Mini Project 1 - Markov Chains

Richard Hall

Oct 24, 2019

This is the first project for the class ECE586 - Vector Space Methods taken at Duke University during the fall of 2019. In this project, we explore the application of linear algebra techniques and statistics to analysis simple Markov Chains. There are two primary parts to the project. In the first part, we are exploring the effects of absorbing states. In the second part, we look at a chain's steady state probability distribution.

A Markov chain can be modeled as an $n \times n$ matrix P where n is the number of states. The element $P_{i,j}$ represents the probability of being in state i and moving to state j. Given a starting state, a chain of random states will be generated with transitions occurring based on the probabilities in P.

In our first example, we look at flipping a coin to arrive at a state where three heads have been flipped in a row. Note that, once this condition is met, every following flip also satisfies the condition, regardless of the outcome of the flip. This means that, when we enter this state, we will never leave and the state is called absorbing.

We can model the probability matrix with 4 states. The first state represents the previous flip not being heads and 3 heads never having being flipped in a row previously. The second state represents the previous flip being heads, the flip before that not being heads, and the 3 heads never having being flipped in a row previously. The third state is similar to the previous two and the forth and final state indicates that 3 heads have been flipped in a row at some point in the past. The complete matrix is

$$P = \begin{bmatrix} 0.5 & 0.5 & 0 & 0 \\ 0.5 & 0 & 0.5 & 0 \\ 0.5 & 0 & 0 & 0.5 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Observe how the forth state is, in fact, absorbing since it will always remain in state 4.

Assuming we begin in state 1 (end condition not met and previous flip was not heads), lets say we start flipping the coin. How many times would we need to flip the coin to expected we are in state 4? Another way to ask this is what is the expected hitting time from state 1 to state 4, $\mathbb{E}[T_{1,4}]$. To calculate this, we first compute the probability that the first time we are in state 4 is at time m,

$$\Pr(T_{1,4} = m) = \Pr(T_{1,4} \le m) - \Pr(T_{1,4} \le m-1).$$

We then take a weighted sum of all the probabilities with and their respective values of m. The result of this sum is the expected hitting time from state 1 to 4. A short Python script was written to calculate this. A matrix Φ is computed recursively such that $\phi_{i,j}^{(m)} = \Pr(T_{i,j} \le m)$. This is done by the recursive algorithm,

$$\Phi^{(0)} = I$$

$$\Phi^{(m+1)} = I + (1 \ominus I) \otimes \left(P\Phi^{(m)} \right)$$

With this recursion, we can create a new matrix defined as

$$E = \sum_{i=1}^{\infty} i(\Phi^{i-1} - \Phi^{i}).$$

Each index of E represents the average hitting time from state i to state j. We estimate the value of E but summing the first 100 terms in the summation. This produces a average hitting time of

$$\boxed{\mathbb{E}[T_{i,j}] = 13.97056826}.$$

Using the same Python script, we can simulate the 500 instances of the Markov chain starting in state 1 and running till we are in state 4. We generate a random number between 0 and 1. If it is less than 0.5, the 'flip' is tails

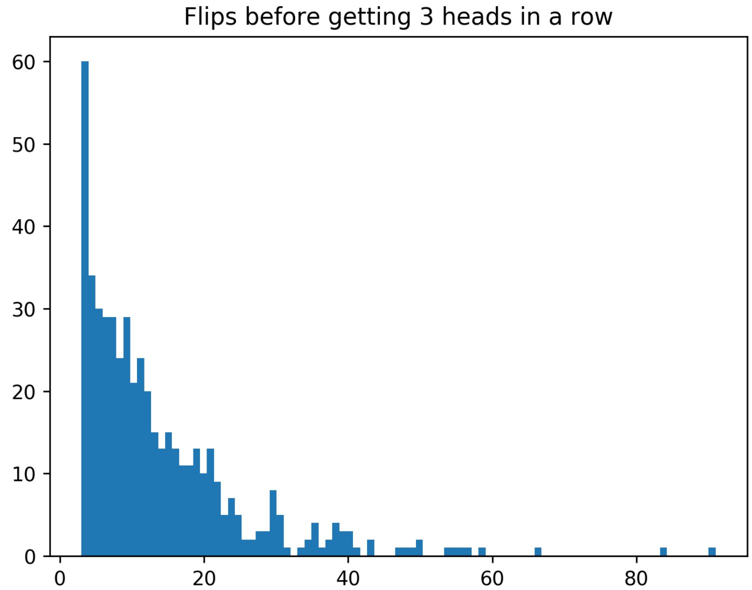

Figure 1: Hitting Times

Figure 2: Simple Snakes and Ladders game

and we move to state 1. Otherwise, we increase the state by one and terminate if we move to state 4. The hitting times of the 500 runs are plotted in Figure 1.

