Predicting the monetization percentage with survival analysis in free-to-play games

Riikka Numminen
Department of Future Technologies
University of Turku
Turku, Finland
rimanu@utu.fi

Markus Viljanen
Department of Future Technologies
University of Turku
Turku, Finland
majuvi@utu.fi

Tapio Pahikkala
Department of Future Technologies
University of Turku
Turku, Finland
aatapa@utu.fi

***Abstract*—Understanding and predicting player monetization is very important, because the free-to-play revenue model is so common. Many game developers now face a new challenge of getting users to buy in the game rather than getting users to buy the game. In this paper, we present a method to predict what percentage of all players will eventually monetize for a limited follow-up game data set. We assume that the data is described by a survival analysis based cure model, which can be applied to unlabeled data collected from any free-to-play game. The model has latent variables, so we solve the optimal parameters of the model with the Expectation Maximization algorithm. The result is a simple iterative algorithm, which returns the estimated monetization percentage and the estimated monetization rate in the data set.**

***Index Terms*—Free-to-play, Monetization, Survival Analysis**

I. Introduction

Total revenue from video games reached approximately 110 billion dollars in 2018 [1]. As the gaming industry has grown, the revenue models have also evolved. Most revenue in the past was from game purchases and subscriptions whereas today free-to-play games account for the majority of all game titles and revenues [1]. Money is made through advertisements, premium upgrades and in-app purchases. However, only around 5 % of players in a successful free-to-play game can be expected to monetize [2]. These developments have made it important to understand exactly how many percentage of players monetize and why they do so.

The goal of this paper is to present a method to predict the proportion of all players that will monetize over time. Our method belongs to the field of game analytics, which is concerned with understanding player behavior. Game developers often want to perform real-time analytics when they are developing or planning to launch a game. A data set is generated by tracking players for a certain duration, which we call the follow-up. However, data that is collected over a limited duration presents challenges. The data sets can be divided into a scale between two extremes:

1) Extensive historical game data spanning maybe years.
2) A completely new game data set with short follow-ups.

The first setting allows the usage of supervised machine learning models, since it is known which users made the purchases and can thus be generalized to the same game. However, in the second setting a new game with only few observed purchases may be dealt with and the data set is unsupervised because the correct answers are not known for most of the players. Players who have not yet made a purchase are difficult to separate from those who will never purchase. The first setting is well suited for academic research. The second setting occurs when game developers want to use real-time analytics to understand their current game. They want to know as soon as possible how profitable a game can be expected to be, as they do not want to expend finances on advertising if the game is expected to be unprofitable.

Game literature has demonstrated that it is possible to train machine learning models with good predictive performance in the first setting, many of which were featured in a recent competition [3]. Research has been more limited in the second setting. Our method works in the second setting, for a new game with only a short period of data collection. However, at least one player must have made a purchase before the method can be used. Many real world data sets are somewhere between these two extremes and the approaches could complement each other. We focus on the monetization percentage and rate in this paper, but in principle our latent variable formulation could be used together with many supervised machine learning models to train them on unsupervised data due to a limited follow-up.

This article is organized as follows: first there is a short literature review, then the method is described along with survival analysis and mixture cure model theory, after that the data used for testing the method is described, and finally the results are presented and discussed.

II. Related work

Regression and machine learning have been used in studies to predict player purchases in various problem formulations for labeled historical data. Random Forest, linear SVM, and Decision Tree were used first in [4] to classify whether a player would buy an in-game item after a match. Both general in-game items purchases and hard currency purchases were predicted. Similarly, given purchasing and non-purchasing players with two weeks of history before the purchase, Decision Tree, Logistic Regression, and SVM were used in [5] to predict which group the player belonged to, with a focus on the game

978-1-7281-1884-0/19/$31.00

agnostic features. In [6], both classification and regression were used to predict whether the user would make a first purchase and how many purchases would occur. Decision Trees, Random Forests, and Support Vector Machines were used with data set balancing methods. Finally, in [7] two linear regressions were used to model the number of purchases and the number of coins purchased at a given level. The focus was on understanding the impact of gating mechanisms on retention and monetization.

Survival analysis has been used in gaming for various other tasks, see [8] for a review. Noncontractual probability models used in marketing [9] are closest to our approach. These models predict player purchase counts over time, given a data set in the second setting. However, one of the most popular models (BG/NBD) was tested in free-to-play games and the authors found that the model struggled with covering real data [10]. It has been suggested that further research should be conducted to redesign or adjust existing models, in order to examine better assumptions for free-to-play games. In a recent approach [11], a model-free method was developed to measure the mean customer lifetime value (LTV) in the second setting, but this approach did not predict into the future. Studies have also investigated how the first purchase predicts overall LTV [12], which is very useful when used together with our model.

III. Method

In this section we present the method. The first subsection shows how game data is assumed to follow a simple mixture cure model, which is an extension of standard survival analysis. The second subsection shows how the monetization percentage and conversion rate can be predicted by finding optimal model parameters via the Expectation Maximization algorithm (EM-algorithm).

A. Monetization as a mixture cure model

Survival analysis is a field of statistics used to analyze time-to-event data with limited follow-ups. The type of event depends on the field where the data are collected. In medical research the event is often the death of a patient and in industry the event might be that a certain part of a machine breakdowns. Limited follow-up causes data censoring, which means that it is not possible to know all the event times because the follow-up ends before some of the events occurred. In standard survival analysis, each individual is assumed to eventually have the event, and this event happens once, at most, even though observations may be censored [13].

