Markov Chains

Michael Zhang

Updated November 15, 2016

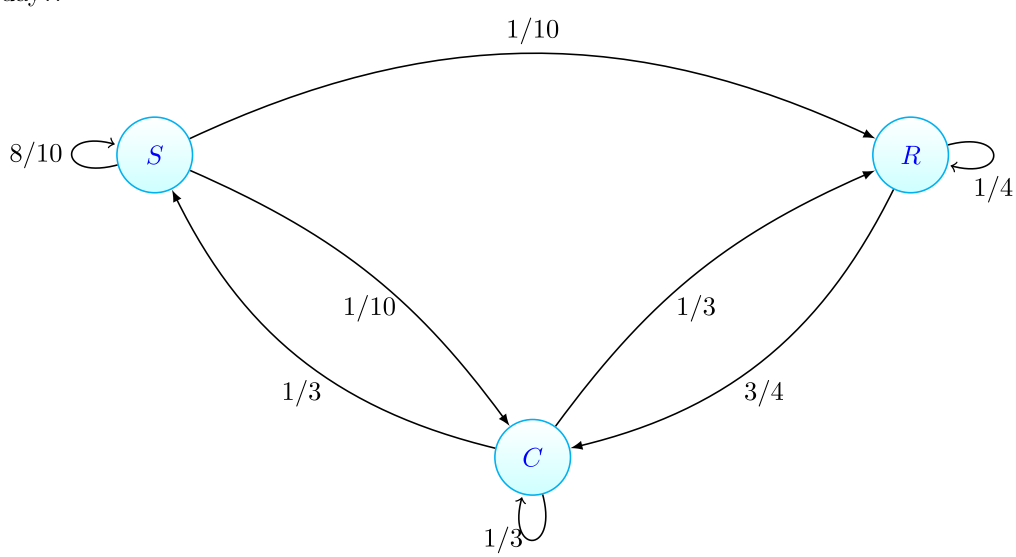
This handout will give an overview of Markov Chains and describe characteristics of Markov Chains we are interested in. We will dive into a problem to try to build intuition on Markov Chains and to illustrate why Markov Chains are useful. I have included conceptual questions that you should stop to think about to help solidify your understanding of Markov Chains.

1 Introduction

1.1 Motivating Example

Problem 1. Farmer George does not believe in weather forecasters and wishes to predict the weather himself. With years of farming experience, he has observed that certain patterns for weather always hold to be true. He classifies weather into three categories: sunny, cloudy, and rainy. To simplify the model, Farmer George assumes that he can perfectly predict the distribution for the weather tomorrow, if he knows the weather today. If it's sunny one day, there is a 80% chance it's sunny, 10% chance it's cloudy, and 10% chance it rains the next day. If it's cloudy, Farmer George thinks there's an equal chance we see sunny, cloudy, or rainy weather the following day. Lastly, if it's raining on one day, there is a 25% chance that rain continues and a 75% chance the weather is cloudy the next day.

a) If it's sunny today, what is the probability that the weather is sunny two days from now? What's the probability that it's cloudy?
b) What is the long term invariant distribution for the weather?
c) Given that it's sunny today, find the expected number of days until the weather is cloudy.
d) Given that it's sunny today, what is the probability we see another sunny day before we see a rainy day?.



1.2 Some Terminology

The graph above models this Markov Chain, where each arrow represents the probability of going from a state to another state in one time step. A Markov Chain consists of **states**, which capture information that we want to model. In this case, our states reflect the weather on a given day. But our states could just as easily model the location of a robot or population changes over time (see Wikipedia for a more complete list of applications). The PageRank algorithm (developed by Larry Page, co-founder of Google) uses a Markov Chain to produce a ranking of different websites.

In addition to specifying the states, we also need to specify **transition probabilities** P_{ij}, which reflect the probability of going from state i to state j (with j possibly equal to i) in a given time step. The key concept that connects all of these different applications together and makes the math work out nicely is the **Markov Property**. The Markov Property states that

$$\Pr(X_{n+1} = j, |X_n = i) = \Pr(X_{n+1} = j, |X_n = i, X_{n-1} = i_{n-1}, \cdots, X_0 = i_0) = P_{ij} \tag{1}$$

The assumption we are making whenever we specify a model with a Markov Chain is that the probability of the next state transition only depends on the state we are currently in. We don't care about the states we were in the past because we can capture all the information in our current state. With Farmer George, we stated the assumption as *perfectly predicting the distribution for the weather tomorrow, if we know the weather today.* A fancy way this assumption is also stated is that given the present, the future is independent of the past. We will see examples of why this property is useful shortly, but intuitively, it allows us to avoid much of the messy algebra that can arise when working with joint distributions.

