A Statistical Analysis of Player Improvement and Single-Player High Scores

Aaron Isaksen and Andy Nealen
Tandon School of Engineering, New York University
aisaksen@appabove.com and nealen@nyu.edu

ABSTRACT

We present the analytical and empirical probabilities of a player achieving a single-player high score after playing a series of games. Using analytical probabilities, simulated game data, and actual game analytics data from two popular mobile games, we show that the probability of reaching a high score decrease rapidly the more one plays, even when players are learning and improving. We analyze the probability of beating the previous k scores, placing on a Top m Leaderboard, completing a streak of k consecutively increasing scores, beating the mean score, and introduce a metric called "decaying high score" that is parameterized and easier for players to achieve. We show that players exhibit different types of learning behavior, which can be modeled with linear or power-law functions – but that in many conditions skill improvement happens too slowly to affect the probability of beating one's high score.

INTRODUCTION

High scores – the maximum score that a player has achieved – appear commonly in many video games. For single player games, they are often used as a motivator to encourage players to keep playing a game, and as a metric to demonstrate improvement and mastery (Elias et al. 2012; Schell 2014; Trepte and Reinecke 2011). High scores can also add a competitive component to single player games, allowing players to compete by comparing their scores.

In this paper, we focus on high scores when used as a metric for single-player games. We analyze the probability of obtaining a high score, showing that it rapidly decreases the more one plays a game. Figure 1 shows examples of single-player scores achieved by playing a game 1000 times – we show data for both a simulated game (Nelson 2011) and actual player data obtained via game analytics (El-Nasr et al. 2013). Each point represents the final

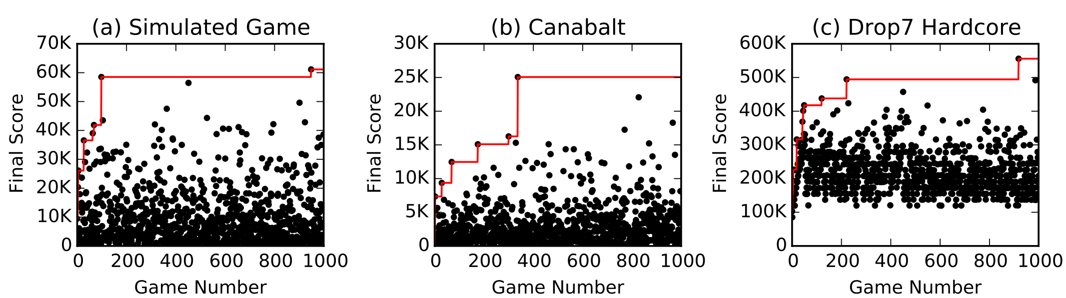

Figure 1: High scores (red line) are rarely set as more games (black dots) are played. (a) Simulated data assuming an exponential score distribution. (b,c) Scores captured from game analytics for Canabalt and Drop7 Hardcore.

Proceedings of 1st International Joint Conference of DiGRA and FDG

score achieved, and the red line shows the current high score for each player. Except for the beginning when a player sets high scores quite often, a new high score is set quite rarely.

This implies that it becomes increasingly difficult to beat one's best score. If high scores are used as encouragement to keep playing, designers should at least be aware that achieving the goal becomes increasingly rare. Therefore, we present and analyze several other metrics that could be used for motivating players with score-based goals: beating the previous score, beating the previous k scores, number of high scores achieved, placing on a Top m Leaderboard, completing a streak of k consecutively increasing scores, and beating the mean score. We also introduce a new metric called "decaying high score" which is a parameterized, designer-tunable metric that becomes easier to reach the more the player fails to reach it. We also present simulated and actual game data recorded from two popular mobile games to verify our results and show where empirical data diverges from our models.

We do not make any claims here about which of these metrics would be the best to include in a game (Desurvire et al. 2004), or how to effectively motivate or reward players (Hamari et al. 2014; Bowey et al. 2015). Instead, our goal is to present several options that a designer might wish to incorporate into their games, armed with a hopefully deeper understanding of how often a player will achieve those goals. We use data-driven player modeling (A. M. Smith et al. 2011; Yannakakis and Togelius 2011) for the goal of helping designers analyze and craft better games (G. Smith et al. 2011; Yannakakis et al. 2014).

High scores are related to *extreme value theory* (Reiss and Thomas 2001), a branch of statistical analysis for predicting the outcome of a maximal event occurring over a specific period of time. It is often used to model the likelihood of floods, severe events in financial markets, or the fastest time for humans to run a race (Einmahl and Magnus 2008). For example, one could use a generalized extreme value distribution (Coles et al. 2001) to obtain a model of the probability of a player beating a chosen score over any given time period. In this paper we instead focus on a player's single-player experience and the probability that they will achieve a new high score on the next play, based on their current series of scores – without being concerned what exact value that high score might be.

Scores can also be used to measure player improvement and learning. For the games we analyzed, much of the player improvement happens either so slowly it does not significantly impact the player's ability to achieve a high score, or happens mainly in the first few plays of the game. After many plays, player improvement is relatively minor and effectively negligible. If there is no learning or slow learning, we can model the scores as independently and identically distributed (i.i.d.), which is considerably simpler to model. Under conditions of rapid learning, we have independently and *non-identically* distributed distributed data, which is more complicated to model because it involves time-varying probabilities. When possible, we analyze the more general case when a player exhibits learning and we can quantitatively describe improvement with a skill acquisition model (Lane 1987).

High scores were originally used in the 1930's as target scores to reward pinball players with prizes for reaching a particular score. *Space Invaders* (1978) was the first video game to save the single highest score in the context that we use today, allowing people to return to the same machine to compete against each other for the single top score[1] and *Asteroids* (1979) introduced the modern Top 10 Leaderboard (Symonds 2010). *Star Fire* (1978) introduced storing three initials for each player[2] and *Defender* (1980) included a AAA-battery powered memory to semi-permanently record the eight "All-Time Greatest" for the machine even if the machine was unplugged, as well as non-permanent "Today's Greatest" leaderboard – allowing new players a chance at placing on the leaderboard (Edge Magazine 2003). *Twin Galaxies*, started in 1981, permanently records the worldwide top scores for each game for official all-time high scores (Day 1997).

METHODS

For each of the metrics discussed in this paper, our general strategy is to first analyze the theoretical probability of a specific event occurring, such as achieving a new high score or beating one's last score. We can represent the chance that a player achieves a specific final score in a game as a probability distribution (Isaksen et al. 2015).

