Exploring Snakes and Ladders

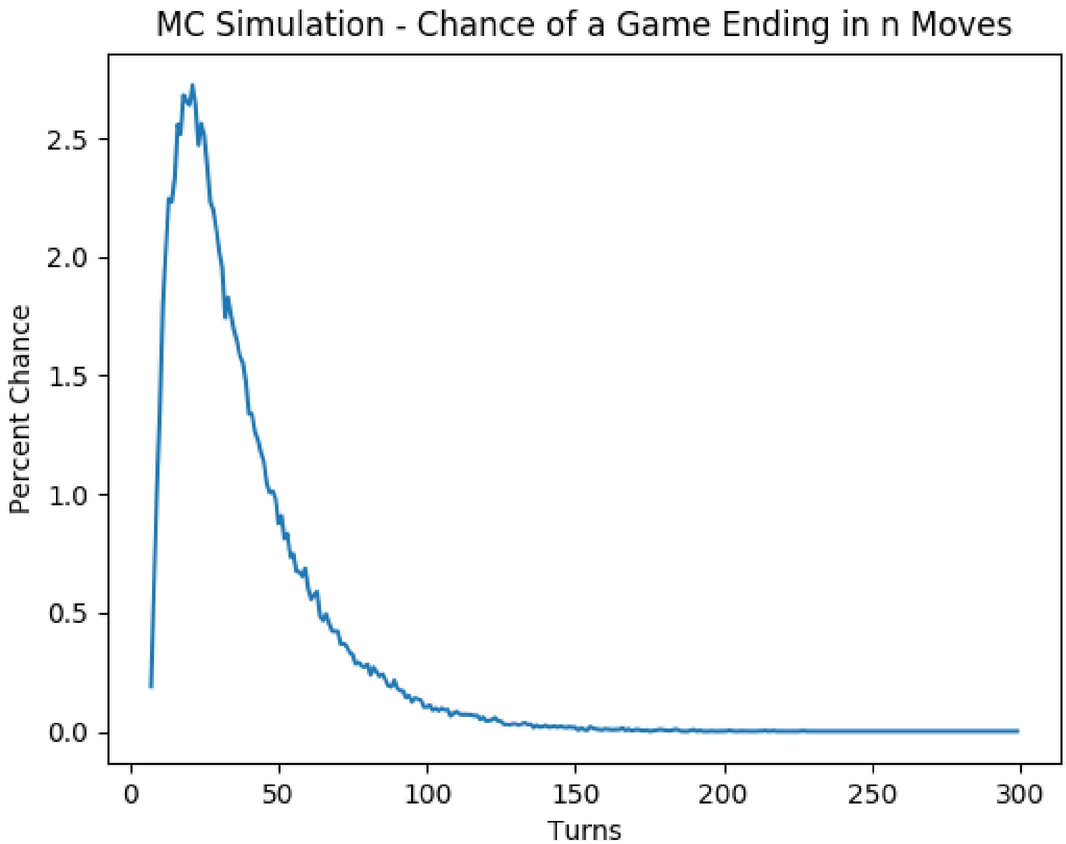
This document investigates the probabilities involved in a one player game of snakes and ladders using Monte Carlo simulations and Markov Chains. The board we will be working with is of size 100×100. We will allow the player to reach square 100 without an exact roll (i.e. a roll of 6 on 99 will end the game). The positions of the snakes and ladders are show in a table below.