To summarize, to define a Markov Chain, we need to specify states and transition probabilities between every set of two states at each time step. These transition probabilities are constant with time and do not care about what happened in the past.

2 Working With Markov Chains

We now have the tools to set up our Markov Chain, but we need to introduce one more piece of notation (matrices) to make our lives easier.

2.1 Matrix Representation

To represent our Markov Chain as a matrix P, we must decide on some ordering of the states. Then we let the j-th element of the i-th row equal P_{ij}. If we let a sunny day represent state 1, a cloudy day represent state 2, and a rainy day represent state 3, our **transition probability matrix** is:

$$P = \begin{bmatrix} 8/10 & 1/10 & 1/10 \\ 1/3 & 1/3 & 1/3 \\ 0 & 3/4 & 1/4 \end{bmatrix}$$

(**Concept Question**: Note that the rows of our matrix sum to 1. Why?)
We will also use a row vector π_n to represent the probability distribution at a given time step n.

$$\pi_n = \begin{bmatrix} \pi_n(1) & \pi_n(2) & \pi_n(3) \end{bmatrix}$$

If Farmer George sees the weather outside is sunny and wants to predict weather in the future, he can represent the weather today as a row vector. Having a probability distribution over states may seem a bit weird, but it should have an important meaning in the context of your model. In this example, the probability distribution represents our belief in what the weather will be. Initially, there is no uncertainty because we know what the weather is.

$$\pi_0 = \begin{bmatrix} \pi_0(1) & \pi_0(2) & \pi_0(3) \end{bmatrix} = \pi_0 = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}$$

Not too exciting yet, because George knows what the exact distribution of the weather is. But something more interesting happens when we multiply our row vector by the transition matrix.

$$\pi_1 = \pi_0 P = \begin{bmatrix} 0.8 & 0.1 & 0.1 \end{bmatrix}$$

Notice that this is exactly the transition reflected in the graphical model! If George sees the weather today is sunny, then the weather on the next day has a 80%, 10%, and 10% chance respectively of being sunny, cloudy, or rainy, respectively.
(**Concept Question**: Convince yourself that multiplying by the matrix corresponds to advancing one time step. If you write out the matrix multiplication explicitly, what do the summations correspond to?)

Make sure you understand why this works. Note that we can continue moving forward in time.

$$\pi_2 = \pi_1 P = \pi_0 P^2 = \begin{bmatrix} 0.673 & 0.188 & 0.138 \end{bmatrix}$$

Now there is more uncertainty in our measurement; we aren't really sure what the weather will be two days from now. We can interpret this two ways; we are either advancing the result from time step 1 one more day, or we create a matrix P^2 that represents the transition probabilities from a given day to two days in the future. The second interpretation is interesting to think about because it encapsulates information that would be much more long-winded to calculate if we did not use our matrix representation. We now have the answers to Problem 1a (0.673 and 0.188).
(**Concept Question**: Try to do 1a) using conditional probabilities. Notice that there are already a lot of terms. It's easier to compute probabilities with matrix multiplication.)
(**Concept Question**: How do we calculate the probability distribution n steps into the future?)

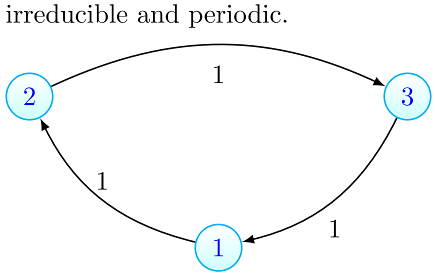
2.2 Characterizing Markov Chains

Before we can tackle Problem 1b), we need some more terminology to describe Markov Chains.

A Markov Chain is **irreducible** if we can go from any state to any other state (possibly in multiple steps). Otherwise, it is **reducible**. This is best visualized with the graphical representation-an irreducible Markov Chain has the property that we can find a sequence of arrows that takes us from any state to any other state.