We then compare our analytical models to simulated player data and actual player data. Simulated player data provides a base to explore various score distributions as well as the contribution of different theoretical rates of learning. Unless otherwise indicated, the simulations assume exponentially distributed scores, which shows up in games of constant difficulty (Isaksen and Nealen 2015). When modeling learning and improvement, we adjust the probability distribution after each game to increase the simulated player's expected final score – this is the mean of the probability distribution.

Actual player data is essential for verifying the accuracy of our models. For player data obtained via game analytics, we used a collection of scores obtained from two independently developed video games. The first dataset uses scores obtained between 2009 and 2012 for Canabalt (Saltsman 2009), a mobile one-button infinite runner action game that focuses on dexterity skill and accurate timing of jumps in a procedurally generated scrolling map. The second data set uses scores obtained between 2009 and 2010 from the Hardcore mode of the original mobile version of Drop7 (Area/Code Entertainment 2009), a mobile turn-based puzzle game on a 7x7 grid where the player stacks numbered circles making rows and columns that contain the indicated same number of circles; clearing circles requires careful strategic planning as typical in many deep puzzle games. Ideally we would desire to examine more games, but there were only two datasets available to us which contained every score achieved, organized by device ID and timestamp. In contrast, datasets obtained from popular analytics sites such as Google Analytics or Flurry do not provide access to every score achieved – they mainly provide aggregate data over time which would not allow us to perform the type of analysis we use in this paper. These two games have many differences in their scoring systems, real-time vs turn-based, strategic planning vs dexterity, etc. and therefore provide two unique and independent case studies that show the applicability of our analysis to a wide variety of games. In particular, our analysis does not always require knowledge of the underlying game rules or player skill distributions and so is less likely to be affected by the particular games chosen to validate the models.

1. The high score in *Space Invaders* was lost when the machine was powered off.
2. *Star Fire* was actually the first game to store more than one high score, but used a non-standard format: there was only one high score slot per number of coins/continues used to achieve the score (Exidy Inc. 1979).

For each reported score, we assume all scores with the same device ID come from the same player – this is a commonly accepted practice from game analytics of this era in mobile gaming where people typically only had one device. We are not comparing scores between Drop7 and Canabalt, or making any attempt to match up players who may have played both games. We then sort each player's scores by the timestamp, giving us a series of scores $X_p = x_{p,1}, x_{p,2}, ...x_{p,n_p}$ for each unique player identifier p in the order they achieved them.

Each player p has played a different number of games n_p, equal to the length of X_p. Because players all have a different number of games and we want to maximize the number of scores we can study, we perform the analysis on the top N players, sorted by decreasing n_p. N varies between 200 and 1000, depending on the metric we are examining, and is chosen to (a) make sure the figures can clearly demonstrate the empirical probabilities and (b) so that $n_p > 100$ for enough plays to see trends in the data. For our simulated games, $N = 1000$ and $n_p = 1000$ for all simulated players. Using k-fold cross validation, we can show that the models are robust to selecting different subsets of the N games.

We had to perform some cleaning of the data before analyzing – in particular we were not able to examine some Canabalt players. The Canabalt game analytic system would always sort any batched offline scores in decreasing order no matter what order they were actually played in, so we threw out any of the players with clearly modified data.

HOW SCORES CHANGE OVER TIME

As players repeatedly play a game, they become better at it. We can examine the results of players playing the same game multiple times to see how their scores improve *on average* over time. We examine two models of learning commonly discussed in the skill acquisition literature: linear learning and power-law learning. Linear learning can be modeled by an average score μ_{linear} increasing linearly; Power-law learning can be modeled by an average score μ_{power} increasing as a negative-exponent power law:

$$\mu_{linear}(t) = at + b \qquad\qquad \mu_{power}(t) = A - Bt^{-r} \tag{1}$$

For μ_{linear}, the average minimum score b increases by a on each game. For μ_{power}, A is the asymptotic average score, with learning rate $r > 0$ and average minimum score $A - B$.

To see how actual game scores improve over repeated plays, we examine the top $N = 500$ players ranked by number of plays. We take the average score of all players achieving a score on game t. In Figure 2 we plot the average value over all N players for all game numbers t:

$$\text{average score on game } t = \frac{1}{N}\sum_{p=1}^{N} x_{p,t} \tag{2}$$

Figure 2 demonstrates how average scores increase with the number of games played. Comparing the simulated game data on the left with the actual game data on the right, we can see that Canabalt exhibits an approximately linear improvement, while Drop7 Hardcore exhibits an approximate power-law curve. We fit the models with ordinary least squares regressions, indicated with the dashed lines; because we are fitting mean scores, the central limit theorem justifies assuming normally distributed residuals. We do not have a predictive

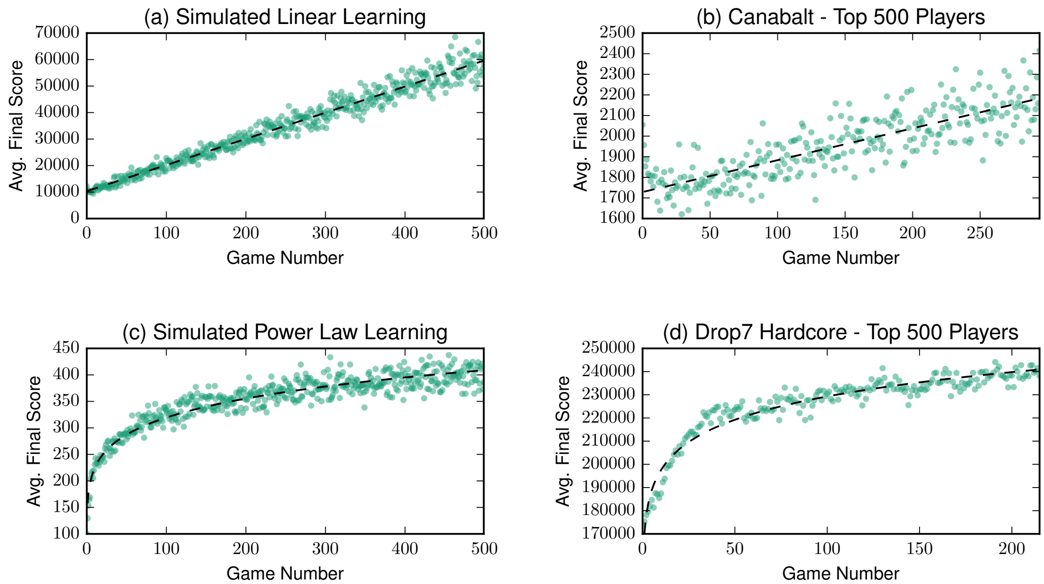

Figure 2: Average scores improve as players play more games, due to learning. Best fit least-squares regressions drawn with dashed lines. (a) Simulated exponentially distributed scores with linear learning ($R^2 = 0.967$). (b) Canabalt exhibits approximately linear learning ($R^2 = 0.675$). (c) Simulated scores with a power law learning ($R^2 = 0.914$). (d) Drop7 exhibits approximately power law learning ($R^2 = 0.929$).

model for why each game exhibits a specific type of learning, but it is known that different tasks will exhibit different learning rates (Lane 1987). We also note the two games are in different game genres and require different types of skill. Based on the learning rates visible in the game data, we aim to model learning into our equations for a better fit.