Ladders	
1	38
4	14
9	31
21	42
28	41
36	44
51	67
71	91
80	100

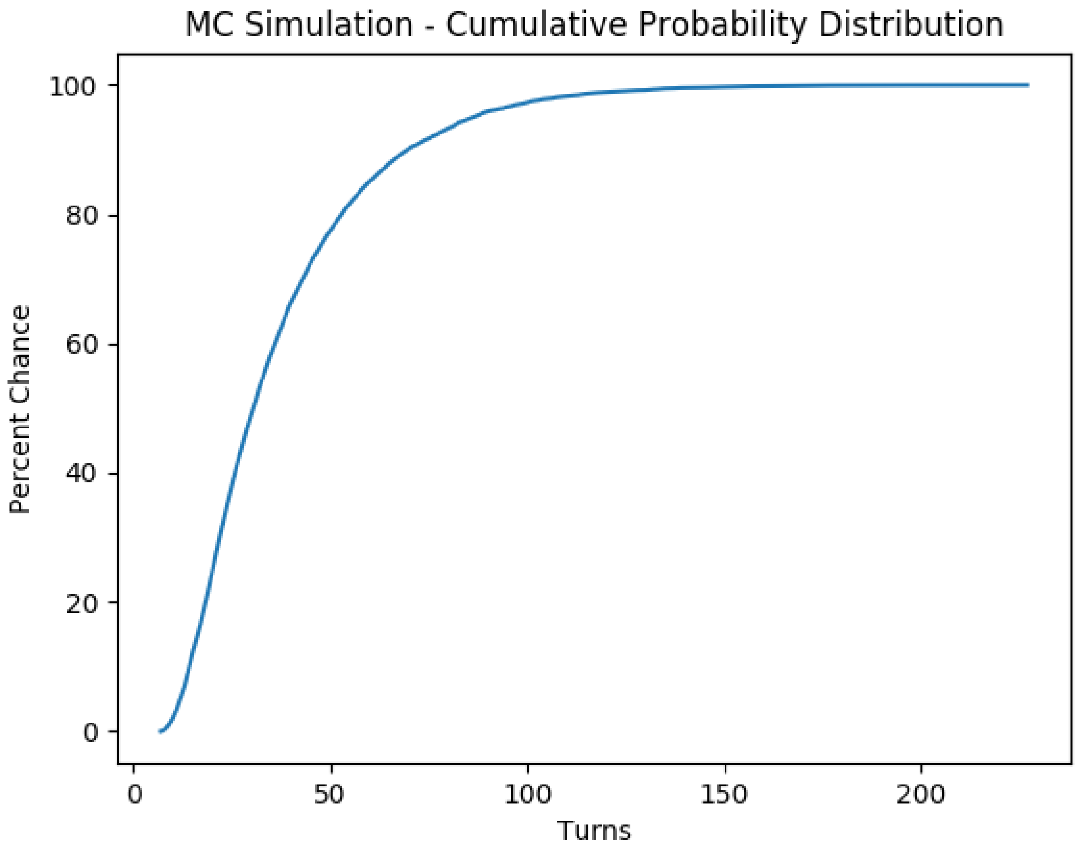
Snakes	
98	78
95	75
93	73
87	24
64	60
62	19
56	53
49	11
48	26
16	6

All players start off of the board, referred to as square 0. They then use a six sided die to move forward in an attempt to reach square 100. Theoretically a game could last forever because of the snakes and ladders. However, the probability of this is asymptotically low.

A simulation of the game was created in Python and iterated for a million runs. The graph on the next page shows the results from the simulations. Some other values were also recorded during the simulations and are presented in a table. Surprisingly the graph and values stayed almost exactly the same when simulations were ran without the snakes and ladders.



The graph illustrates the percent chance of a game ending in n moves. This allows us to get a rough understanding of the probability distribution. We can see from the graph that the earlier claim that the probability of arbitrarily long games approaches zero seems to be verified by the simulation. We found that the longest game was 337 turns, while the shortest was only 7. The average game lasted 37 turns. Games most frqeuently ended in 19 turns, which forms the peak on the graph. To determine the median we also generated a cumulative distribution graph, which is on the next page.



The cumulative probability distribution shows what percent games ended in n turns. Games that lasted longer than 150 turns were in the 1% for example. We extract the median number of turns a game took to finish by looking at how long half of our games lasted. Hence the median obtained from our simulations was 29.

