Predicting Finishing Rank In Epic's Game Fortnite

Connor Boos
Marquette University
Milwaukee, WI 53233, USA
Connor.Boos@marquette.edu

Joe Chudzik
Marquette University
Milwaukee, WI 53233, USA
Joseph.Chudzik@marquette.edu

Justin Furst
Marquette University
Milwaukee, WI 53233, USA
Justin.Furst@marquette.edu

Shivani Kohli
Marquette University
Milwaukee, WI 53233, USA
Shivani.kohli@marquette.edu

Abstract
The goal of the project is to prove the validity of the algorithm developed for predicting the greatest chance of victory based on the number of kills. The data is collected using an anonymous google form, which has been placed on online forums for the game. The form gathers information regarding overall place/rank, landing location/quadrant, kills, and game mode. This data is collected under the assumption that the players filling out the survey have above average skills in the game due to their dedication to the game. After undertaking data wrangling and cleaning techniques we found our data to have an exponential decay relationship. A model was fitted to this data which helped give an estimate of the rank finished based on the number of kills during the game. To prove the effectiveness of the model we had a separate test and train data set.

Author Keywords
Fortnite; Game; Linear Model; Exponential; Prediction; Survey; Kills; Position; Probability; Win; Data; Analysis; Location.

ACM Classification Keywords
Categories: I.6 SIMULATION AND MODELING

I.6.5: Model Development

I.6.4: Model Validation and Analysis

Subject descriptor: *Data Analysis*

Introduction
Fortnite is a free-to-play battle royale video game developed by Epic Games. It takes place on an island and features up to 100 players who either play alone or in small squads. The goal of the game is to be the last player alive by either killing other players or temporarily hiding from them; while staying within a constantly shrinking safe zone that forces players closer together. Additionally, players must scavenge for weapons and armor to gain the upper hand on their opponents. The game adds a construction element, where players can break down most objects in the game world to gain resources and use the resources to build fortifications as part of their strategy.

Our Project
Amid the vast increase in gameplay dedicated to Epic's Game: Fortnite, we decided to develop a model that helps determine the rank you would finish with based on your kills, i.e, competitors killed. The data was collected through an anonymous google form, which had been placed on forums accessible to Fortnite players. The form gathered information regarding the overall rank i.e. position finished, initial landing location, number of kills, and the mode the game was played i.e. Solo, Squad, or Duo. While collecting data, the assumption was made that players who filled out the anonymous google form were amateur players with an above average skill in the game due to their enrollment to fan pages. After undertaking data wrangling and data cleaning techniques such as removing data relating to inaccessible quadrants on the graph, impossible situations, and

	Timestamp	Location	Mode	Position	KillCount
0	2018/03/05 7:16:09 PM EST	D5	Solo	6	8
1	2018/03/05 7:43:32 PM EST	G2	Solo	1	3
2	2018/03/05 7:44:48 PM EST	F8	Solo	6	3
3	2018/03/05 8:00:04 PM EST	D10	Solo	60	1
4	2018/03/05 8:00:05 PM EST	G8	Solo	1	8

Table 1: *The table shows the head of the dataset before cleaning.*

	Location	KillCount	Position	Mode
0	D5	8	6	0
1	G2	3	1	0
2	F8	3	6	0
3	D10	1	60	0
4	G8	8	1	0
5	B2	5	1	0

Table 2: *The table shows the head of the dataset after the data was cleaned.*

converting nominal data to numeric values. We analyzed the relationship between the different labels and found that the kill count and rank had an exponential decay relationship.

Background on the Data

As this game is still new, much data is not available to the public for analysis. Therefore, we had to create our own dataset; we did this by sending out a four-question survey and shared it on multiple social media platforms. The survey consisted of the following questions "Which quadrant of the map did you land in" (As the map in the game is split into quadrants with coordinates), "Which game mode were you playing", "What place did you finish", and "How many kills did you get". This survey was created for it to be easy to fill out while collecting the required data for analysis and each question on the form was mandatory to prevent missing data. The survey was posted to the Fortnite subreddit on Reddit and the official Fortnite Forum; this was done with the assumption that posting the survey to these two forums will allow us to collect credible information from players who have the knowledge and skill to perform well on the game.