PROBABILITY OF BEATING THE PREVIOUS SCORE

We now calculate the analytical probability of a player beating their previous score on average, without specifying their exact previous score. Let $X = x_1, x_2, ..., x_n$ be the series of scores that the player has achieved. During design, we don't know what these values will be, so we calculate the probabilities for all players of the game, not any specific player. Let $f(x,t)$ be the probability of achieving score x on game number t. Let $F(x,t)$ be the cumulative distribution function (CDF) of $f(x,t)$ (the probability of achieving a score $< x$ on game number t). Without specifying the previous score, the probability of beating one's immediately previous score on game t is the probability of a score x on game t times the probability of score $< x$ on the previous game $t-1$, integrated over all possible scores:

$$Pr[\text{beat previous score at game } t] = \int_0^\infty F(x, t-1) f(x,t) dx \tag{3}$$

This is not a joint probability, because t is not a random variable, and we are free to pick any particular pair of games t and $t-1$. Note that we assume we don't know the particular score the player might achieve on game $t-1$. However, if one does know the previous score x_{t-1}, then the probability of beating the previous score is equal to $1 - F(x_{t-1}, t)$.

To evaluate Eq. 3, we first handle the case where the player is improving very slowly over consecutive games and therefore has an effectively constant skill level, so their score probability distribution is independently and identically distributed (i.i.d.) and not dependent on t. Thus, we can eliminate the t variable so $f(x,t)$ becomes $f(x)$ and $F(x,t-1)$ becomes $F(x)$. Under this condition, and knowing that by definition of a CDF that $F(\infty) = 1$, $F(0) = 0$, and the first derivative of $F(x)$ is $f(x)$, we simplify Eq. 3 to:

$$Pr[\text{beat previous score} \mid \text{no learning}] = \int_0^\infty f(x)F(x)dx = \frac{F(x)^2}{2}\bigg|_0^\infty = \frac{1}{2} \qquad (4)$$

This shows the probability of beating one's previous score is always 1/2, if the scores are i.i.d. This is independent of the underlying score distribution $f(x)$ and the hazard rate which describes how the game increases in difficulty.

If the player is improving over time, we can't eliminate the t variable, so the simplification of Eq. 4 can not apply. To evaluate Eq. 3, we need the analytical equations for the underlying distributions $f(x,t)$ and $F(x,t)$, which models how the player performs and improves over time. To begin, we first discuss the case where the score distribution is exponential, as happens in games with constant difficulty. The time-varying probability distribution f_{exp}, cumulative distribution F_{exp}, and mean score μ_{exp} can be written as:

$$f_{exp}(x,t) = \lambda_t e^{-\lambda_t x} \qquad F_{exp}(x,t) = 1 - e^{-\lambda_t x} \qquad \mu_{exp}(t) = 1/\lambda_t \qquad (5)$$

Figure 3 demonstrates how the decay rate λ_t, normally a constant λ, is now time varying and thus changes as t increases. Fig. 3(a) shows how increasing the mean score $\mu_{exp}(t)$ affects the exponential probability distributions. As the mean score increases, the distribution flattens out – low scores become less likely and high scores become more likely. The figure also shows how using a (b) linear model and (c) power-law model (as defined in Eq. 1) change the mean score and λ_t. As a player improves, their mean score $\mu_{exp}(t)$ increases; this implies that λ_t decreases as a player improves.

Now, returning to Eq. 3, we substitute the equations for time-varying exponential $f(x,t)$

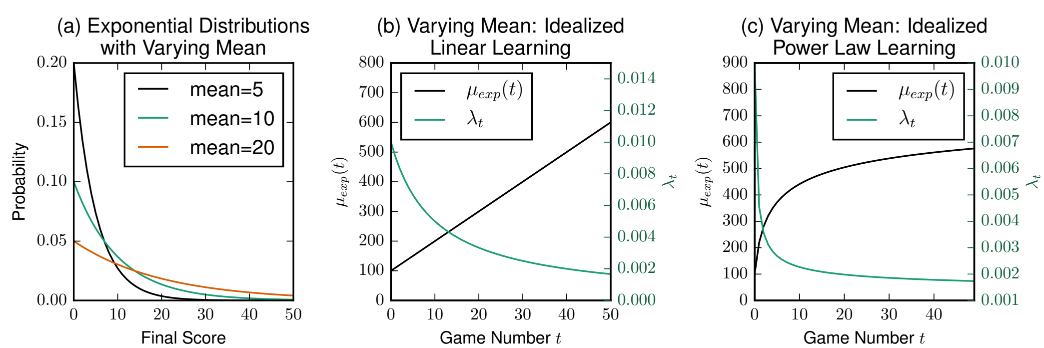

Figure 3: Varying mean score to model learning for simulated games. (a) Increasing the mean μ changes the exponential score probability distribution defined by $\lambda_t = 1/\mu$. (b) Linearly increasing $\mu_{exp}(t)$ reduces λ_t. (c) Power-law increasing learning model where learning slows down over time.

and $F(x,t)$ from Eq. 5. Using a symbolic integrator such as Mathematica, we get:

$$Pr[\text{beat prev. score} \mid \text{exponential dist.}] = \int_0^\infty \lambda_t e^{-\lambda_t x}(1 - e^{-\lambda_{t-1} x})dx = \frac{\lambda_{t-1}}{\lambda_t + \lambda_{t-1}} \quad (6)$$

We can examine how Eq. 6 might change given different learning rates – i.e. different functions of λ_t. If we have a minimal amount of learning, then $\lambda_t \approx \lambda_{t-1}$ and Eq. 6 reduces to 1/2, the result of Eq. 4. With more significant linear learning, as described in Eq. 1, then $\lambda_t = 1/(at+b)$. In this case, Eq. 6 reduces to:

$$Pr[\text{beat prev. score} \mid \text{exp. dist., linear learning}] = \frac{\frac{1}{a(t-1)+b}}{\frac{1}{at+b} + \frac{1}{a(t-1)+b}} = \frac{1}{2 - \frac{a}{at+b}} \quad (7)$$

We can see that for all $a, b > 0$, the above equation is always $> 1/2$, and as $t \to \infty$, it approaches 1/2. For small values of t, learning increases the probability of a player beating their previous score. However, as the player plays more games, linearly improving learning has diminishing impact on the probability of a player beating their previous score, assuming exponentially distributed scores. We can also see that in the case of no learning, when $a = 0$, or limited learning, when $a \ll b$, the above equation also reduces to 1/2. Thus, in many conditions, assuming i.i.d. scores is a reasonable approximation.