In this paper, the event is that a player makes his first purchase, and the event time is the calendar time from a player starting to play the game to purchasing something for the first time. Game developers cannot wait indefinitely to collect the data, which leads to a censored data set. Players that did not make a purchase during the time interval when the data were collected are censored.

Two random variables are observed when the data are collected [14]. One of them is the event time T which is either the purchasing time T^* or the censoring time C whichever is smaller: $T = \min(T^*, C)$. The other variable is the censoring indicator $\delta = \mathbb{I}(C \leq T^*)$ which represents whether a player purchased something before censoring or was censored by the follow-up. Realizations of these random variables are denoted with $t_i = \min(t_i^*, c_i)$ and $\delta_i = \mathbb{I}(c_i \leq t_i^*)$ and they are observed for every player.

In this paper, we assume that the purchase time $T^* \sim \text{Exp}(\lambda)$ and the censoring time C is implied by the data collection time. The assumption of purchase time following the exponential distribution is a special case of the playtime principle introduced in [15] and has been used to model player survival [16]. The distribution of T is defined by a survival function S [17], which describes the probability of purchasing after time t. We use the following exponential model

$$S(t) = P(T > t) = e^{-\lambda t}. \tag{1}$$

Hazard function h describes the instantaneous risk that a player purchases at time t given that he did not do so until that time. In the exponential model the risk is constant [14]:

$$h(t) = \lim_{\Delta t \to 0} \frac{P(t < T < t + \Delta t | T > t)}{\Delta t} = \lambda. \tag{2}$$

The third function needed to describe the situation is a probability density function

$$f(t) = h(t) S(t), \tag{3}$$

which is the density of purchases at time t.

The assumption of standard survival analysis, that all individuals eventually purchase, probably does not apply in free-to-play games since many players seem to never buy anything. This would mean that there are players of two kinds, monetizing and unmonetizing, and the whole population is a mixture of the two sub-populations. Hence a mixture cure model [18] is required in order to properly model this situation. All unmonetizing players are always censored, but also some of the monetizing players might be censored if they were not followed for long enough. Thus the division into censored and purchased players does not provide sufficient information about the number of monetizing players and a third variable, a monetizing indicator, is needed.

The monetizing indicator ζ describes whether a player will monetize or not: $\zeta = 0$ for the monetizing population and $\zeta = 1$ for the unmonetizing. The probabilities that a player is a monetizing or an unmonetizing player are $P(\zeta = 0) = \pi$ and $P(\zeta = 1) = 1 - \pi$, respectively. Monetizing indicator $\zeta \sim \text{Bern}(1 - \pi)$ and is partly latent because the value of it is known only for those players that made a purchase before the censoring. In a mixture cure model, the survival function is a weighted sum of the survival functions of the subpopulations: $S(t) = \pi S_m(t) + (1 - \pi) S_u(t)$, where the weight π is the percentage of monetizing individuals in the whole sample, $S_m(t) = e^{-\lambda t}$ and $S_u(t) \equiv 1$. Given that the purchase time follows the exponential distribution there are now two parameters that describe the model: $\mathbf{\Psi} = (\pi, \lambda)$. They are monetization percentage and conversion rate.

An example of this kind of data is shown in Fig. 1. In the example there are 50 players, of whom nine are monetizing. It can be seen that two monetizing players have not purchased before censoring. The players have started within three units of the calendar time and censoring occurs five time units after the first player arrived. This results in different follow-ups for the players. In reality the monetization status of players, i.e. the colors in the figure, are not known, and it is necessary to infer from the data set how many players are going to make a purchase.

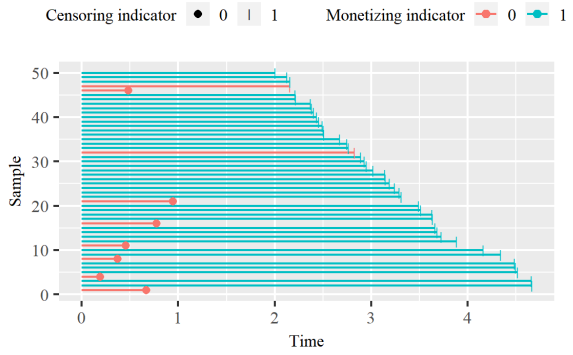

Fig. 1. Simulated data example with 50 players: 9 monetizing and 41 unmonetizing. The data was generated with parameters $(\pi, \lambda) = (0.1, 1.0)$.

B. Fitting the mixture cure model

We can infer the monetization percentage and the conversion rate by finding the model parameter vector $\mathbf{\Psi}$. The likelihood function L shows how likely the probability distribution samples are, given values for the parameters. The maximum likelihood estimate $\hat{\mathbf{\Psi}}$ is the parameter vector that maximizes the likelihood function, i.e. parameter values that make the given data most likely. In survival analysis the likelihood function

$$L(\mathbf{\Psi}) = \prod_{i=1}^{n} f(t_i|\mathbf{\Psi})^{1-\delta_i} S(t_i|\mathbf{\Psi})^{\delta_i}, \tag{4}$$

where n is the sample size. However, the logarithm of it,

$$l(\mathbf{\Psi}) = \sum_{i=1}^{n} \{(1-\delta_i) \log f(t_i|\mathbf{\Psi}) + \delta_i \log S(t_i|\mathbf{\Psi})\}, \tag{5}$$

is often used instead [19]. Since the logarithm is a strictly increasing function, the maximum likelihood estimate $\hat{\mathbf{\Psi}}$ is the same for both (4) and (5). In the maximum likelihood estimation the parameter values converge in probability to the true parameter values as $n \to \infty$ [20].