Data Credibility

The biggest shortcoming of collecting anonymous data from social platforms is the data's credibility, the users might add incorrect data or skewed data to rig our results. However, as most of our group plays the game we used our domain knowledge to analyze results for their credibility. For instance, two data points entered claimed to get the last place with multiple kills. This is impossible because if someone gets the last place then nobody else had died before them. To combat this issue, we manually cleaned the data and deleted data points that we knew were either a result of human error or individuals trying to skew our data. However, cleaning the data was not enough because people could enter data points that are plausible but are not very likely to occur. We decided the best idea was to have two data sets. The first one consisted only of data that we entered which we knew was correct and second data set consisted of responses from the internet. The reason was that we could analyze each data set separately and if they had similar descriptive statistics then we could confirm that the data entered was good data.

Data Cleaning

Based on the method of data collection, the "Locations" data was entered as nominal values. To perform analysis on this data, we converted it to numerical data. This was done by our algorithm that gave each quadrant name a coordinate based on its location on the Fortnite map. Fortnite's map is set up in such a way that it has its quadrants labeled A through J along the horizontal axis and 1 through 10 along the vertical axis to create values such as "D5" or "H7". The first part of the algorithm separated the "letter" part of the coordinate and saved it in a separate list and the same process was done for the "number" part of the coordinate. After doing so, each list was run through a set of 'if' statements that translated the letter with its corresponding numerical value (ex: D would be matched with 5) and each number would be matched with its equivalent numerical value. Each of the correct, numerical values were stored in their respective arrays for x and y to be used later in the confusion matrix and the heat map. To further clean the values within the set, we removed values with the help of the confusion matrix generated. Through our experience playing the game, we knew there are certain areas of the playable map that are unreachable for the player. As there are not many unreachable quadrants, we were able to implement code that would simply remove the points players entered with unreachable areas from the dataset. After cleaning all of the data, we were left with a count of 974 location points within the reachable areas on the map.

```
[[  0,   6,   0,   4,   0,   0,   0,   0,   0,   0],
 [  8,  24,  12,  18,   2,  26,  12,   6,   8,   6],
 [  8,  14,  43,   4,   4,  14,  10,   8,  18,   0],
 [  6,   2,   4,   4,  20,   6,   8,  12,   6,   2],
 [ 35,   0,  14, 124,   0,  14,   2,   2,   2,  10],
 [ 18,   2,   2,  20,   4,   6,  18,  56,   2,   6],
 [ 10,   4,  20,  38,   2,  58,   6,   4,   4,   2],
 [  0,   0,   0,   8,   6,  14,  20,  16,  26,   0],
 [  0,   0,   0,  14,  16,   2,   4,   0,   6,   0],
 [  0,   0,   0,   0,   0,  28,   6,   0,   0,   0]]
```

Figure 1: The confusion matrix created from the data where players dropped

	KillCount
count	5.000000
mean	6.900000
std	3.435113
min	3.000000
25%	5.000000
50%	5.500000
75%	10.000000
max	11.000000

Table 3: *The table shows the descriptive statistics of all values in the working set where position equals 1.*

	KillCount
count	19.000000
mean	6.260526
std	3.605437
min	0.000000
25%	5.000000
50%	5.333333
75%	8.000000
max	14.200000

Table 4 *The table shows the descriptive statistics of all values in the testing set where position equals 1.*

Descriptive Statistics

From the data, we collected we discovered that there exists a relationship between two of the values. These values were rank (or position placed) and the number of kills. There existed a very large variance between the data points where the position is equal to one. Some people were able to win with 0 kills and some won with an unrealistically large kill count. To look deeper into why this might be, we looked at the descriptive statistics. From these values we discovered that the mean rank was 8, the median rank was 2, the mode rank was 1, the standard deviation of the rank was 15.105, the maximum rank was 99, and the minimum rank was 1. Through these statistics, we noticed a discrepancy in our data, the data submitted through the survey largely excluded data of losing matches. Although the minimum rank and maximum rank were as expected, the mean rank was far too superior. Even though we expected skilled players to be submitting the data, a mean rank of 8 was far better than expected.

Additionally, we discovered the median kill count was 4, the mean kill count was 5.34, the mode kill count was 3, the standard deviation of the kill count was 9.054271, the maximum kill count was 99, and the minimum kill count was 0. The kills also showed an error the collected data. When looking at the results of the kills per win compared to those from the working set of data, the results are more similar than anticipated. For the working set, the average amount of kills received per win was 6.9 and the standard deviation of kills was 3.4 while the standard deviation for the testing set was 3.6 and the average amount of kills was 6.2. The fact that the two crucial statistics are so similar confirms that the online dataset was valid even though some outliers exist.

