Game of Chance: Exploring the Expected Duration of a Die Rolling Contest

Jackson Schieber – Team 20

Abstract

This team was tasked with calculating the expected number of cycles that a die rolling game would take to end and determining the most representative distribution of the data. This game is played between two players who alternate turns. Each player begins with four chips/coins, and two chips start in the pot. On each turn, a player rolls a die. If the die rolled is a one, nothing happens. If it's a two, the player takes the entire pot. If it's a three, half of the chips rounded down are owed to the player. Finally, the player owes 1 chip to the pot any time he rolls a four, five, or six. At any point where a player owes money to the pot but cannot pay, the game is over. Each time where both players take one turn is considered a cycle. The team used "first step analysis" via a Markov chain to statistically predict the number of cycles. The prediction was between 13 and 14. Next, the team simulated the game by modeling the rules in Python using the randint(1,6) function and recorded the number of cycles per game. The games averaged 17.5 cycles, had a median of 14, and followed a Chi distribution.

Background

A quick google search will reveal that measuring the probability of occurrence of an event in any kind of game is a popular and heavily considered topic and statistical challenge. While predicting how long a die rolling game will last may seem trivial, it's notably consequential as it serves as proof of concept of other types of predictions about an uncertain future. If one can predict a die rolling game maybe one can apply similar concepts/principles to predict the economy, natural disasters, disease propagation, industry, etc.. One common technique of measuring the probability of an event occurrence includes first step analysis through Markov chains. The team referenced Dan Ma's article: "Topics in Probability." He explains that the purpose of the method is to "break down the possibilities resulting from the first step (first transition) in the Markov chain. Then use the law of total probability and Markov property to derive a set of relationship among the unknown variables." Next, he proceeded to offer some examples of the technique which informed the team's own analysis. Therefore, the team was able to predict the occurrence of a sequence/chain of die rolls and chip adjustments.

Markov

In order to predict the length of the game, it's important to know the exact probability of every event. There is a 1/6 chance of nothing happening on a player's turn. There is a 1/6 chance of obtaining the entire pot, and there is a 1/6 chance of obtaining half of the pot rounded down. Finally, there is a 3/6 chance of a player owing one chip to the pot. One of these events will happen every time. Knowledge of these probabilities enables one to predict the probability of arriving at any point in the rest of the game; therefore, one can predict when a player will be indebted to the pot without funds to pay (game over) and the number of cycles this will occur at. Because this game is unique, there were not previous efforts by others to solve it for the team to reference. One challenge for the team was representing each part of the game as a state in order to take advantage of Markov chain state change analysis. The team accomplished this by creating a tuple of length four, including the first player's chip count, the second player's chip count, the pot's chip count, and the turn number (1 or 2) of the player. The team created a vector of every possible combination of states – 132 in total. Then one additional state was added to represent a game ending condition. Next, the Markov probability matrix was created using an algorithm that assigned a given probability of transitioning from each state to another. The code used two "for loops" to cycle through each beginning state and each ending state. If the beginning to ending state matched the correct criteria it would be assigned the appropriate probability; otherwise, it was assigned a zero. For example, if the beginning state were (4,4,2,1) and the ending state were (8,8,3,1), that particular index of the probability matrix would be zero since there is no scenario in which that transition can happen according to the rules.

The first check that is run against each beginning-ending state pair assigns a 1 to the matrix if both states say “End Game” as there can be zero state changes after the game ends. The next check assigns 1/6 to the matrix if the states are identical in each way but turn number – if a one is rolled do nothing (change turns). The next set of checks are further indented into a new “if statement” measuring which player is taking their turn and are slightly altered depending on who it is. The first check represents a die roll of two. If the ending state pot = 0, the ending state’s player who is taking the turn has a chip count = beginning player + beginning pot, the inactive player’s chip count is unchanged, and the ending state’s turn number differs from the first (always must change turns), there will be a 1/6 assigned. In order to check for a die roll of three, a similar strategy is used except the active player’s chips = chips + pot // 2 and the pot = pot – pot // 2 since only half of the pot rounded down is taken. Finally, to measure the probability of rolling a 4/5/6, the algorithm checks for an origin-destination state pair where the turn number is different, and the active player has one fewer chips in the ending state than he did in the first. All else must be identical. If so, a 3/6 is assigned to the matrix since there is a 50% chance of rolling one of these numbers.

