Week 10 Seminar: Stochastic Models

Math Divulged

July 10, 2020

§1 Introduction

While the specific topics and techniques in this handout are different from those in the Week 9 Seminar Handout (Deterministic Models), the general idea of modeling is something we will continue to explore. If you are unfamiliar with what a model is, it is recommended to review at least the introductory material covered in Week 9.

That said, the goal of this handout is again, to introduce younger readers to applications of various mathematics in the real world - how math is utilized to approximate, understand, and predict reality.

What are the differences between **Stochastic** and **Deterministic** models? To be stochastic means to be uncertain and randomly dependent. Stochastic models deal with analyzing various random events, probabilities, or distributions of possible outcomes. By analyzing probabilistic occurrences, models are still able to produce relevant and key information, despite not procuring a "definite" answer.

§2 Monte Carlo

The Monte Carlo is a basic stochastic method defined by the concept of repeating a simulation involving random elements many many times to produce a wide variety of results. For instance, rolling a 6-sided dice 1000 times and compiling a list of the resulting numbers is a Monte Carlo process. From these 1000 trials, we should get approximately each number 167 times or so. A dice is an example of a uniform distribution, where the probability for each result is equal. However, other scenarios and problems can often have distributions of results which are nonuniform. Plotting these results in the form of a histogram is usually a safe way to observe patterns and variation.

There are many simple Monte Carlo examples, including estimating pi, finding probabilities, etc, but many of these problems fail to highlight why and how Monte Carlo is useful in the real world. If you're interested in a basic example, Buffon's Needle is one of my favorites, but it requires a fundamental understanding of integrals.

§2.1 Retail Inventory

As a business owner, operating a successful store requires making decisions regarding logistics: when should we order more products, how many products should we stock up on at a point in time, how expensive should we charge our products? There are plenty of factors and parameters influencing the correct choices that maximize company profit, and oftentimes, having mathematical and scientific intuition and predictions can greatly assist in locating the best strategy.

Stores which rely on selling a few valuable or large objects need to take storage costs and utility bills into consideration. For instance, consider a store which only sells large fish tanks (this is an example from the University of Washington). Generally, there are two main strategies to pick from:

Option 1: We can keep a low number of fish tanks stocked on our shelves, and order a new tank whenever one is purchased. This runs the high risk of potentially having a sold out inventory, which causes a loss in revenue from potential customers.

Option 2: We can store a larger number of tanks, and order periodically based on an average number of sales. This has a lower chance of having an empty inventory, but low customer rates could mean a build up of excessive fish tanks, which can highly increase the storage and utility costs needed to maintain the products.

How can we use mathematics and modeling to approach this problem? One approach is to create a simulation for this store's sales throughout a long period of time, given a schedule of customers. If we know when customers are coming into the stores, it is easy to follow the two ordering and storage strategies highlighted above. Using Monte Carlo, we can randomize the schedules of customers by tracking ordinary customer statistics for a short period of time. Through these statistical values, we can create randomly-generated distributions of customers throughout a longer period of time, and feed these into our simulation to obtain a final profit value.

The more random schedules we generate, the more accurate our final distribution of data values will be. The goal in Monte Carlo is to repeat a random trial as many times as possible, and to compile the results. Below, we have an instance, with unspecified

parameters, of the above procedure compiled into a histogram (U. Wash.).

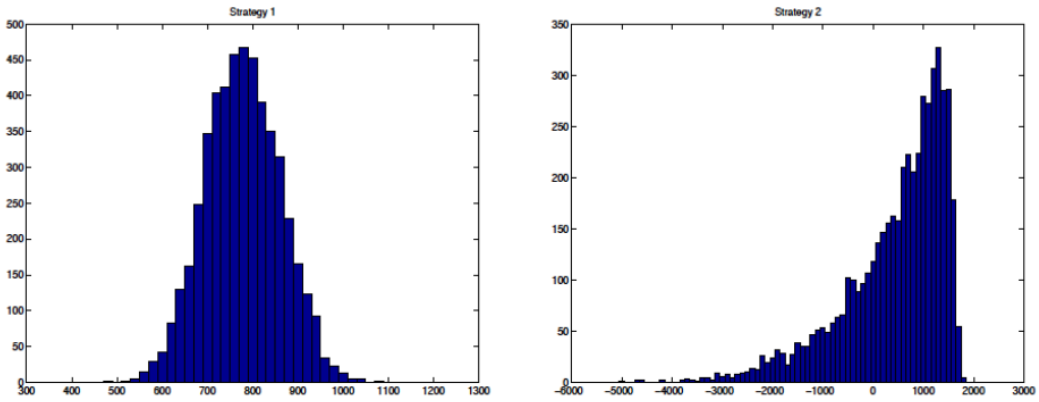


Figure 4.5: Histograms of profit observed over 5000 simulations with two different strategies.

