment simply by forming 32 mutually exclusive cohorts from 2 plat-

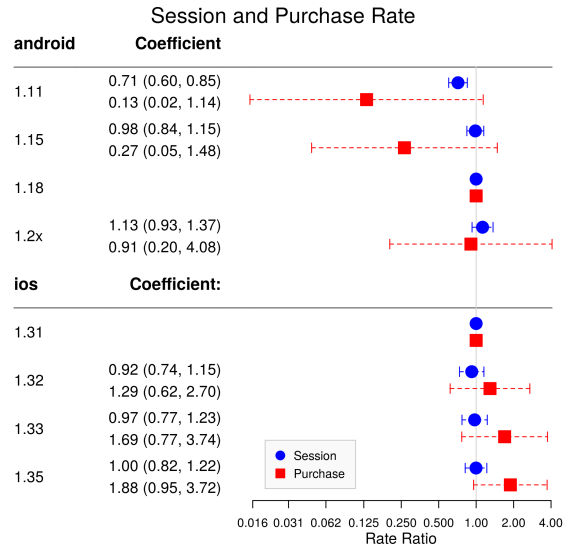


Figure 6: Session and purchase rate ratios with robust confidence intervals in Android development versions w.r.t 1.18 and current iOS versions w.r.t 1.31. Retention improves during development and monetization was adjusted later.

Platform	Country	1.31	1.32	1.33	1.35
android	US	23	152	8	17
	GB	40	232	5	13
	AU	3	15	1	5
	NL	0	0	0	1
iOS	US	20	19	17	36
	GB	6	3	6	6
	AU	1034	764	797	1843
	NL	1484	996	1097	1908

Table 2: Cohorts: 2 Players, 4 Countries, 4 Versions

forms, 4 countries and 4 versions. However, this approach would quickly result in not having enough data for the cohorts, and would not be informative on the effect of a particular feature. Rather, Cox regression allows us to perform the segment based analysis and estimation of feature effects.

The country may be expected to influence retention slightly but, given cultural factors and income levels, especially monetization should be affected. Since most android devices cost less than iOS devices, platform may also contribute to monetization differences. We already observed that the version changes both retention and monetization. Since the composition of user countries also changes across versions (see Table 2), this difference could potentially be caused by the change in the ratio of countries in the sample.

The Cox model estimates in Figure 7 answer these questions. The iOS retention rate differs significantly, being only half (0.47) of Android retention rate. The purchase rate is estimated four-fold (4.01) in comparable cohorts, though the confidence interval for this estimate has slight overlap with value 1.0, meaning that null hypothesis cannot be re-

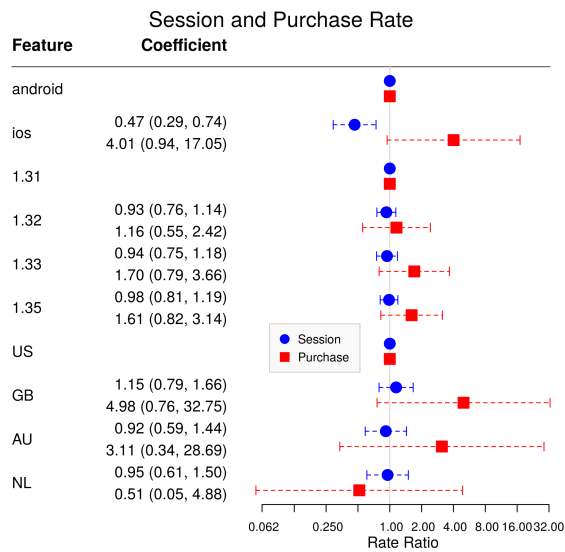


Figure 7: The coefficient forest plot displays how each categorical feature changes the session and purchase rate relative to android, 1.31 or US baseline.

jected. Estimates for versions relative retention rates (0.92, 0.97, 1.00, vs. 0.93, 0.94, 0.98) and purchase rates (1.29, 1.69, 1.88 vs. 1.16, 1.70, 1.61) change slightly from previous ones, due to correction for changes in country and platform, since these are included as features.