If the latent monetizing status ζ_i is somehow known for every player, the data is said to be complete and the solution is both simple and intuitive. The total number of players is denoted with $n = n_0 + n_1$, where n_0 stands for the number of monetizing players and n_1 is the number of unmonetizing players. In order to find the maximum likelihood estimate, the roots of the partial derivatives of (4) or (5) with respect to π and λ are found separately. Then the parameters simply are

$$\hat{\pi} = \frac{n_0}{n} \text{ and } \hat{\lambda} = \frac{n_0}{\sum_{i:\zeta_i=0} t_i}. \tag{6}$$

In other words, the monetization percentage is the fraction of players that purchase something. The conversion rate is the number of monetized players divided by their total exposure time. When considering the fact that $\mathbb{E}[T] = 1/\lambda$, it can be seen that the expected purchase time is the average of exposure times in the monetizing population.

However, there is a latent variable in the mixture cure model since it is not known which players are monetizing. The latent variable makes it impossible to find an equation for the maximum likelihood estimate. EM-algorithm [21] is an iterative algorithm that is suitable for maximum likelihood estimation in situations where there are missing data or latent variables. In a situation like this, the observed data is said to be incomplete. As shown in the appendix, this results in an iterative algorithm, which updates the current value of the parameter $\pi^{(k)}$ by

$$\pi^{(k)} = \frac{1}{n}\left[\sum_{i:\delta_i=1} \frac{\pi^{(k-1)} e^{-\lambda^{(k-1)} t_i}}{1-\pi^{(k-1)}+\pi^{(k-1)} e^{-\lambda^{(k-1)} t_i}} + \sum_{i:\delta_i=0} 1\right] \tag{7}$$

and the current value of $\lambda^{(k)}$ is calculated with

$$\lambda^{(k)} = \frac{\sum_{i:\delta_i=0} 1}{\sum_{i:\delta_i=1} \frac{\pi^{(k-1)} e^{-\lambda^{(k-1)} t_i}}{1-\pi^{(k-1)}+\pi^{(k-1)} e^{-\lambda^{(k-1)} t_i}} t_i + \sum_{i:\delta_i=0} t_i}. \tag{8}$$

These estimates converge to the global maximum of the log-likelihood function when $k \to \infty$. The values of the log-likelihood function (5), and the method iterations (7) and (8) are illustrated in Fig. 2 for the data represented in Fig. 1.

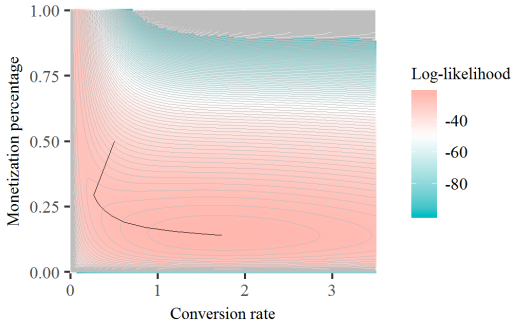

Fig. 2. Values of the log-likelihood as a function of π and λ. The path of the EM-algorithm iterations is presented as a curve from the initial guess $\mathbf{\Psi}^{(0)} = (0.5, 0.5)$ to the maximum likelihood estimate $\left(\hat{\pi}, \hat{\lambda}\right) = (0.1, 1.7)$.

There were also some problematic cases when the model did not work correctly. A zero-frequency problem is present when almost all players are censored and there is no information showing that some of the censored players never monetize. In such a situation the model is not able to distinguish between the survival analysis model and the mixture cure model, and the method predicts what the survival analysis assumes, i.e. that every player monetizes eventually. This problem can be

avoided by using Laplace smoothing [22] which is a method that makes all classes (unmonetizing and monetizing populations in our case) possible by adding pseudo-observations to the data. It is enough to add one unmonetizing pseudo-observation with infinite follow-up to the data to avoid the zero-frequency problem and obtain reasonable results in reasonable computation time. The algorithm should not be run at all when none of the players have been observed to make a purchase because then $\lambda^{(k)} = 0$ for all $k \geq 1$ and the convergence rate of the algorithm is not defined. In this kind of situation the monetization percentage is defined as zero.

IV. Data sets

Based on the formulas derived, the model was implemented with the R programming language. We tested the model on both generated and real data.

A. Generated data

There is no publicly available person level data set with many free-to-play games. One of our primary motivations is to apply the algorithm to completely new games with limited follow-ups, therefore we simulated different sample sizes n, follow-up times c, and monetization percentages π and then calculated the estimate $\hat{\pi}$ for each data set.

In the first experiment, we calculated the monetization percentage estimates for different sample sizes and follow-up times, given a true value of $\pi = 0.10$. The effect of sample size n was tested with values 100, 500, 1000 and 5000. The follow-up time c was tested with censoring times that are defined in a way that there is 25 %, 50 %, 75 % or 100 % probability for the monetizing players to purchase before censoring. For each combination, we conduct a thousand experiments $r = 1, ..., 1000$. In each experiment, we sampled a player data set of size n, where the observed time $t_i = \min(t_i^*, c)$, and calculated the predicted value of $\hat{\pi}_r$ using the EM-algorithm. The resulting distribution of $\hat{\pi}_r$ is compared to the true value.