If the mean score were to increase as a power law, then Eq. 6 becomes:

$$Pr[\text{beat prev. score} \mid \text{exp. dist., power law learning}] = \frac{1 - (B/A)t^{-r}}{2 - (B/A)(t^{-r} + (t-1)^{-r})} \quad (8)$$

As the number of games $t \to \infty$ or the learning rate $r \to \infty$, then $t^{-r} \to 0$, and the above equation approaches $1/2$. Also, if the learning rate is very small, such that $r \to 0$, then we also approach 1/2. Therefore, a small learning rate or a very large learning rate has little impact on the probability of beating one's previous score after playing several games.

For both linear learning and power-law learning, we see that player improvement mostly matters in early games, and has relatively little impact as more games are played. The learning effect, in the limit of many games, is approximately zero. Thus, i.i.d. scores can be a reasonable assumption under certain conditions, which allows us to analyze much more complex probabilities in the following sections.

PROBABILITY OF REACHING A HIGH SCORE

We now proceed to the case of beating all of one's previous scores – the condition for achieving a single-player high score. We calculate the analytical probability of achieving a high score on game n. If we know that x is the high score set on game n, then we must have all the other $n-1$ scores be less than x. The probability of setting a high score on game n is therefore the probability of setting a score of x on game n times all the cumulative probabilities of setting a score $< x$ on the other $n-1$ games, integrated over all values of x:

$$Pr[\text{high score on game } n] = \int_0^\infty f(x,n) \prod_{t=1}^{n-1} F(x,t)dx \quad (9)$$

In the cases where there is approximately no learning, then the probabilities are not dependent on t and we replace $f(x,t)$ with $f(x)$ and $F(x,t)$ with $F(x)$. Given $F(\infty) = 1$, $F(0) = 0$, and the first derivative of $F(x)$ is $f(x)$ for all probability distributions, we get:

$$Pr[\text{high score on game } n \mid \text{no learning}] = \int_0^\infty f(x)F(x)^{n-1}dx = \left.\frac{F(x)^n}{n}\right|_0^\infty = \frac{1}{n} \quad (10)$$

This result says that the more one plays a game, the less likely they will beat their best score. Given the importance that game designers often give to the high score (e.g. displaying it on the top of the screen or at the end of each game), this reward becomes increasingly more difficult for all players to achieve.

Eq. 9 can be solved for cases where there is a significant amount of learning, but the solution is quite complicated due to non-identically distributed scores. Due to the complexity of the result, the analytical solution does not provide much insight and we do not present it here. However, since learning would be expected to make it easier to perform better and therefore increase the probability of getting a high score, we can assume a simple linear correction factor $\kappa \geq 1$, which allows us to replace $1/n$ with κ/n when fitting data. We can then use a linear regression to solve for κ to see the overall effect that learning causes us to diverge from the standard model. As before, because we are dealing with mean scores, the central limit theorem justifies the use of least squares to fit normally distributed residuals.

We now analyze the empirical data containing player improvement. We verify the results of this section by examining the probability of achieving a high score on each game using simulated games as well as actual score data, as shown in Figure 4. The plots on the left show the actual probability of achieving a high score after n games, while on the right we display the same probabilities in log-log plots; a function that follows a power law, such as $1/x = x^{-1}$, will show up as a straight line in a log-log plot.

For the simulated games in Fig. 4(a) and Fig. 4(b), with no learning and independently and identically distributed (i.i.d.) scores, we generate a series of N scores for each of M simulated players using a chosen probability distribution. For each of the series of scores, we calculate when a new single player high score is achieved. We store the game number when each high score is achieved into a single histogram, with one bin for each game number. By dividing the histogram bin values by M, each bin k tells us the probability of achieving a high score on game k. We then display these probabilities vs the current game number. We can see that the observed values from the simulated games closely match the expected distribution, no matter what underlying chosen score distribution is used. We used $M = 1000$ and $N = 100$ to generate these figures. Using a least squares linear regression to fit κ/n to the exponential data, we find $\kappa = 0.9983$ with 95% confidence interval (CI) [0.990, 1.006], indicating that our model $1/n$ is a very good fit. Similarly for the uniform data, we fit $\kappa = 0.9927$ (CI [0.984, 1.001]), also a good fit for the $1/n$ model.

For the plots in Fig. 4(c) and Fig. 4(d), we use real game data from Canabalt and Drop7 Hardcore mode to generate the same plots. Again, we use the most active 1000 players, showing the first $N = 100$ scores for each player. For Canabalt, we fit $\kappa = 1.0293$ (CI [1.014, 1.044]) and for Drop7 Hardcore we fit $\kappa = 1.0682$ (CI [1.018, 1.119]) also indicating our $1/n$ model only requires a small correction, which can be explained by non-i.i.d scores.