As a more complex example, we consider the miniature chutes and ladders game shown in Figure 2. Assume a player starts on the space labeled 1 and plays by rolling a fair four-sided die and then moves that number of spaces.To finish the game, players must land exactly on space 20. The game can be modeled with $n = 16$ states and the probability matrix

$$P = \begin{bmatrix}
0 & {}^1/_4 & {}^1/_4 & {}^1/_4 & 0 & 0 & {}^1/_4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & {}^1/_4 & {}^1/_4 & {}^1/_4 & 0 & {}^1/_4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & {}^1/_4 & {}^1/_4 & {}^1/_4 & {}^1/_4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & {}^1/_4 & {}^1/_4 & {}^1/_4 & {}^1/_4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & {}^1/_4 & {}^1/_4 & {}^1/_4 & {}^1/_4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & {}^1/_4 & {}^1/_4 & {}^1/_4 & {}^1/_4 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & {}^1/_4 & {}^1/_4 & {}^1/_4 & {}^1/_4 & 0 & 0 & 0 & 0 & 0 \\
0 & {}^1/_4 & 0 & 0 & 0 & 0 & 0 & 0 & {}^1/_4 & {}^1/_4 & {}^1/_4 & 0 & 0 & 0 & {}^1/_4 & 0 \\
0 & {}^1/_4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & {}^1/_4 & {}^1/_4 & 0 & 0 & 0 & {}^1/_4 & 0 \\
0 & {}^1/_4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & {}^1/_4 & {}^1/_4 & 0 & 0 & {}^1/_4 & 0 \\
0 & {}^1/_4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & {}^1/_4 & {}^1/_4 & 0 & {}^1/_4 & 0 \\
0 & 0 & 0 & 0 & 0 & {}^1/_4 & 0 & 0 & 0 & 0 & 0 & 0 & {}^1/_4 & {}^1/_4 & {}^1/_4 & 0 \\
0 & 0 & 0 & 0 & 0 & {}^1/_4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & {}^1/_4 & {}^1/_4 & {}^1/_4 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & {}^1/_2 & {}^1/_4 & {}^1/_4 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & {}^3/_4 & {}^1/_4 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix}$$

For this example, we found the cumulative distribution of the number turns a player takes to finish. This is the probability that, at turn m, the player has reached square 20 which is an absorbing state. Since 16 is an absorbing state, it can easily be calculated by taking increasingly higher powers of P and plotting the value of $P^m_{1,16}$. The results of this calculation of plotted in Figure 3.

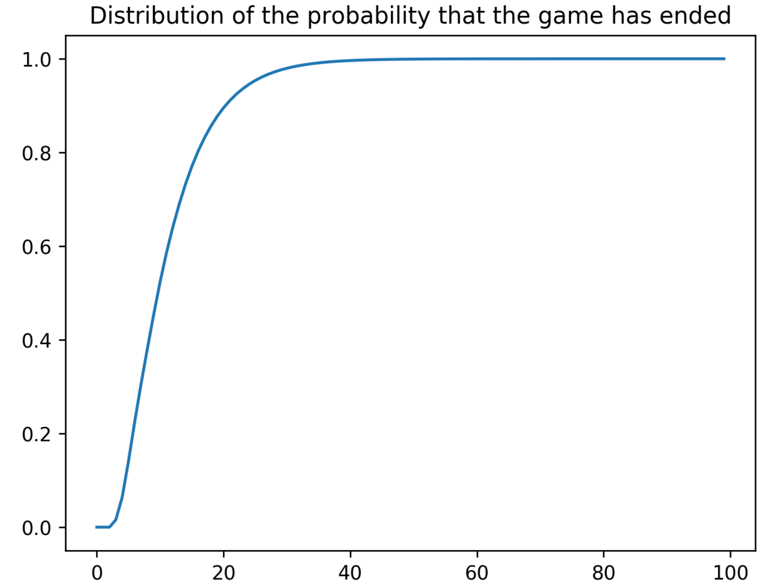

Figure 3: Simple Snakes and Ladders game

For this Markov chain, write a computer program to compute the cumulative distribution of the number turns a player takes to finish. Write a computer program that generates 500 realizations from this Markov chain and uses them to plot a histogram of $T_{1,20}$.

Like in the coin flipping example, we simulated 500 runs of the game. Figure 4 histogram of the resulting hitting times.