In the second experiment, we tested the effect of the true monetization percentage on the estimate. That effect was tested with monetization percentages of 0.01, 0.05, 0.1, 0.5 and 0.75. Follow-up times are defined the same way as in the first simulation but with 10–100 % probabilities to purchase before censoring. The effect of the sample size was tested with 10 different sample sizes varying from 100 to 1000 by 100. For each (n, π, c) triplet, 1000 estimates were computed and averaged.

B. Real data

The real data were collected from a free-to-play mobile game. The game was in-development during data collection, and many developments were made to each version over the game development cycle. Periodical user acquisition tests were used to evaluate the current performance. In these tests, a group of players was obtained by using paid advertisement in social networks and the behavior of the players was recorded.

The number of players varies between different versions and only a small percentage of the players purchased an item as can be seen in Table I. The game probably improved monetization from version 1.18 to version 1.21, but the subsequent development in the 1.3x series had no large effect on the percentage of purchasing players. The time to monetization was the calendar time from the beginning of the first session to the first purchase. The censoring times were defined as the time from the first session to the data collection time.

TABLE I
Numbers of players and monetized players in the real data.

Version	# of players	# of monetized players	π
1.18	1604	6	0.004
1.21	309	6	0.019
1.31	1691	24	0.014
1.32	1582	21	0.013
1.33	1211	18	0.015
1.35	2364	35	0.015

We used this data to create censored data sets that replicate the actual version user test. We took the first player's the first session and defined censored data sets by varying the data collection date as $1, ..., D$ days of calendar time from this time. Maximal follow-up D denotes the actual data collection date, after which we have no data. This created censored players with different follow-up times, exactly the same way the data set would be obtained if it was updated at the end of each day after the test began. Since there are no purchases for months after the last purchase in each version, we assumed that the final data collection is effectively uncensored in that we can directly see who monetized and who did not.

V. Results and discussion

A. Simulation

The results of the first simulation are shown in Fig. 3. Where we know the monetizing players in advance we present the complete data parameters for comparison. It can be seen that for complete data the peak of the density function is always at true value $\pi = 0.1$, i.e. the method is unbiased. Nonetheless, even with the complete data, there is some variance, which decreases as the sample size becomes larger. For our method in Fig. 3, the incomplete data estimates may be initially slightly biased downward. The variance is somewhat larger, as there is less information. Both parameter estimates converge in probability as we increase the sample size or the follow-up, but this occurs surprisingly slowly for the sample size in the incomplete data case. Maximum likelihood theory guarantees that even the limited follow-up estimates become asymptotically unbiased as the sample size is increased. Fig. 3 however demonstrates, that even 5 000 samples are not sufficient if the real monetization percentage is small.

The results of the second simulation shown in Fig. 4 provide additional description of the bias. Relative bias is the difference between the estimate and the real value divided by the real value. The estimate is negatively biased when the follow-up is very short, and unbiased when the follow-up approaches infinity, i.e. the complete data case. The values between these

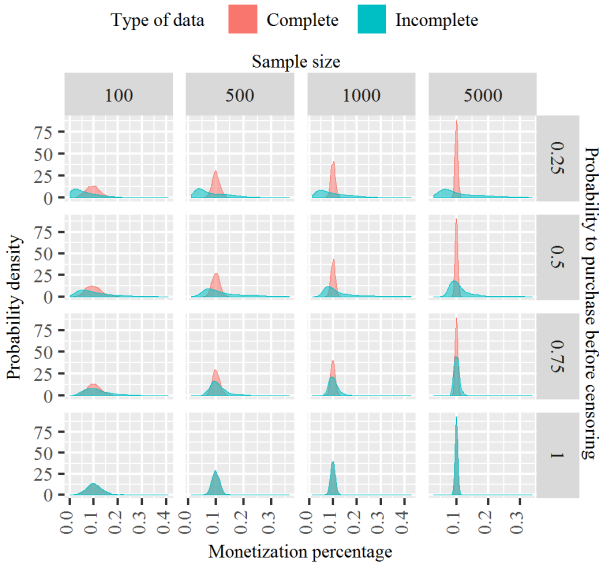

Fig. 3. Simulated estimates of monetization percentage π.

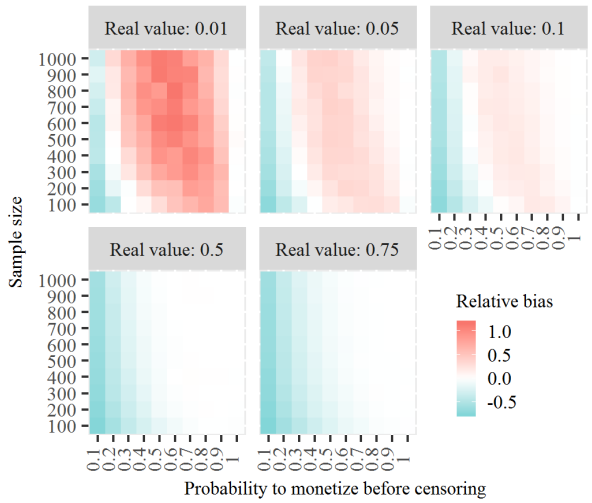

Fig. 4. Relative bias for monetization percentages π as a function of censoring time and sample size.