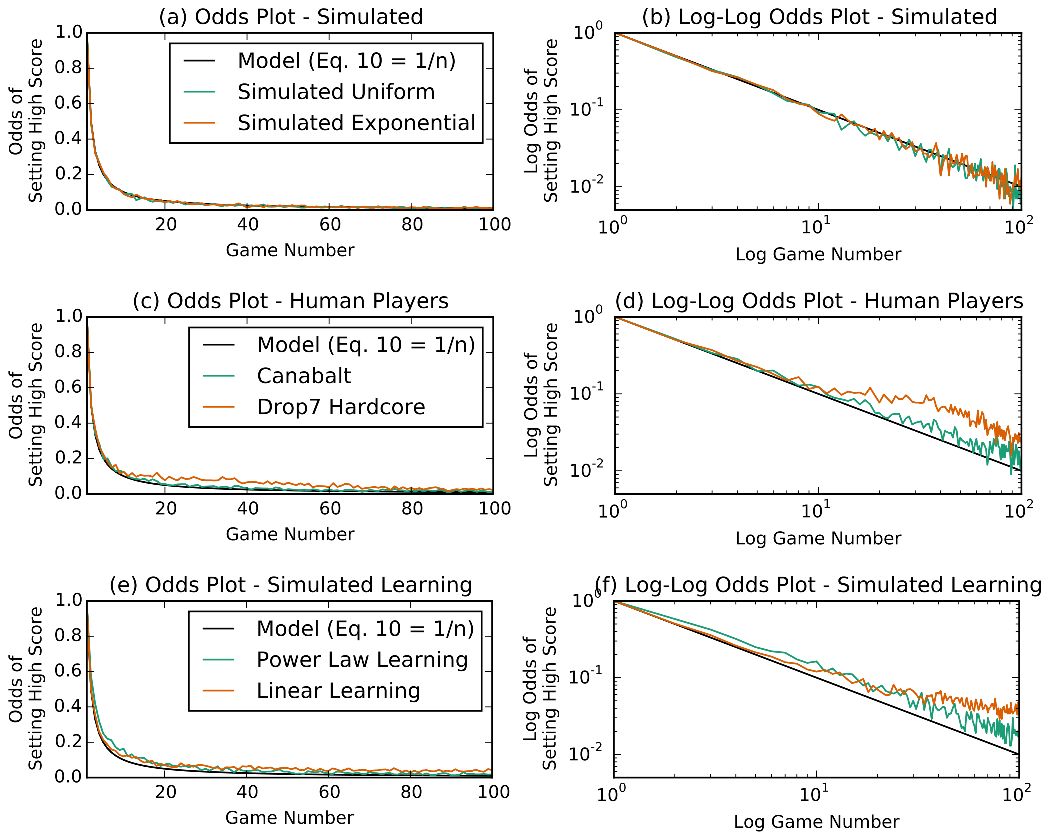

Figure 4: Probability of achieving a high score after n games decreases as $1/n$. This shows up as a hyperbola on the left and linearly in the log-log plots on the right. (a,b) Simulated games with no learning, no matter the underlying distribution, tightly follow the model. (c,d) For Canabalt and Drop7 Hardcore mode the probabilities slightly diverge. (e,f) Adding learning effects in the simulation have a similar divergence.

For the plots in Fig. 4(e) and Fig. 4(f), we generate sets of simulated exponentially distributed scores, but increase the mean value for the exponential distribution by a linear learning model and a power law learning model. We see that this gives rise to a similar departure from the model $1/n$, indicating that learning effects are likely responsible for this type of departure. Linear learning fits a correction of $\kappa = 1.1094$ (CI [1.076, 1.143]) and power law learning fits a smaller correction of $\kappa = 1.0567$ (CI [1.015, 1.099]).

From a design perspective, a designer can increase the probability that a player will achieve the goal of setting a high score by resetting the high score so that n can not get too large. For example, the designer could use a rolling window and only compare the high score for the last k games: then the probability of setting a new maximum would be $1/k$ (so if the game gave the high score for the last 10 games, the probability of setting it would be 10%). Another approach to limiting n, which is often used in practice, is to reset the high score after an elapsed amount of time (e.g. daily, weekly, monthly, or by season). This also makes it more likely that a player will achieve the reward of setting a high score. For example, *Tsum*

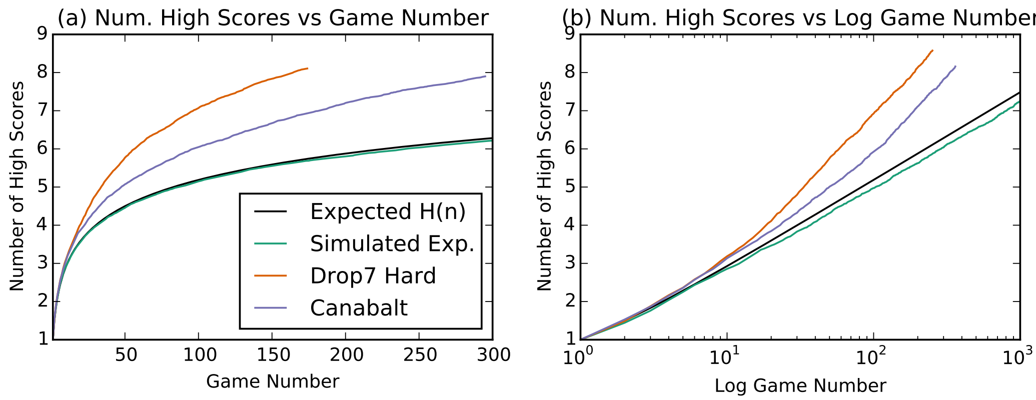

Figure 5: The expected number of high scores achieved after n games is approximately $H(n)$, the harmonic series summing from $1/1$ to $1/n$. Although actual player data diverges from this, the average actual number of high scores achieved is still within about 2 even after 200 games.

Tsum, a popular match-3 mobile game, uses a weekly high score, *Spelunky* has a "daily" mode where a unique map is generated each day and the scores are only presented for that day, and *Defender*'s "Today's Greatest" leaderboard is also reset daily.

NUMBER OF HIGH SCORES ACHIEVED

Given i.i.d. scores, the expected number of high scores set after n plays is $H(n)$, the harmonic series sum up to $1/n$. It can be shown that $H(n) \approx \ln(n) + \gamma + 1/2n$ where $\gamma \approx 0.5772...$ is the Euler-Mascheroni constant (Sondow and Weisstein n.d.). Thus, the number of high scores achieved after n plays is $O(\ln n)$. To prove this, we need to show that $H(n)$ is the expected number of high scores, assuming i.i.d. scores. The expected value is the sum of probabilities of achieving a high score on each game, so we calculate the probability of achieving a high score after 1 game, 2 games, and so on up to n games. Given that the probability of achieving a high score after N games is $1/N$, we have $1/1 + 1/2 + 1/3 + ...1/n$, which is the definition of the harmonic series. Thus the expected number of high scores achieved by game n is:

$$E[\text{number high scores achieved by game } n] = H(n) = \sum_{k=1}^{n} 1/k \tag{11}$$

Figure 5 shows how simulated data and actual data conforms to the expected number of high scores. For simulated scores with no learning, the number of high scores tracks the expected $H(n)$ very closely. For actual game data, with learning, the expected number of high scores does diverge from $H(n)$. However, the rate at which it increases is very slow – we show in the figure that even after 200 games the divergence is about 2 high scores. Thus, unless the player is improving at a very rapid rate, learning has a negligible effect on the number of high scores achieved.