Having Created the matrix, the team verified that the sum of each row equals 1. This is a necessary requirement for a Markov chain to accurately reflect the probability of an event. A number greater than one would indicate that one side of a six-sided die is rolled more than 1/6 of the time. This is clearly inaccurate. Similarly, anything less than one would indicate that a side is rolled less than it should be. After verifying the numbers, the team created a vector representing the initial state. Every row is 0 except for the index corresponding to (4,4,2,1) which is 1 since there is 100% probability of starting in this state. Next a “for loop” is utilized to multiply the starting vector by the probability matrix several different numbers of times which symbolize the number of turns taken. The last row of the vector (created after each multiplication) represents the end game condition probability. Therefore, this number is recorded after each set of turns (the number of turns is divided by two and rounded up to derive cycles). After 13 cycles, there is a 49.4 percent chance of the game ending, and after 14 cycles there is a 51.3 percent chance of game ending. Thus, using Markov techniques, one can conclude that the game should last between 13 and 14 cycles.

Simulation

Having produced a prediction using math/probability, the team next attempted to predict cycles through simulation of the game. A second algorithm, similar to the first, was produced to “simulate” a playthrough of the game. Randint(1,6) is used to represent the die roll, and there are variables to symbolize each player’s and the pot’s chips. Another variable tracks the total number of turns, and the cycles are derived by dividing the turns by two and rounding up. Again, the game ends when one player is obligated but unable to deposit a chip. A “for loop” is used to run through 100,000 simulations of the game and the duration of each game is appended to a list. The average number of cycles is about 17.5 which is in the same realm but notably higher than when using the Markov chain.

At first, the team was unsure of why such a discrepancy exists. It speculated that it was possible that the numbers would converge slightly with additional trials. It also wondered if the randint() function does not offer a truly random distribution of numbers causing some of the disparity. Eventually, the team realized that the average is not the best metric for predicting when the game will end. The team then measured the different quantiles of the data. They were the following: 8, 14, and 23. Therefore, 50% of the games are over within 14 cycles just like when calculated using Markov chain analysis. Moreover, the mode is 6. Therefore, there are multiple different statistical measures of cycle length, but the two methods – simulation and Markov – do seem to agree with each other.

Having gathered a set of samples, the team plotted a histogram. Cycles under four are nonexistent since it is impossible to lose in this amount of time. Five is the second most common game length, six is the most common as stated above. After five, the number of games ending in x+1 cycles decreases in an exponential looking pattern. Inspect Figure 1 and Figure 2 below for further details.