Figure 2: Representation of a heatmap of the of the count of players dropping to specific locations. Whiter the quadrant the more people landed in it.

Model

After collecting and cleaning 974 data points we began to visualize our data. We found there existed an exponential decay relationship between the number of kills and the rank finished. A log transformation did not make the data linear, therefore, it was not viable to fit a linear regression on the data. Additionally, as the data contained more possibilities than the required two for logistic regression, we did not receive good values with that models. Therefore, we decided to create an exponential decay model.

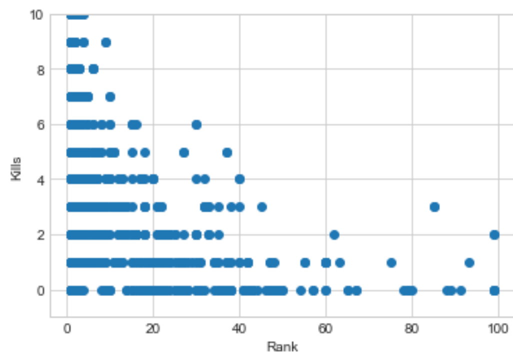

Figure 3: Scatter plot of Rank achieved at the end of the game and number of kills received

To fit the shown data, we used scipy's curve fitting library and then used the exponential decay equation

$$Y \;=\; Ae^{-bx}{+}\mathrm{c}$$

Here, y is the rank finished, A is the initial value, b is the rate of decay and c is the horizontal asymptote. We received the value for A as 29.580, b as 0.649, and c as 2.813.

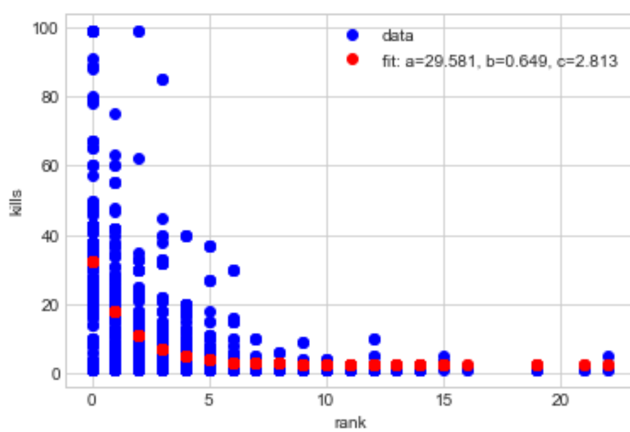

Figure 4: This graph shows our model in red versus the data set

Results

Using the model, we calculated our $\hat{y}$. We received an array of predictions and found our maximum predicted rank as 33 and our average rank as 10. Although our model's average rank prediction was very close to that of the actual data. Its maximum prediction was extremely low. To validate our findings, we calculated the relative residuals which is

$$\frac{(\hat{y} - \widehat{y})}{y}$$

As you can see in figure 5, There is a wide range in our residuals. This show us that our model is not performing as well as we would like. We assume this is happening due to our model underfitting.