LEADERBOARDS

Many games report and store more than one high score, often called a top-ten list or a leaderboard. For approximately no learning and i.i.d. scores, we can show that the probability of placing on the leaderboard is related to the size of the leaderboard m and the number of games n that have been played.

$$Pr[\text{placing on leaderboard of size } m \mid \text{no learning}] = \begin{cases} 1 & \text{if } n \leq m \\ m/n & \text{if } n > m \end{cases} \tag{12}$$

To prove this combinatorially, lets first consider the first m games. Since the m-leaderboard starts empty, the first m games are guaranteed placement on the leaderboard. After playing a game n times, we have a series of scores $x_1, x_2, ...x_n$. For x_n to place on the leaderboard, we need x_n to be in the top m scores in the series. However, we don't care about the actual scores for each of the games in the series, only the relative rankings between them. So we can give each score x_i a rank r_i between 1 and n, with 1 signifying the highest score in the series. There are n! possible ways the scores could be ranked. However, we require $r_n \leq m$ because there are m possible ranks which would get the player on the leaderboard. Once we have this condition, there are $(n-1)!$ ways to rank the other games in the series. This gives us $m(n-1)!/n! = m/n$ ways to place on the leaderboard once the leaderboard is full.

Calculating a closed-form analytical solution for the probability under non-i.i.d learning conditions is quite difficult, because of the many ways of ordering the scores which filled the leaderboard. Figure 6 empirically demonstrates the accuracy of the model under various learning conditions and for different sized leaderboards. Lines represent the idealized probability for reaching a leaderboard assuming approximately no learning and i.i.d. scores, while dots indicate the average probability calculated from simulated and actual game play data. Simulated exponentially distributed scores match the model, but with learning, the probability of reaching the leaderboard significantly improves for small values of n. This effect decreases as n gets larger. Canabalt matches the ideal model relatively closely while Drop7 Hardcore exhibits a significant departure early. As n grows, all games more closely approximate the ideal model of Eq. 12.

PROBABILITY OF ACHIEVING A STREAK

A *scoring streak* or *streak* occurs when a player beats their previous score on consecutive plays. If we have a series of k scores $x_0, x_1, ...x_k$ such that $x_0 \leq x_1 \leq ... \leq x_k$, then we have a streak of length k. By this definition, the shortest streak is $k = 1$, which occurs when beating one's previous score.

We analyze the probability of achieving a streak of length k, assuming that the scores are i.i.d. This means that the player can not be intentionally trying to score low in order to make it easier to get a streak. Also, the player can't be receiving score bonuses for playing longer. We will show the probability of achieving a streak of length k in the next k games is:

$$Pr[k \text{ streak} \mid \text{no learning}] = 1/(k+1)! \tag{13}$$

We prove by counting the possible ways that the scores $x_0...x_k$ could be sorted; there are $(k+1)!$ possible permutations. Only one of these permutations has a streak of length k, namely the one which is $x_0 \leq x_1 \leq ... \leq x_k$. Thus, the probability is $1/(k+1)!$ of achieving a run of length k, no matter what the underlying score distribution is for the game.

The probability of extending a k-streak to a $k+1$-streak, given that the player has already achieved a k-streak, is $1/(k+1)$. This is the same result of determining the chances of

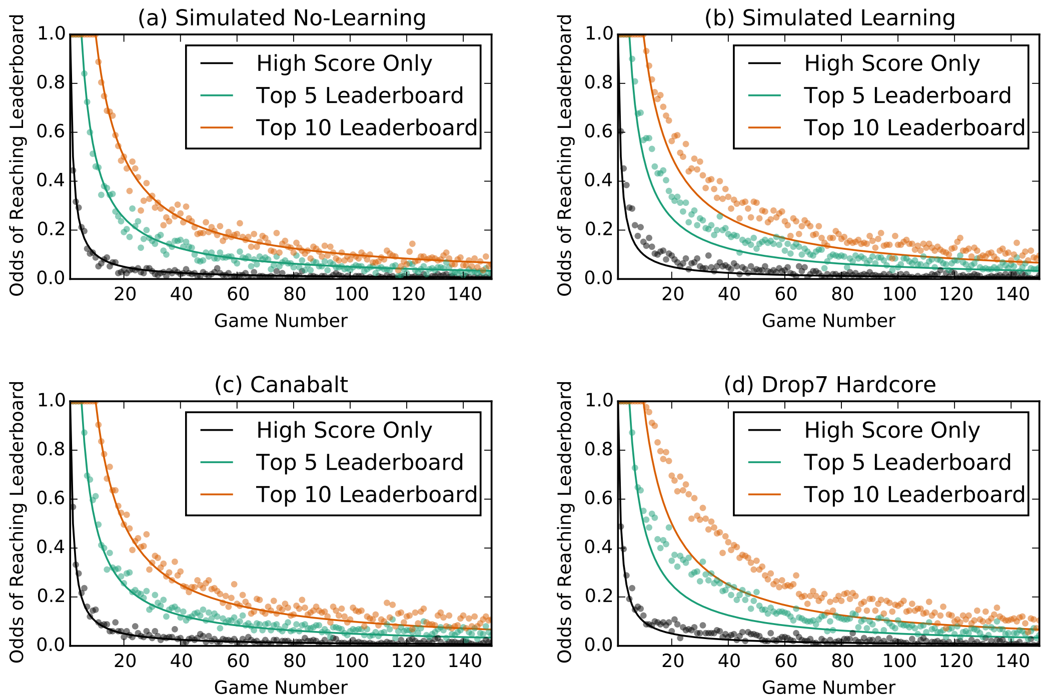

Figure 6: Leaderboard probability plots for varying leaderboard sizes. Lines represent ideal model assuming i.i.d. scores. Dots indicate average probability calculated from simulated and actual game play data. (a) Simulated exponentially distributed scores match the model closely. (b) With learning, the probability of placing improve for small n. (c) Canabalt matches the ideal model while (d) Drop7 Hardcore exhibits a significant departure for small n.

setting a high score for $k+1$ games. In order to extend the streak by 1 game, the most recent score must be larger than all previous k scores.

In Table 1, we can see how longer streaks become increasingly more difficult to achieve. 1-streaks happen 50% of the time, but 2-streaks (16.7%), 3-streaks (4.1%), and 4-streaks (.83%) seem most practical for in-game rewards.

In Figure 7, we verify the probability of achieving a streak with simulated and actual game data. We generate a histogram of score streaks and compare to a model histogram. The model matches the analytically expected probability from Eq. 13. Since the bins become exceedingly small as k grows, we only will use bins up to $k = 5$. For each game we process 500 players, counting all score streaks into a histogram with one bin for each k-streak. The model and observed probabilities of achieving a streak are close, using both simulated and real data. Generally, the observed probability closely match the expected value of $1/(k+1)!$. However we can see from Fig. 7(b) that the log-probability show a small unexplained departure for Drop7 Hardcore, which is not likely to be due to learning since it does not show up in the simulated learning case. This warrants more investigation, although the effect is very slight as it barely shows up in Fig 7(a) and is also unlikely to be noticed by a player.

k: streak length	1	2	3	4	5	6	7	8	9
probability	1/2!	1/3!	1/4!	1/5!	1/6!	1/7!	1/8!	1/9!	1/10!
	1/2	1/6	1/24	1/120	1/720	1/5040	1/40320	1/362880	1/3628800
	0.5	0.1667	0.0417	0.0083	0.0014	0.0002	2.48e-05	2.76-06	2.76e-07

Table 1: Probability of achieving a streak length of k.