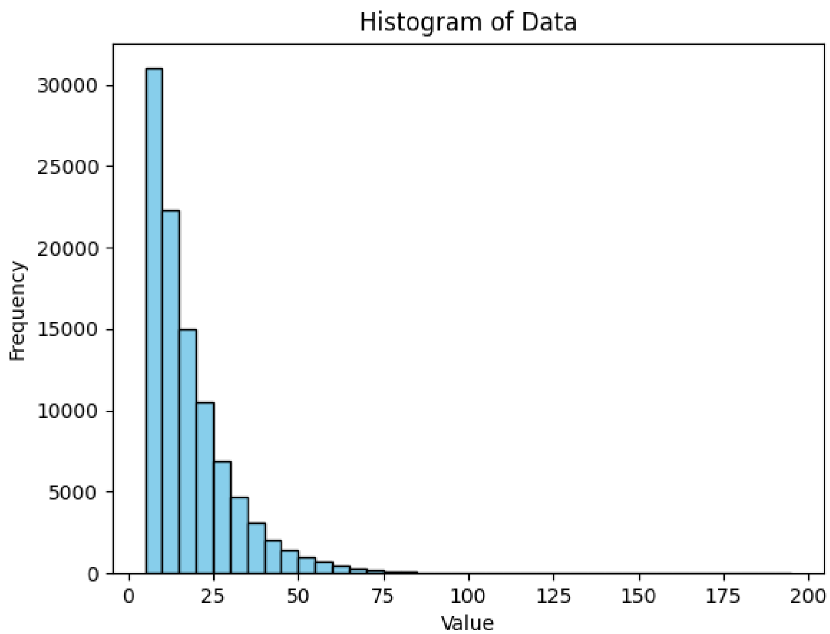

Figure 1: Histogram of cycles with bin size of five

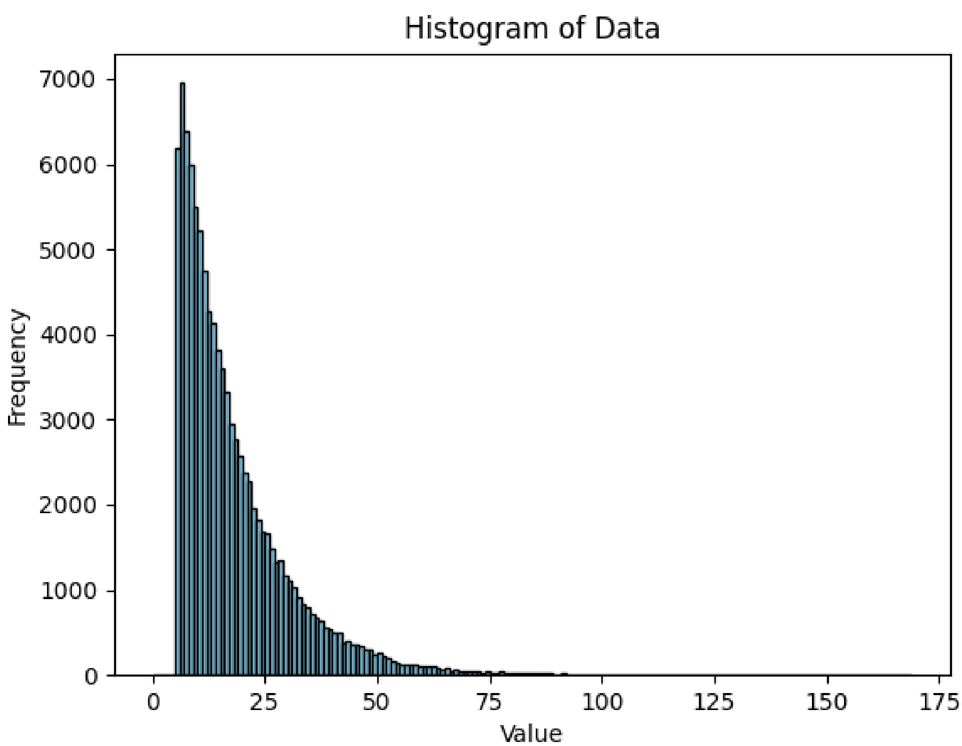

Figure 2: Histogram of cycles with bin size of one

In order to identify the distribution that best fit the data, the team utilized the fitter library. Additionally, the team attempted 50 thousand games instead of 100 thousand to reduce the load on the library. When using 100, it timed out on a large number of possible distributions. However, when using 50 thousand, it identified chi, exponentiated weibull, generalized gamma, exponential power, and generalized inverse gaussian, as the top five best fits in descending order. Chi had a squared error of just 0.002397. This explains why there was a smaller disparity between the 25^{th} and 50^{th} percentile than the 50^{th} and 75th percentile and why the average was so high. It is biased by the large numbers of cycles that the game has the potential to be played for.

Conclusion

The thesis statement or purpose of this project is to find the “expected number of cycles” that the game will last for. This isn’t as clear and concise an objective as one might think as this variable can be measured in different ways. One could make an argument that 6 cycles is the best answer since it is the most popular finish in the simulation. However, 6 still represents just a small percentage of the overall number of finishes. The 50^{th} percentile is approximately 14; therefore, it’s not irrational to argue that 14 is the best guess since it falls in the very middle of every result. Furthermore, this number is mathematically supported by Markov chain analysis. Ultimately, this analysis serves as a lesson that it is important to understand the intent behind the objective and the risk/reward involved. If a gambler were promised a reward for guessing the exact number of cycles but had only one guess, the most logical answer would be 6 since it is the most common. However, if the objective were to simply get closer than another gambler to the number of cycles, 6 would not be the smartest decision. 14 would be wiser since it sits in the middle of every possible result. Ultimately, this project serves as a good starting point and learning experience for further, more complicated or practical analysis on probability and game theory (poker for example), and there is no short supply of these things to analyze. Afterall, the better one can predict the future the greater he can optimize for it.

Works Cited

Ma, Dan. "First Step Analysis and Fundamental Matrix." *Topics in Probability*, 8 Mar. 2018, probabilitytopics.wordpress.com/2017/12/15/first-step-analysis-and-fundamental-matrix/.