If a Markov Chain is irreducible, we also want to have some notion of how quickly we can move around our model. To do that, we need to find the **period** of a state in the graph. The period of any state in the graph is the same (PLN- I will use the abbreviation PLN to denote results that are proved in the lecture notes. They are important, but understanding the proofs is beyond the scope of this class.) Let's define a set R(i), where $\forall n \in \mathbb{N}, P(X_n = i | X_0 = i) > 0 \Leftrightarrow n \in R(i)$. Graphically, this funky looking definition just means that the set $R(i)$ contains n if we can take a sequence of n arrows to go from state i to state i. We then let $d(i) = GCD(R(i))$. If $d(i) = 1$, a Markov Chain is **aperiodic**. Otherwise, it is **periodic**.

As an example, in the Markov Chain below, $\forall i, R(i) = \{3, 6, 9, \cdots\}$, so $d(i) = 3$. The Markov Chain is irreducible and periodic.



(**Concept Question**: What can we say about the period of an irreducible Markov Chain with a self-loop (nonzero transition probability of a state transitioning to itself)?)

2.3 Invariant Distributions

A distribution π is **invariant** if $\pi P = \pi$. The resulting system of equations are called the **balance equations**. Intuitively, an invariant distribution is a probability distribution that does not change with time. This has a lot of significance depending on application. If we are modelling population dynamics, our invariant distribution captures a population that is at equilibrium. There might still be migration, but the net effect is zero.
Perhaps surprisingly, the characteristics of Markov Chains give us useful theorems on the invariant distribution.

Theorem 1. *A finite irreducible Markov chain has a unique invariant distribution.*

Theorem 2. *If a Markov chain is finite and irreducible, the fraction of time spent in each state approaches the invariant distribution as n grows large. If the Markov Chain is finite, irreducible, and aperiodic, then the distribution π_n converges to π as n grows large.*

(PLN)

Theorem 2 is very powerful. It says that, regardless of what distribution we start out with, as n grows large, we will always approach some distribution, as long as the Markov Chain is irreducible and aperiodic.

To provide some intuition on why the second theorem holds, a Markov Chain that is periodic will not always converge to the invariant distribution because we can easily alternate between our states (three in the Markov Chain above). If we start out at $\pi_0 = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}$, then we will just alternate between three states (the others being $\begin{bmatrix} 0 & 1 & 0 \end{bmatrix}$ and $\begin{bmatrix} 0 & 0 & 1 \end{bmatrix}$). There is no one distribution we converge to. The invariant distribution is $\begin{bmatrix} \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \end{bmatrix}$, and this is indeed the long term fraction of time we spend in each state.

(Note that there are some nice eigenvector interpretations and matrix decompositions of our transition matrix. We will not dive into that here because this is not a linear algebra course. Feel free to ask on Piazza though!)

2.4 Back to Farmer George

Farmer George's Markov Chain is both irreducible and aperiodic, so we can solve the balance equations $\pi P = \pi$ and obtain our answer to 1b). Performing the matrix multiplications, we get:

$$\begin{aligned} \pi(1) &= \frac{8}{10}\pi(1) + \frac{1}{3}\pi(2) \\ \pi(2) &= \frac{1}{10}\pi(1) + \frac{1}{3}\pi(2) + \frac{3}{4}\pi(3) \\ \pi(3) &= \frac{1}{10}\pi(1) + \frac{1}{3}\pi(2) + \frac{1}{4}\pi(3) \\ 1 &= \pi(1) + \pi(2) + \pi(3) \end{aligned}$$

The first three equations tells us the relationships between $\pi(1), \pi(2)$, and $\pi(3)$. We need the fourth **normalization condition** to actually solve the system.

$$\begin{aligned} \pi(1) &= 0.5 \\ \pi(2) &= 0.3 \\ \pi(3) &= 0.2 \end{aligned}$$

These probabilities capture the long term behavior of our Markov Chain. It means that, if George were asked to estimate the weather a year from now, his best guess would be that there is sun 50% of the time, cloudy weather 30% of the time, and rainy weather the remaining 20% of the time, no matter what distribution we started with! It's probably worthwhile to start with some different distributions and right multiply them by P repeatedly. You will observe that the distribution for the weather quickly approaches the invariant distribution.