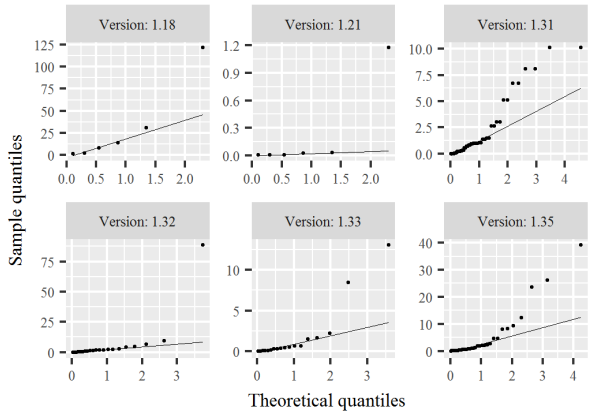

Fig. 5. Q–Q-plots of the event times of each version.

extremes vary depending on the real monetization percentage. For small monetization percentages, which are common in free-to-play games, there is a wide range of censoring times when the estimate is positively biased. The model seems to predict the true value when around 20 % have monetized, with a negative bias before and a positive bias after.

This bias is related to the identifiability issues we encountered with the model without a single pseudo-observation. The problem we had was to identify the cure model from the alternative explanation $\hat{\pi} = 1, \hat{\lambda} \approx 0$ offered by the standard model. With small samples and short follow-up times, both models are at times almost equally likely according to maximum likelihood. The positive bias can be explained by a model that is more tilted to the standard model direction, whereas the negative bias exists because the single pseudo-observation obtains considerable weight with only a few monetized players.

B. Real data

There are two assumptions in the model about the data which need to be verified for the real data. The assumptions are:

1) the event time is exponentially distributed and
2) there are some unmonetizing players, i.e. $\pi < 1$.

Assumption 1 is verified with Q–Q-plot which compares the quantiles of observed event times to the quantiles of an exponential distribution. The Q–Q-plots are shown in Fig. 5. There are some exceptions with very late purchase times, but most of the points are along a straight line. This suggests that we may assume event times to follow exponential distribution, for the purposes of estimating the monetization fraction.

Akaike's information criterion [23] is used to verify assumption 2. This method consists of calculating an AIC value for each possible model and the smaller the value, the better the model describes the data. The value is calculated with

$$\mathrm{AIC} = 2n_p - \log\left(\hat{L}\right), \tag{9}$$

where n_p is the number of parameters in a model and $\hat{L} = L\left(\hat{\mathbf{\Psi}}\right)$. A maximum likelihood estimate is calculated for the incomplete data likelihood function because the values of the monetizing indicator ζ are not known and the value of the complete data likelihood function cannot be calculated. AIC values in Table II show that the mixture cure model is better than the regular survival model at explaining the data for every game version. This is not surprising, given our prior knowledge about free-to-play monetization.

We show both the computed estimate $\hat{\pi}$ and the percentage of monetized players so far at each censoring time for every game version in Fig. 6. Obviously the percentage of monetized players increases as there are new monetized players. The value decreases if the number of monetized players does

TABLE II
AIC VALUES FOR MIXTURE CURE AND REGULAR SURVIVAL MODELS.

Version	Model	
	$\pi < 1$	$\pi = 1$
1.18	135.6720	154.0050
1.21	114.0415	132.1888
1.31	450.3302	501.7088
1.32	401.4259	436.8337
1.33	338.0034	364.8098
1.35	659.8481	693.9284

not increase at the same rate as the total number of players increases. At first, the estimates are greater than the monetized percentages as they extrapolate to the true value, but as the follow-up time is increased, both estimates become equal to the supposed true value.

The larger the purchase times, the smaller the $\hat{\lambda}$, and the more slowly the value of survival function (1) decreases. More players are then expected to monetize than have been observed so far, implying that the estimate $\hat{\pi}$ is larger than the real value. This effect cannot be seen with version 1.21 because all purchase times are small. As can be seen also in Fig. 6, the actual monetization percentage is revealed when no more purchases occur and the survival function decreases to practically zero at this point. This seems to happen within some months, depending on the version of the game.

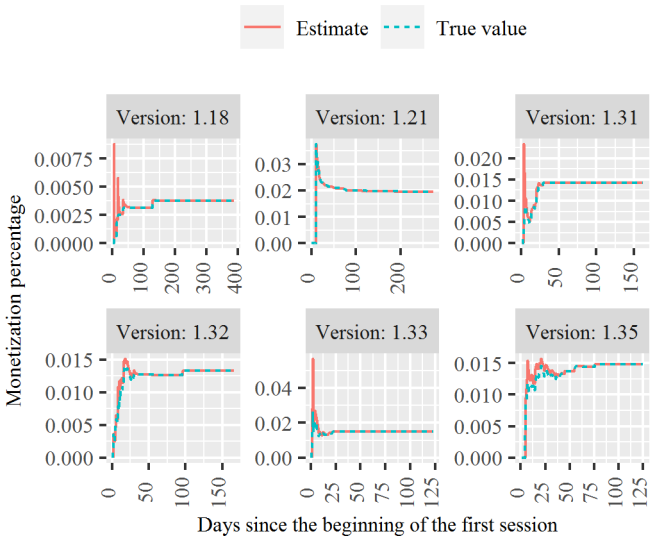

Fig. 6. Estimates of monetization percentage π of the real data compared to the true value at each censoring time.