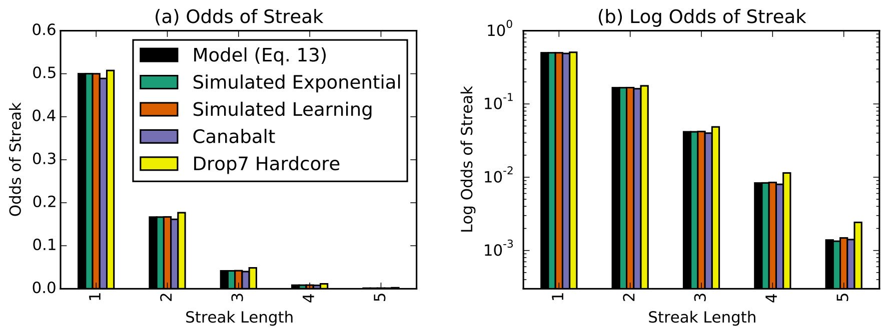

Figure 7: (a) We show the expected and observed probability of achieving a streak, using both simulated and real data. The observed probability closely match the expected value of $1/(k+1)!$. (b) The log-probability shows a small unexplained departure for Drop7 Hardcore, which is not likely to be due to learning since it does not show up in the simulated learning data.

DESIGN SUGGESTION: DECAYING HIGH SCORES

We propose a decaying "score to beat" to make a high score easier to set. If B is the score to beat, and the player surpasses this with a score $s > B$, then s becomes the new score to beat B. Otherwise, B is not reached, and we multiply B by a decay constant $\delta \leq 1$. After k games where B is not reached, the score to beat will be $B\delta^k$. The analytical equation for the probability is difficult to write because it depends on the underlying score distribution, but we can examine its results empirically.

As shown in Figure 8, $\delta = 1$ means no decay, so this is the same as beating a high score. However, as δ decreases, then the probability of beating a decaying high score increases, allowing the game designer to choose the rate at which they want a player to likely reach the decayed high score, as shown in Fig. 8(a,b). In Fig. 8(c,d), we fix the decay constant $\delta = .90$ and try different underlying score distributions – this shows that the probabilities are influenced by the distribution $f(x)$ for decaying high scores.

Although we do not analyze it here, a similar method would be to give players a bonus or score multiplier the more games they play. The amount of multiplier is also designer-tunable, and allows the designer to control how often they would expect a player to receive a reward.

DESIGN SUGGESTION: BEATING THE MEAN

We now consider how often a player would beat their mean score. The probability of the player beating their mean score, that is surviving past the mean score μ, is $Pr[x \geq \mu] = S(\mu) = 1 - F(\mu)$. Because the mean of a distribution depends on $f(x)$, the value of $S(\mu)$ is distribution-dependent. However, we can still analyze the probability of this occurring in various classes of games.

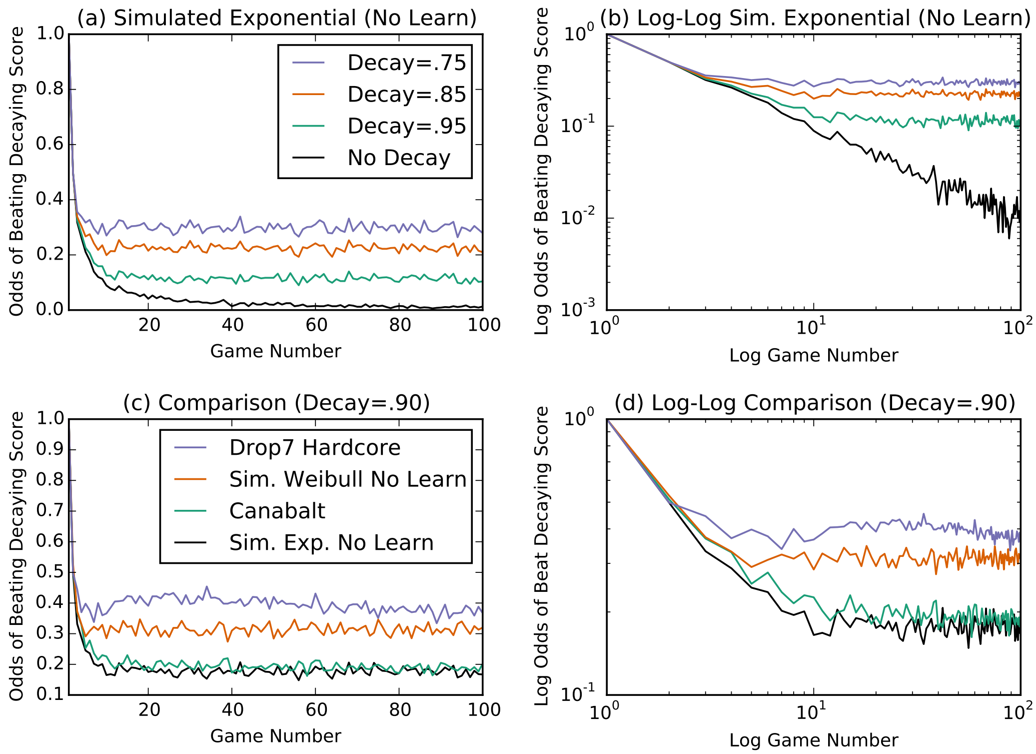

Figure 8: Probability of Achieving a Decaying High Score. (a,b) Decreasing the decay constant increases the probability of achieving a decayed high score; this constant is tunable by the designer. (c,d) Different score distributions react differently to decayed high scores.

If a game has scores that are exponentially distributed, then we can show that this probability is a constant independent of the difficulty tuning parameter λ. Starting with the definition of an exponential distribution $f_{exp}(x) = \lambda e^{-\lambda x}$:

$$S_{exp}(x) = e^{-\lambda x} \qquad \mu_{exp} = 1/\lambda \qquad S_{exp}(\mu_{exp}) = e^{-1} \approx 0.3679... \tag{14}$$

Therefore, the probability of beating the mean score in a exponential distribution have nothing to do with how difficult the game is. Exponentially distributed scores occur in games that have a repeated challenge with a constant level of difficulty, unchanging as the player progresses. If a game has a linearly increasing difficulty, then it can be modeled using a Rayleigh probability distribution $f_{ray}(x) = bxe^{-bx^2/2}$ (Isaksen and Nealen 2015):

$$S_{ray}(x) = e^{-bx^2/2} \qquad \mu_{ray}(x) = \sqrt{\frac{\pi}{2b}} \qquad S_{ray}(\mu_{ray}) = e^{-\pi/4} \approx 0.4559... \tag{15}$$

We see that the probability of beating the mean score for a game with linearly increasing difficulty is independent of the rate at which the game becomes more difficult. The probability of beating the mean for other common probability distributions, such as Weibull or Pareto, depend on the parameters that define the shape of the distributions and do not have constant values independent of the distribution parameters.