Stratification

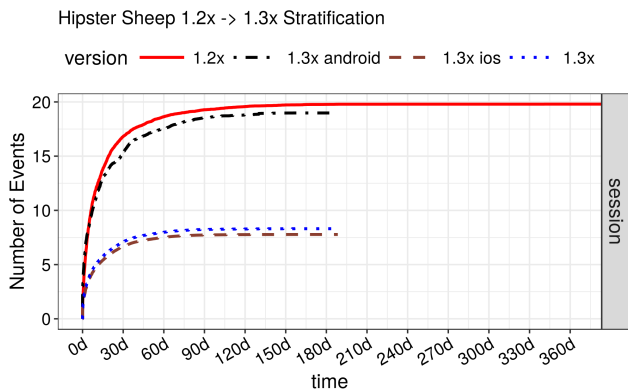


Figure 8: Comparing 1.2x and 1.3x directly implies that retention declined precipitously; stratifying on platform shows that this is caused by the fact that 1.3x included many iOS players who have worse retention in general.

Suppose we are analyzing how the upgrade from 1.2x to 1.3x affected retention. Figure 8 shows that direct comparison of these cohorts implies 1.2x retention (red) dropped to under a half in 1.3x (purple). However, one needs to take into account that 1.2x consisted of Android users, whereas 1.3x included mostly iOS users. Looking at the 1.3x upgrade

just for Android (blue), the retention stayed the same! Because the iOS users in 1.3x (green) play less, having them reduced retention.

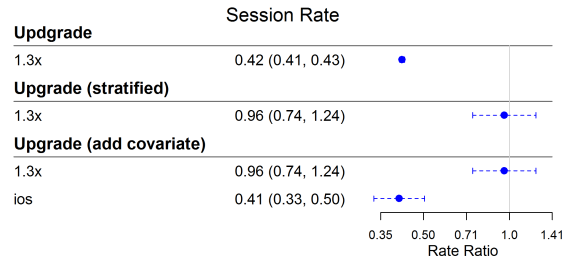


Figure 9: The importance of stratification is demonstrated by the misleading coefficient estimate of the non-stratified model compared to stratified and full model with platform covariate included.

Stratification allows reliable analysis in such cases. Stratification by platform can be implemented by analyzing player data from each platform separately. However, a better alternative to naive stratification is to model the relationship between the platforms explicitly by adding them to the Cox model as a feature. This achieves the same goal as stratification, and we also get an explicit estimate of the convoluting effect. In Figure 9 we have fit three Cox models to the data displayed in Figure 8. The direct comparison we discussed previously produces a highly significant 58% smaller retention (0.42), whereas the stratified and the feature added version have a barely noticeable 4% change, as was expected. The full model also estimates the effect of the iOS platform, which we can see at 59% reduction (0.41) was the actual cause of reduced retention in the direct comparison.

Conclusion

We introduced the use of recurrent event Cox model as a scientific multivariate test for comparing session and purchase rates between player cohorts. The method allows a game analyst to quantify effect size and assess statistical significance with respect to retention and monetization. We presented a number of practical use cases based on data from free-to-play game. These included choosing the best way to implement a feature into a game, following over time how much better (or worse) each release is, and evaluating and stratifying on effects variables such player platform or country have on retention and monetization. The method constitutes the first tool proposed in the literature, that allows statistical comparison of retention and purchase rates of games, while allowing churn uncertainty and differing follow-up times of players. We believe that the proposed approach provides both a valuable theoretical model for academic game analytics, as well as a very practical tool for game developers wanting to implement comparison tests.

Acknowledgments

We thank TribeFlame Ltd. for providing the game data.