C. Future work: prior distributions

The results suggest that the model, where some players monetize and some never do, is a correct interpretation of the data. However, they also demonstrate that it is difficult to estimate the monetization percentage reliably, even with complete data, since there is often little actual information. For example, with 100 players, 5 % monetization percentage, and a follow-up until 20 % are expected to monetize, we have on average $100 \cdot 0.05 \cdot 0.2 = 1$ monetized player. In fact, many such samples do not include even a single monetized player.

This causes two issues as regards the parameter estimates: bias and variance. The results show that the estimated monetization percentage varies significantly in different samples and that the cure model has some bias. These challenges could be addressed in practical applications by using prior distributions. Maximum likelihood is a sensitive method, because it simply chooses parameters that make each sample most likely. However, we have prior knowledge that most free-to-play games have around 1–5 % monetization rates. A prior distribution and Bayesian inference could be used to reflect this fact [24]. Given correct assumptions, this would result in lower bias and lower variance. For the monetization percentage π, a natural prior is the Beta(a,b) distribution. Three examples of beta distribution with different parameter values are visualized in Fig. 7. In fact, maximum likelihood corresponds to using an uniform prior Beta(1,1) and the one pseudo-observation we added corresponds to a Beta(1,2) distribution. This makes the value $\hat{\pi} = 1.0$ impossible, with a uniformly increasing likelihood of smaller values. A weakly enforced prior with 5 % mean monetization could be created by adding 1 monetizing player and 19 unmonetizing players, or Beta(2,20) prior.

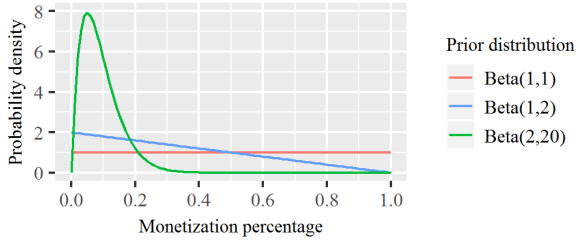

Fig. 7. Possible prior distributions for the monetization percentage π.

VI. CONCLUSION

The method introduced in this paper can be used to predict what percentage of all players will eventually monetize in limited follow-up game data sets. The prediction is made by applying a survival analysis based cure model to unlabeled data collected from any free-to-play game. The model is an iterative algorithm that returns the monetization percentage and the conversion rate in the data set. In the appendix, we motivate the formulas by the Expectation Maximization algorithm, which solves the optimal values for the parameters.

We found that the model can be used for prediction, but that for some data sets there are potential difficulties in identifying between the standard model and the cure model. There is only a little actual information in the data concerning limited sample sizes and follow-up times, because the monetization percentage is very small. This introduces some bias and variance to the estimates. The variance of the estimates can be quite large when the sample size and the follow-up time are limited. The bias and variance can both be reduced

with longer follow-up times and larger samples, or using prior distributions. The same results were observed with both generated data and real data.

We motivated the method by contrasting historical labeled game data and completely new unlabeled game data. As discussed, most game data sets are between these two extremes. In this case, the approaches are complementary. The latent variable formulation can be used to infer the probabilities for the labels, which can be trained using supervised learning techniques. For example, in medicine the model has been extended with covariates $\overline{X}$ as $\zeta \sim \text{LogReg}\left(\overline{\beta}^T\overline{X}\right)$ and $T \sim \text{CoxPH}\left(\overline{\beta}^T\overline{X}\right)$ [18]. In gaming, a similar approach could be used to predict monetization status using player features, even though it is not known for certain who has monetized. The latent variable formulation can be applied to other problems, such as churn prediction, which is a very popular machine learning prediction task.

APPENDIX

We derive the iteration formulas (7) and (8) using the EM-algorithm in this section. These formulas find the maximum likelihood estimate in the incomplete data case. EM-algorithm finds the maximum likelihood estimate $\hat{\mathbf{\Psi}}$ for incomplete data by taking advantage of conditional expectation value of complete data likelihood function L_c. The algorithm consists of two parts:

1) E-step: estimate the missing data by taking the expectation of the complete data likelihood function

$$Q\left(\mathbf{\Psi};\mathbf{\Psi}^{(k)}\right) = \mathbb{E}_{\mathbf{\Psi}^{(k)}}\left\{\log L_c\left(\mathbf{\Psi}\right)|\boldsymbol{y}\right\} \quad \text{and} \tag{10}$$

2) M-step: find a vector $\mathbf{\Psi}^{(k+1)} \in \mathbf{\Omega}$ for which

$$Q\left(\mathbf{\Psi}^{(k+1)};\mathbf{\Psi}^{(k)}\right) \geq Q\left(\mathbf{\Psi};\mathbf{\Psi}^{(k)}\right) \tag{11}$$

for all $\mathbf{\Psi} \in \mathbf{\Omega}$, where $\mathbf{\Omega}$ is the space of possible parameter values.

These two phases are repeated until convergence is achieved. It is shown in [21] and [25] that the algorithm converges to a local maximum of the likelihood function. In our case the log-likelihood function is concave, which implies that there is only one maximum and it is the global maximum. Thus these estimates converge to the global maximum of the log-likelihood function when $k \to \infty$. We first derive a formula for (10) and then we obtain the maximum of (11) by finding the roots of the partial derivatives with respect to π and λ separately. At first the conditional probabilities are shown in table III.