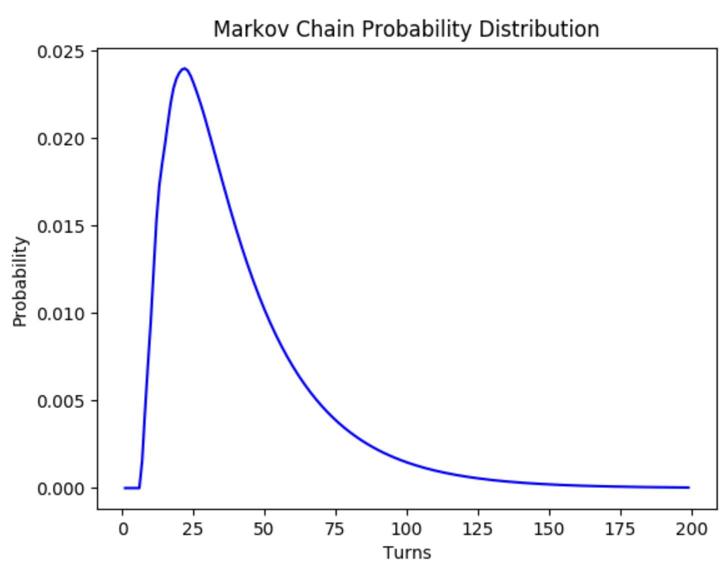
By running a simulation we were able to get a rough approximation. The more iterations we were to run, the more accurate our results would be. Unforuntately, we cannot run an infinite number of simulations so we must resort to an alternate avenue to obtain exact results.

The game of snakes and ladders is perfect for being modelled as a Markov Chain. At each point in the game, the probability of moving to another state is independent of previous states and only depends on which state the player is currently in.

To represent the probabilites of moving from one state to another we can create a 101×101 matrix, M, which is referred to as a stochastic or transition matrix (the size is 101 because we include square zero). $M_{i,j}$ is the probability of moving from square i to square j. The last row would be zero-filled as square 100 is an absorbing state. We used a Python script to generate this transition matrix.

For representing the current distribution of the game we use a state vector, $\vec{v}$ of length 101. The state vector at the start of the has a one in its first row, and the rest of the rows are zero-filled. This is because there's probability of 1 that the player is at square zero at the start of the game.

To see the probability of a player being in a certain square after n moves we compute $\vec{v}M^n$. The value in row i corresponds to the probability of being in square $i-1$. So to see the probability of a game ending in n turns we can simply use the value of the last row of $\vec{v}M^n$. This allows us to create a probability distribution which can be seen below.



The distribution we obtained from using Markov Chains is quite similar to the earlier distribution produced from our Monte Carlo simulation. This seems to be verification of both results.

But now the question arises, what is the expected number of turns a game should take to complete? It seems that the Markov Chain approach, like our simulation, has just spit out a graph and not much else. However, we can easily produce the expected number of games using some elementary manipulations.

Let us define the fundamental matrix as $N = I + M + M^2 + \cdots + M^n$, where M is the stochastic matrix for an absoring Markov process.

Then $MN = M + M^2 + \cdots + M^{n+1}$

And so $N - MN = I - M^{n+1}$

But then $(I - M)N = I - M^{n+1}$ then $N = (I - M)^{-1}(I - M^{n+1})$

We get $\lim_{n\to\infty} N = \lim_{n\to\infty}(I - M)^{-1}(I - M^{n+1}) = (I - M)^{-1}$

Note that this limit only exists if $(I - M)$ is non-singular.

Also, $(I - M^{n+1})$ converges to I as M is the transition matrix for an absoring Markov process.

Let $\vec{u}$ be a vector of length 100 whose elements are all 1, and let P be M with the last column and row removed. Then the expected distribution would be $N\vec{u} = (I - P)^{-1}\vec{u}$ where each row represents the expected number of moves until finishing the game from the corresponding square. We see that the expected value of finishing a game from square one, which is the case we were concerned with, is 39.

That concludes our analysis of snakes and ladders.