References

- Allart, T.; Levieux, G.; Natkin, S.; and Guilloux, A. 2016. Design influence on player retention : A method based on time varying survival analysis. In *Proc. IEEE Computational Intelligence and Games Conf.* IEEE.
- Chen, K.-T.; Huang, P.; and Lei, C.-L. 2009. Effect of network quality on player departure behavior in online games. *IEEE Trans. Parallel Distrib. Syst.* 20(5):593–606.
- Cook, R. J., and Lawless, J. F. 2007. *The statistical analysis of recurrent events*. Statistics for biology and health. New York, London: Springer.
- Debeauvais, T., and Lopes, C. V. 2015. Gate me if you can: The impact of gating mechanics on retention and revenues in Jelly Splash. In Zagal, J. P.; MacCallum-Stewart, E.; and Togelius, J., eds., *Proc Int. Foundations of Digital Games Conf.* Society for the Advancement of the Science of Digital Games.
- Drachen, A.; Lundquist, E.; Kung, Y.; Rao, P.; Sifa, R.; Julian, R.; and Klabjan, D. 2016. Rapid prediction of player retention in free-to-play mobile games. In *Proc. AIIDE Artificial Intelligence and Interactive Digital Entertainment Conf.*
- El-Nasr, M. S.; Drachen, A.; and Canossa, A., eds. 2013. *Game Analytics, Maximizing the Value of Player Data*. Springer.
- Feng, W.-c.; Brandt, D.; and Saha, D. 2007. A long-term study of a popular MMORPG. In *Proc. ACM SIGCOMM Network and System Support for Games Workshop*, 19–24. New York, NY, USA: ACM.
- Hadji, F.; Sifa, R.; Drachen, A.; Thurau, C.; Kersting, K.; and Bauckhage, C. 2014. Predicting player churn in the wild. In *Proc. IEEE Computational Intelligence and Games Conf.*, 1–8.
- Isaksen, A.; Gopstein, D.; and Nealen, A. 2015. Exploring game space using survival analysis. In Zagal, J. P.; MacCallum-Stewart, E.; and Togelius, J., eds., *Proc. Int. Foundations of Digital Games Conf.* Society for the Advancement of the Science of Digital Games.
- Moore, D. F. 2016. *Applied survival analysis using R*. Use R! Cham: Springer.
- Periáñez, A.; Saas, A.; Guitart, A.; and Magne, C. 2016. Churn prediction in mobile social games: Towards a complete assessment using survival ensembles. In *Proc. IEEE International Conf. on Data Science and Advanced Analytics (DSAA)*. IEEE.
- Runge, J.; Gao, P.; Garcin, F.; and Faltings, B. 2014. Churn prediction for high-value players in casual social games. In *Proc. IEEE Computational Intelligence and Games Conf.*, 1–8.
- Seufert, E. B. 2014. Freemium metrics. In *Freemium Economics*. Boston: Morgan Kaufmann. 83–113.
- Sifa, R.; Bauckhage, C.; and Drachen, A. 2014. The playtime principle: Large-scale cross-games interest modeling. In *Proc. IEEE Computational Intelligence and Games Conf.*, 1–8. IEEE.
- Sifa, R.; Hadji, F.; Drachen, A.; and Runge, J. 2015. Predicting purchase decisions in mobile free to play games. In *Proc. AIIDE Artificial Intelligence and Interactive Digital Entertainment Conf.*
- Sifa, R.; Srikanth, S.; Drachen, A.; Ojeda, C.; and Bauckhage, C. 2016. Predicting retention in sandbox games with tensor factorization-based representation learning. In *Proc. IEEE Computational Intelligence and Games Conf.*, 142–149. IEEE.
- Tamassia, M.; Raffe, W.; Sifa, R.; Drachen, A.; Zambetta, F.; and Hitchens, M. 2016. Predicting player churn in Destiny: A hidden Markov models approach. In *Proc. IEEE Computational Intelligence and Games Conf.*, 325–332. IEEE.
- Tarng, P.-Y.; Chen, K.-T.; and Huang, P. 2008. An analysis of WoW players’ game hours. In *Proc. ACM SIGCOMM Network and System Support for Games Workshop*, 47–52. New York, NY, USA: ACM.
- Viljanen, M.; Airola, A.; Pahikkala, T.; and Heikkonen, J. 2016. Modelling user retention in mobile games. In *Proc. IEEE Computational Intelligence and Games Conf.*, 62–69. IEEE.
- Viljanen, M.; Airola, A.; Heikkonen, J.; and Pahikkala, T. 2017. Playtime measurement with survival analysis. *IEEE Transactions on Computational Intelligence and AI in Games*. In press.
- Weber, B. G.; Mateas, M.; and Jhala, A. 2011. Using data mining to model player experience. In *Proc. FDG Evaluating Player Experience in Games Workshop*. ACM.