TABLE III
PROBABILITIES OF EVENT TIME T CONDITIONED ON MONETIZING INDICATOR ζ

$P(T\|\zeta)$	$T=t$	$T>t$
$\zeta=1$	0	1
$\zeta=0$	$\lambda e^{-\lambda t}$	$e^{-\lambda t}$

Then the formulas of marginals of survival and density functions in this mixture cure model case are

$$\begin{aligned} S(t) &= P(T>t) \\ &= P(T>t|\zeta=0)\,P(\zeta=0) + P(T>t|\zeta=1)\,P(\zeta=1) \\ &= e^{-\lambda t}\cdot\pi + 1\cdot(1-\pi) \\ &= 1-\pi+\pi e^{-\lambda t} \end{aligned} \tag{12}$$

and

$$\begin{aligned} f(t) &= P(T=t) \\ &= P(T=t|\zeta=0)\,P(\zeta=0) + P(T=t|\zeta=1)\,P(\zeta=1) \\ &= \lambda e^{-\lambda t}\cdot\pi + 0\cdot(1-\pi) \\ &= \pi\lambda e^{-\lambda t}. \end{aligned} \tag{13}$$

Now the formulas of incomplete data likelihood and log-likelihood functions are functions (4) and (5) having functions (12) and (13) substituted. The EM-algorithm requires the complete data likelihood function which is

$$\begin{aligned} L_c\left(\mathbf{\Psi}\mid \boldsymbol{t},\boldsymbol{\delta},\boldsymbol{\zeta}\right) = &\prod_{i=1}^{n}\left(\pi\lambda e^{-\lambda t_i}\right)^{1-\delta_i} \\ &\cdot\left[\left(1-\pi\right)^{\mathbb{I}(\zeta_i=1)}\left(\pi e^{-\lambda t_i}\right)^{\mathbb{I}(\zeta_i=0)}\right]^{\delta_i} \end{aligned} \tag{14}$$

and the logarithm of it is

$$\begin{aligned} l_c\left(\mathbf{\Psi}|\boldsymbol{t},\boldsymbol{\delta},\boldsymbol{\zeta}\right) = \sum_{i=1}^{n}\{&(1-\delta_i)\left[\log\pi+\log\lambda-\lambda t_i\right] \\ &+\delta_i\left[\mathbb{I}\left(\zeta_i=1\right)\log\left(1-\pi\right)\right. \\ &\left.+\ \mathbb{I}\left(\zeta_i=0\right)\left(\log\pi-\lambda t_i\right)\right]\}. \end{aligned} \tag{15}$$

The last thing needed to define (10) is the probability to be an unmonetizing player conditioned on purchase time. These probabilities are calculated with the Bayes' theorem:

$$\begin{aligned} &P\left(\zeta_i=j|T>t_i,\mathbf{\Psi}^{(k-1)}\right) \\ &= \frac{P\left(T>t_i|\zeta_i=j,\mathbf{\Psi}^{(k-1)}\right)P(\zeta_i=j)}{P\left(T>t_i|\zeta_i=0,\mathbf{\Psi}^{(k-1)}\right)P(\zeta_i=0)+P\left(T>t_i|\zeta_i=1,\mathbf{\Psi}^{(k-1)}\right)P(\zeta_i=1)} \end{aligned} \tag{16}$$

and

$$\begin{aligned} &P\left(\zeta_i=j|T=t_i,\mathbf{\Psi}^{(k-1)}\right) \\ &= \frac{P\left(T=t_i|\zeta_i=j,\mathbf{\Psi}^{(k-1)}\right)P(\zeta_i=j)}{P\left(T=t_i|\zeta_i=0,\mathbf{\Psi}^{(k-1)}\right)P(\zeta_i=0)+P\left(T=t_i|\zeta_i=1,\mathbf{\Psi}^{(k-1)}\right)P(\zeta_i=1)}, \end{aligned} \tag{17}$$

and the results are shown in table IV.

TABLE IV
PROBABILITIES OF MONETIZING INDICATOR ζ CONDITIONED ON EVENT TIME T

$P(\zeta\|T)$	$\zeta=1$	$\zeta=0$
$T>t$	$\frac{1-\pi}{1-\pi+\pi e^{-\lambda t}}$	$\frac{\pi e^{-\lambda t}}{1-\pi+\pi e^{-\lambda t}}$
$T=t$	0	1

The resulting E-step function in a reduced form is

$$
\begin{aligned}
&Q\left(\boldsymbol{\Psi}|\boldsymbol{\Psi}^{(k-1)}\right) = \mathbb{E}_{\zeta|T,\boldsymbol{\Psi}^{(k-1)}}\left[l_c\left(\boldsymbol{\Psi}|\boldsymbol{t},\boldsymbol{\delta},\boldsymbol{\zeta}\right)\right] \\
&= \sum_{i=1}^{n}\sum_{j=0}^{1} P\left(\zeta_i = j|T > t_i, \boldsymbol{\Psi}^{(k-1)}\right)^{\delta_i} \\
&\quad \cdot P\left(\zeta_i = j|T = t_i, \boldsymbol{\Psi}^{(k-1)}\right)^{1-\delta_i} l_c\left(\boldsymbol{\Psi}|t_i,\delta_i,\zeta_i\right) \\
&= \sum_{i:\delta_i=0} \Bigg[\frac{1-\pi^{(k-1)}}{1-\pi^{(k-1)}+\pi^{(k-1)}e^{-\lambda^{(k-1)}t_i}}\log\left(1-\pi\right) \\
&\quad + \frac{\pi^{(k-1)}e^{-\lambda^{(k-1)}t_i}}{1-\pi^{(k-1)}+\pi^{(k-1)}e^{-\lambda^{(k-1)}t_i}}\left(\log\pi - \lambda t_i\right)\Bigg] \\
&\quad + \sum_{i:\delta_i=1}\left[\log\pi + \log\lambda - \lambda t_i\right].
\end{aligned}
\tag{18}
$$

Finally the formulas (7) and (8) are achieved by solving the roots of the partial derivatives of (18) with respect to π and λ separately.