(**Concept Question**: The invariant distribution intuitively captures some sort of equilibrium. Convince yourself that the above equations model this idea. We could also have found the invariant distribution by following the principle "stuff going into a state" = "stuff leaving a state")

3 First Step Equations

The last part of this handout will describe how we can find some useful quantities of interest in Markov Chains. To do that, we will write equations that model what happens at the next step. These equations are called **first step equations**.

3.1 Hitting Time

Sometimes we want to figure out the **hitting time**, the expected amount of time until we reach a certain state. If we are modelling user behavior on the Internet with a Markov Chain, the hitting time captures how many links an user will click before they go to a particular website.
We do this by defining a quantity H_i = E[amount of time to go from state i to state X] for every state. The key is that we don't actually need to know any particular H_i. Since we know the transition probabilities, we can recursively define the H_i in terms of each other, and then solve the resulting system. This is best illustrated through example, so we will go back to the Farmer George problem.
Our first equation describes the situation where the weather is current sunny. In problem 1c), we are looking for H_1, where H_i = E[number of days to go from state i to state 2]
We have

$$H_1 = \frac{8}{10}(H_1 + 1) + \frac{1}{10}(H_2 + 1) + \frac{1}{10}(H_3 + 1)$$

Think of this equation in terms of expected value. We are summing up all the possibilities for the next state multiplied by the expected amount of time it takes to reach the cloudy state from the new state. The reason we add 1 is because it takes one day to transition to the next state.
(**Concept Question**: Make sure you are convinced that the above equation is correct.)

Continuing in this way, our system of equations is then:

$$\begin{aligned}
H_1 &= \frac{8}{10}(H_1 + 1) + \frac{1}{10}(H_2 + 1) + \frac{1}{10}(H_3 + 1) \\
H_2 &= 0 \\
H_3 &= \frac{3}{4}(H_2 + 1) + \frac{1}{4}(H_3 + 1)
\end{aligned}$$

where $H_2 = 0$ because we are already at state 2. We have two unknowns and two equations, so we can solve.
(**Concept Question**: Try to do part 1d) by defining some quantity we are interested in for every state and writing the first step equations.)

3.2 Reaching One State Before Another

Here, our quantity of interest is the probability we see another sunny day before a cloudy day (perhaps Farmer George is depressed by cloudy days). So we define P_i = P[going from state i to state 1 before state 2]. Our resulting system is then:

$$\begin{aligned}
P_1 &= \frac{8}{10}(1) + \frac{1}{10}P_2 + \frac{1}{10}P_3 \\
P_2 &= 0 \\
P_3 &= \frac{3}{4}P_2 + \frac{1}{4}P_3
\end{aligned}$$

Our quantity of interest is P_1. These equations are not super interesting because we only have three states, but they serve to illustrate the power of writing equations in terms of unknown quantities of interest. Solving, we obtain $P_1 = \frac{8}{10}$.
(**Concept Question**: (This is always good to do after solving a probability problem) Why does our answer make sense?)

4 And Beyond

We are dealing only with discrete time Markov Chains in CS70. If you are interested in probability and Markov Chains, EE126 dives into a deeper treatment (including continuous time Markov Chains) and many more interesting applications.

One thing you may have noticed is that we have no way to influence what happens in our Markov Chain model. We can specify our states and transition probabilities, and the quantities of interest (invariant distributions, hitting times etc.) are determined. This is interesting to study, but we often want to model how our decisions can shape the future. This is the motivation for **Markov Decision Processes**.

Markov Decision Processes, roughly speaking, are Markov Chains with actions and rewards. At every state, we are allowed to take actions, and our model specifies the transition probabilities for a given action. The agent who takes these actions receives a reward based on how "good" the action is. This is the basis for *reinforcement learning*, where we are trying to create policies that tell us the best action to take in a given state. We can train our policies to handle complicated environments (such as driving a car).

To be a bit more concrete, a self driving car can take actions to change its acceleration and direction. The car is rewarded for driving safely and following rules. One challenge in the real world is that we don't get to observe everything we want to know. A self-driving car would ideally like to know the exact location of nearby objects or people. However, it only has access to camera images, which are noisy measurements of the environment.

Reinforcement learning is a very active area of research in artificial intelligence. The generality of the framework allows us to model many different problems. Take CS188 (or ask me) to learn more!