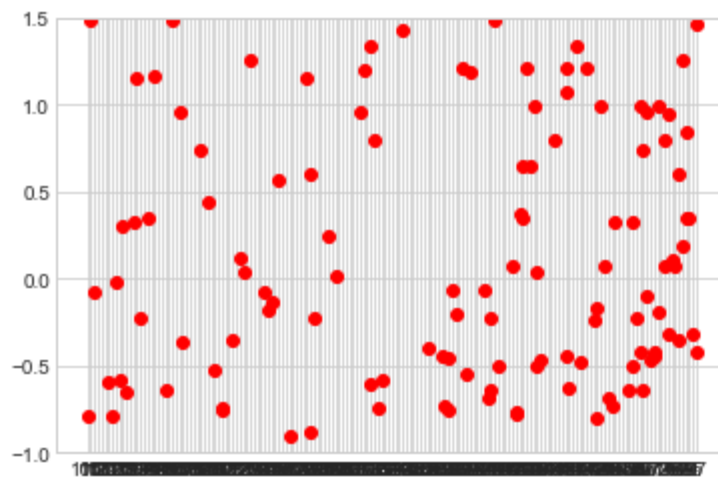

Figure 5: Plot of the relative residuals

Discussion

During the data collection process, we faced many challenges. Due to the data collection being conducted by a survey on Fortnite Forums and Reddit forums we ran into issues initially collecting data. Within the data we received there existed some very unrealistic game data. There also was a great deal of "inflated" datapoint entries. This was obvious as 35% of the data points entered were wins. These abnormalities affected our model. We had assumed to amass data that created a bell curve whereas we received an exponential decay graph, positing most of our data points as wins, i.e., position 1 with a large variation. This caused another challenge as the model was underfitting, most of the data points were restricted to a smaller range than our dataset indicated.

The challenges we encountered reflected on the complexity caused by collecting data from anonymous sources. It intensified the need to be more aware of the domain of the data being collected and practicing better cleaning techniques. Although we had to visually clean a lot of data as an individual finishing last could not have had 100 kills. We also cleaned our data through the confusion matrix, it helped remove data points that were not viable and were potentially entered to make our dataset messy.

Additionally, the data showed that there was a tendency for people who finished in a higher position to have a higher kill count. The data did not show a strong enough correlation among any other factors. We used this for our model. The data also showed that most of the survey entries were played in the solo game mode, followed by squad mode, and lastly the duo mode. After conducting data wrangling and cleaning we developed an exponential decay model using kill count and position finished. Although our model was not complex we were capable of approximately predicting finishing ranks.

Conclusion

Through our modeling and data collection process we learned that it is harder to model an exponential decay model as compared to a linear regression as lesser techniques are available to do the same. Additionally, we learned that collecting data from the internet, although is a quick process, it is not the most reliable form of data collection. Individuals can purposely skew data or add incorrect values which can only be cleaned through domain knowledge. From our observations, we would recommend collecting data from more reliable sources such as friends. Also, collecting the data straight from the games played through Epics help, although, we were unable to get the gameplay data from Epic Games when we reached out, maybe more persistence would be helpful

Acknowledgments

We thank all the volunteers who contributed data. All the authors of the reference articles and well as the web developers of similar stat websites. These provided helpful insight on what was already available and how it functioned. Special thanks to Professor Dr. Guha and Teaching Assistant Devansh Saxena for their support.

Appendix

Our code is available at https://github.com/Justin1st/Fortnite_Heatmap.

References

1. Ed Thorn. 2018. Fortnite: Battle Royale - Best places to land. Retrieved March 6, 2018 from https://www.metabomb.net/fortnite-battle-royale/gameplay-guides/fortnite-battle-royale-best-places-to-land-for-loot-2
2. Rich Edmonds. 2018. How (and why) Battle Royale games like PUBG took the world by storm. Blog post. Retrieved March 6, 2018 from https://www.windowscentral.com/why-battle-royale-games-have-taken-gaming-world-storm
3. Tae Kim. 2018. 'Fortnite' is becoming biggest game on internet. News article. Retrieved March 6, 2018 from https://www.cnbc.com/2018/03/15/fortnite-is-becoming-biggest-game-on-internet.html
4. Fortnite: Battle Royale Chest Location Coordinates. Kaggle dataset. Retrieved March 6, 2018 from https://www.kaggle.com/jruots/fortnite-battle-royale-chest-location-coordinates

5. Wikipedia. Fortnite Battle Royale. Retrieved March 6, 2018 from https://en.wikipedia.org/wiki/Fortnite_Battle_Royale
6. Scipy Documentation, optimize curve fitting, Retrieved April 25, 2018, https://docs.scipy.org/doc/scipy/reference/generated/scipy.optimize.curve_fit.html
7. Wes McKinney. *Data Structures for Statistical Computing in Python*, Proceedings of the 9th Python in Science Conference, 51-56 (2010)
8. Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, Édouard Duchesnay. Scikit-learn: *Machine Learning in Python, Journal of Machine Learning Research*, 12, 2825-2830 (2011)
9. John D. Hunter. *Matplotlib: A 2D Graphics Environment, Computing in Science & Engineering*, 9, 90-95 (2007),DOI:10.1109/MCSE.2007.55 (publisher link)
10. Travis E, Oliphant. *A guide to NumPy*, USA: Trelgol Publishing, (2006)