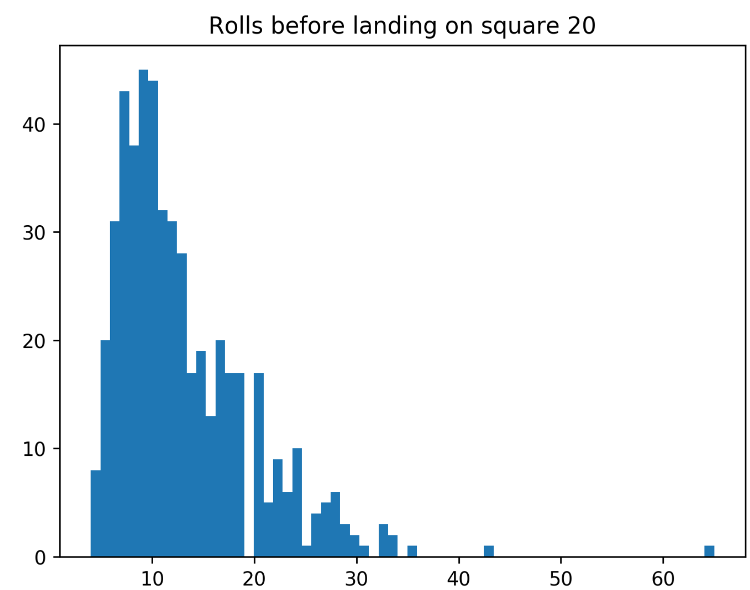

Figure 4: Simple Snakes and Ladders game

Finally, we will consider the case of a board game with 8 spaces on the board. The spaces are arranged in a circle and players move clockwise the number of spaces given by the sum of two four-sided dice. We are interested in the steady state distribution of the spaces on the board. This is the destitution $\underline{\pi}$ such that $\underline{\pi}P = \underline{\pi}$. With basic linear algebra, we can show that the stead state distribution is in the nullspace of $(P - I)^T$ and has elements which sum to 1.

$$(P-I)^T \underline{\pi}^T = \underline{0}.$$

The matrix P can be constructed by first examining the possible values the dice can sum to. There are 16 total possible combinations spanning for $1+1=2$ up to $4+4=8$. Notice that there are four combinations that create a value of 5. As we increase up to 8 and decrease down to 2, the number of combinations decrease by one at each step till there is one combination for 8 and one for 2. From this examination, we see that the odds of the dice summing to a given number is given in the table below.

2	$^1/_{16}$
3	$^1/_8$
4	$^3/_{16}$
5	$^1/_4$
6	$^3/_{16}$
7	$^1/_8$
8	$^1/_{16}$

Table 1: Probabilities of the outcomes

From these, we can construct the probability matrix.

$$P = \begin{bmatrix} ^1/_{16} & 0 & ^1/_{16} & ^1/_8 & ^3/_{16} & ^1/_4 & ^3/_{16} & ^1/_8 \\ ^1/_8 & ^1/_{16} & 0 & ^1/_{16} & ^1/_8 & ^3/_{16} & ^1/_4 & ^3/_{16} \\ ^3/_{16} & ^1/_8 & ^1/_{16} & 0 & ^1/_{16} & ^1/_8 & ^3/_{16} & ^1/_4 \\ ^1/_4 & ^3/_{16} & ^1/_8 & ^1/_{16} & 0 & ^1/_{16} & ^1/_8 & ^3/_{16} \\ ^3/_{16} & ^1/_4 & ^3/_{16} & ^1/_8 & ^1/_{16} & 0 & ^1/_{16} & ^1/_8 \\ ^1/_8 & ^3/_{16} & ^1/_4 & ^3/_{16} & ^1/_8 & ^1/_{16} & 0 & ^1/_{16} \\ ^1/_{16} & ^1/_8 & ^3/_{16} & ^1/_4 & ^3/_{16} & ^1/_8 & ^1/_{16} & 0 \\ 0 & ^1/_{16} & ^1/_8 & ^3/_{16} & ^1/_4 & ^3/_{16} & ^1/_8 & ^1/_{16} \end{bmatrix}$$

Then, using the Scipy package, we solved for the nullspace of $(I-P)^T$ to get

$$\boxed{\underline{\pi} = \begin{bmatrix} ^1/_8 \\ ^1/_8 \\ ^1/_8 \\ ^1/_8 \\ ^1/_8 \\ ^1/_8 \\ ^1/_8 \\ ^1/_8 \end{bmatrix}}$$

Suppose we complicate the game a little further. The only way to escape the first state is if you roll doubles, otherwise, we stay in state 1. If this change is made, the matrix P becomes

$$P = \begin{bmatrix} ^13/_{16} & 0 & ^1/_{16} & 0 & ^1/_{16} & 0 & ^1/_{16} & 0 \\ ^1/_8 & ^1/_{16} & 0 & ^1/_{16} & ^1/_8 & ^3/_{16} & ^1/_4 & ^3/_{16} \\ ^3/_{16} & ^1/_8 & ^1/_{16} & 0 & ^1/_{16} & ^1/_8 & ^3/_{16} & ^1/_4 \\ ^1/_4 & ^3/_{16} & ^1/_8 & ^1/_{16} & 0 & ^1/_{16} & ^1/_8 & ^3/_{16} \\ ^3/_{16} & ^1/_4 & ^3/_{16} & ^1/_8 & ^1/_{16} & 0 & ^1/_{16} & ^1/_8 \\ ^1/_8 & ^3/_{16} & ^1/_4 & ^3/_{16} & ^1/_8 & ^1/_{16} & 0 & ^1/_{16} \\ ^1/_{16} & ^1/_8 & ^3/_{16} & ^1/_4 & ^3/_{16} & ^1/_8 & ^1/_{16} & 0 \\ 0 & ^1/_{16} & ^1/_8 & ^3/_{16} & ^1/_4 & ^3/_{16} & ^1/_8 & ^1/_{16} \end{bmatrix}$$

.

Going through the same process as before but with the new P gives the steady state distribution of

$$\boxed{\underline{\pi} = \begin{bmatrix} 0.418 \\ 0.083 \\ 0.102 \\ 0.071 \\ 0.093 \\ 0.063 \\ 0.096 \\ 0.074 \end{bmatrix}}$$