Metric	Learning?	Result
Prob. Beating Previous Score	No	1/2
Prob. Beating Prev. Score (after n games)	Yes	$> 1/2$; as $n \to \infty, \to 1/2$
Prob. Beating High Score	No	$1/n$
Prob. Beating High Score	Yes	as $n \to \infty, \to 1/n$
Prob. Beating Last k Scores	No	$1/k$
Number of High Scores Achieved	No	$H(n) = \sum_{k=1}^{n} 1/k$
Number of High Scores Achieved	Yes	$\geq H(n)$
Prob. Reaching a Top m Leaderboard	No	$L(m, n) = \{1 \text{ if } n \leq m \text{ else } m/n\}$
Prob. Reaching a Top m Leaderboard	Yes	$\geq L(m, n)$
Prob. Making a k-Streak	No	$1/(k+1)!$
Prob. Making a k-Streak	Yes	$\approx 1/(k+1)!$
Prob. of Beating Decaying High Score	Yes/No	Depends on $f(x)$ and Decay δ
Prob. of Beating Mean for Exponential $f(x)$	No	$e^{-1} \approx 0.3679...$
Prob. of Beating Mean for Rayleigh $f(x)$	No	$e^{-\pi/4} \approx 0.4559...$

Table 2: Summary of analytical and empirical findings from this paper.

SUMMARY AND FUTURE WORK

To summarize, we present our analytical and empirical findings in Table 2. For designers who wish to include score-based achievements in their single-player games, we hope an understanding of the probability of players achieving those goals can influence the design of those achievements. We also aim to give designers a better understanding of how learning slows down after a relatively small number of games – this means that players could potentially become frustrated at being unable to beat their highest score. To address this, the decaying high scores metric becomes easier to achieve the more a player fails to meet it. Ideally, this would keep a player more interested in playing a game and less likely to quit – they may be improving but because a high score is not reached they may quit in frustration.

We believe that game designers can benefit by using statistical player models to understand what they can and can't control about player experience. The mathematics can be powerful, but much needs to be done to bring these tools into the hands of practicing game designers.

BIBLIOGRAPHY

Area/Code Entertainment. 2009. *Drop7.* Apple App Store.

Bowey, Jason T, Max V Birk, and Regan L Mandryk. 2015. "Manipulating Leaderboards to Induce Player Experience." In *2015 CHI Play,* 115–120. ACM.

Coles, Stuart, Joanna Bawa, Lesley Trenner, and Pat Dorazio. 2001. *An Introduction to Statistical Modeling of Extreme Values.* Vol. 208. Springer.

Day, Walter. 1997. *Twin Galaxies' Official Video Game & Pinball Book of World Records.* Sunstar Publishing, Ltd. ISBN: 1887472258.

Desurvire, Heather, Martin Caplan, and Jozsef A Toth. 2004. "Using Heuristics to Evaluate the Playability of Games." In *CHI'04,* 1509–1512. ACM.

Edge Magazine. 2003. "Score Wars," no. 128: 52–58.

Einmahl, John HJ, and Jan R Magnus. 2008. "Records in Athletics Through Extreme-Value Theory." *Journal of the American Statistical Association* 103 (484).

Elias, George Skaff, Richard Garfield, K Robert Gutschera, and Peter Whitley. 2012. *Characteristics of Games.* MIT Press.

El-Nasr, Magy Seif, Anders Drachen, and Alessandro Canossa. 2013. *Game Analytics: Maximizing the Value of Player Data.* Springer.

Exidy Inc. 1979. *Star Fire Operator's Manual.* The International Arcade Museum.

Hamari, Juho, Jonna Koivisto, and Harri Sarsa. 2014. "Does Gamification Work? – A Literature Review of Empirical Studies on Gamification." In *System Sciences (HICSS), 2014 47th Hawaii International Conference on,* 3025–3034. IEEE.

Isaksen, Aaron, Dan Gopstein, and Andy Nealen. 2015. "Exploring Game Space Using Survival Analysis." In *Foundations of Digital Games.*

Isaksen, Aaron, and Andy Nealen. 2015. "Comparing Player Skill, Game Variants, and Learning Rates Using Survival Analysis." In *First Player Modeling Workshop at AIIDE.*

Lane, Norman E. 1987. *Skill Acquisition Rates and Patterns.* Springer Media.

Nelson, Mark J. 2011. "Game Metrics Without Players: Strategies for Understanding Game Artifacts." In *Artificial Intelligence in the Game Design Process.*

Reiss, Rolf-Dieter, and Michael Thomas. 2001. *Statistical Analysis of Extreme Values.*

Saltsman, Adam. 2009. *Canabalt.* Apple App Store.

Schell, Jesse. 2014. *The Art of Game Design: A Book of Lenses.* CRC Press.

Smith, Adam M, Chris Lewis, Kenneth Hullet, and Anne Sullivan. 2011. "An Inclusive View of Player Modeling." In *FDG,* 301–303. ACM.

Smith, Gillian, Jim Whitehead, and Michael Mateas. 2011. "Computers as Design Collaborators: Interacting with Mixed-Initiative Tools." In *Workshop on Semi-Automated Creativity, co-located with ACM Creativity & Cognition 2011.*

Sondow, Jonathan, and Eric W. Weisstein. n.d. "Harmonic Number." *MathWorld–A Wolfram Web Resource.*

Symonds, Shannon. 2010. *The Changing Face of Winning in Video Games.* `http://www.museumofplay.org/blog/chegheads/2010/11/the-changing-face-of-winning-in-video-games/`.

Trepte, Sabine, and Leonard Reinecke. 2011. "The Pleasures of Success: Game-Related Efficacy Experiences as a Mediator Between Player Performance and Game Enjoyment." *Cyberpsychology, Behavior, and Social Networking* 14 (9): 555–557.

Yannakakis, Georgios N, Antonios Liapis, and Constantine Alexopoulos. 2014. "Mixed-Initiative Co-Creativity." In *Foundations of Digital Games.*

Yannakakis, Georgios N, and Julian Togelius. 2011. "Experience-Driven Procedural Content Generation." *Affective Computing, IEEE Transactions on* 2 (3): 147–161.