REFERENCES

[1] "2018 year in review," https://adindex.ru/files2/access/2019_01/230617_SuperData%202018%20Year%20in%20Review.pdf, SuperData, A Nielsen Company, 2019, accessed: May 2019.

[2] AppsFlyer, "The state of in-app spending," http://cdn2.hubspot.net/hubfs/597489/IAP_Guide/The_State_of_In-App_Spending_AppsFlyer.pdf, 2016.

[3] E. Lee, Y. Jang, D.-M. Yoon, J. Jeon, S.-i. Yang, S. Lee, D.-W. Kim, P. P. Chen, A. Guitart, P. Bertens *et al.*, "Game data mining competition on churn prediction and survival analysis using commercial game log data," *IEEE Transactions on Games*, 2018.

[4] U. Endriss and J. Leite, "Predicting players behavior in games with microtransactions," in *STAIRS 2014: Proceedings of the 7th European Starting AI Researcher Symposium*, vol. 264. IOS Press, 2014, p. 230.

[5] H. Xie, S. Devlin, D. Kudenko, and P. Cowling, "Predicting player disengagement and first purchase with event-frequency based data representation," in *2015 IEEE Conference on Computational Intelligence and Games (CIG)*. IEEE, 2015, pp. 230–237.

[6] R. Sifa, F. Hadiji, J. Runge, A. Drachen, K. Kersting, and C. Bauckhage, "Predicting purchase decisions in mobile free-to-play games," in *Eleventh Artificial Intelligence and Interactive Digital Entertainment Conference*, 2015.

[7] T. Debeauvais and C. V. Lopes, "Gate me if you can: The impact of gating mechanics on retention and revenues in jelly splash." in *International Conference on the Foundations of Digital Games*, 2015.

[8] M. Viljanen, A. Airola, J. Heikkonen, and T. Pahikkala, "Playtime measurement with survival analysis," *IEEE Transactions on Games*, vol. 10, no. 2, pp. 128–138, 2018.

[9] P. S. Fader and B. G. Hardie, "Probability models for customer-base analysis," *Journal of interactive marketing*, vol. 23, no. 1, pp. 61–69, 2009.

[10] N. Hanner, K. Heppner, and R. Zarnekow, "Counting customers in mobile business-the case of free to play." in *PACIS*, 2015, p. 174.

[11] M. Viljanen, A. Airola, A.-M. Majanoja, J. Heikkonen, and T. Pahikkala, "Measuring player retention and monetization using the mean cumulative function," *arXiv preprint arXiv:1709.06737*, 2017.

[12] S. Voigt and O. Hinz, "Making digital freemium business models a success: Predicting customers lifetime value via initial purchase information," *Business & Information Systems Engineering*, vol. 58, no. 2, pp. 107–118, 2016.

[13] D. R. Cox and D. Oakes, *Analysis of survival data*. Chapman and Hall Ltd, 1984.

[14] D. F. Moore, *Applied Survival Analysis Using R*. Springer International Publisher Switzerland, 2016.

[15] R. Sifa, C. Bauckhage, and A. Drachen, "The playtime principle: Large-scale cross-games interest modeling," in *2014 IEEE Conference on Computational Intelligence and Games*. IEEE, 2014, pp. 1–8.

[16] A. Isaksen, D. Gopstein, and A. Nealen, "Exploring game space using survival analysis." in *International Conference on the Foundations of Digital Games*, 2015.

[17] R. G. Miller Jr, *Survival analysis*. John Wiley & Sons, 1981, vol. 66.

[18] J. Klein, H. van Houwelingen, J. Ibrahim, and T. Scheike, *Handbook of Survival Analysis*, ser. Chapman & Hall/CRC Handbooks of Modern Statistical Methods. CRC Press, 2016. [Online]. Available: https://books.google.fi/books?id=t1vOBQAAQBAJ

[19] X. Liu, *Survival Analysis: Models and Applications*. Higher Education Press, 2012.

[20] G. McLachlan and T. Krishnan, *The EM algorithm and extensions*. New York: John Wiley & Sons, 1997.

[21] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 39, no. 1, pp. 1–38, 1977. [Online]. Available: http://www.jstor.org/stable/2984875

[22] D. Hiemstra, *Probability Smoothing*. Boston, MA: Springer US, 2009, pp. 2169–2170. [Online]. Available: https://doi.org/10.1007/978-0-387-39940-9_936

[23] H. Akaike, "A new look at the statistical model identification," *IEEE Transactions on Automatic Control*, vol. 19, no. 6, pp. 716–723, 1974.

[24] J. G. Ibrahim, M.-H. Chen, and D. Sinha, *Bayesian survival analysis*. Springer Science & Business Media, 2001.

[25] C. F. J. Wu, "On the convergence properties of the EM algorithm," *The Annals of statistics*, pp. 95–103, 1983.