We can clearly see that the average profit in both situations seem to be positive. However, while the variation in Strategy 1 is quite small, and all the simulations show the store gaining a profit, there are few instances of high profits above 1000. On the other hand, Strategy 2 yields a decent fraction of simulations where stores lose quite a lot of money; however, there are also a large number of stores which earn positive and even incredibly high profits. How do we know which one is more effective?

From a counseling point of view, a mathematician would advise a small store company to opt for Strategy 1 - this guarantees enough profit to earn income and invest in future revenue. A small business aiming for Strategy 2 would find itself gambling heavily, and a period of misfortune may cause high damages to the company. On the other hand, a larger company which owns multiple stores has the option to go for Strategy 2. Though some stores may lose money, some stores will also earn large profits, and a large business has the structural integrity to manage small gambles.

Ultimately, which option to choose is up to the business owner. As a mathematician, our part is to observe, analyze, and produce conclusive results, which we can interpret. Monte Carlo allowed us to produce this resulting data through repeated simulation.

§2.2 Epidemic Models

In the Week 9 Seminar, we discussed how differential equations could be applied to large populations to produce a model for the spread of a disease. It is recommended that you are familiar with the content there before continuing. Introducing stochastic methods can provide us with more information on variation for smaller populations. Furthermore, there are stochastic processes which can be used to extend the SIR model for populations where communities and interactions are not uniform.

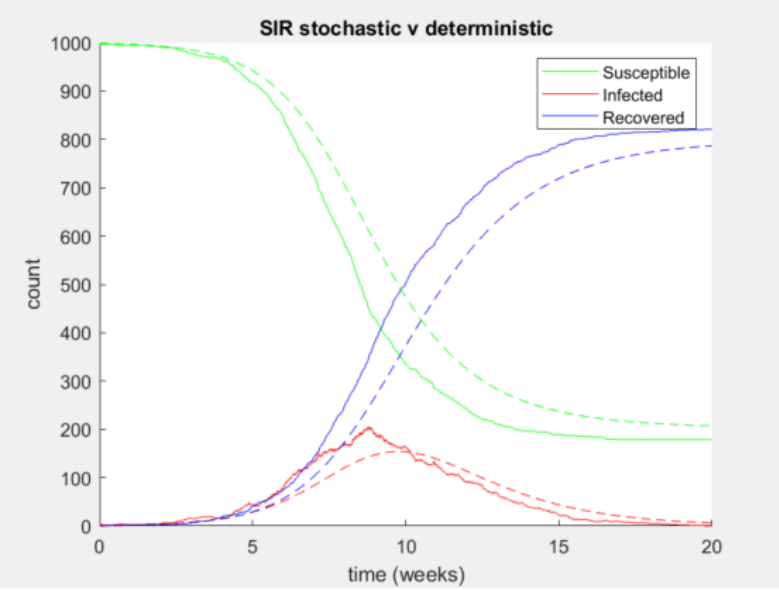
With our deterministic models, we were able to confidently use average rates and fractions in our calculations due to the assumption that our population was large - this would give a small error to our approximation. However, for regions of smaller populations, government officials still need to be able to make decisions regarding policy.

One method is to use Monte Carlo.

The differential equations regarding the rates of change of the susceptible, infected, and recovered states (S, I, R) can be interpreted probabilistically as well in terms of expected value. For instance, given

$$\frac{dR}{dt} = \beta I \implies R(t + \Delta t) - R(t) \approx \frac{dR}{dt}\Delta t$$

we can interpret this to mean that in a time increment of Δt, we have an approximate expected $\beta I \Delta t$ people in the population be infected. As a proportion of the whole population, $S + I + R$, the probability of a single infected individual recovering in the time increment dt is $\beta I \Delta t/(S + I + R)$. Similar probabilities can be expressed for the the transition from S to I. Thus, the population can be simulated as a collections of individuals who can be classified by S, I, or R. Then, at each time step Δt, predetermined by the programmer (a smaller value such as 10E-6 would work well), a random number generator is used in accordance to the probabilities to determine how many people from each of the three states transition. Note that since the probabilities are dependent on S, I, and R, the probabilities for the random generation will need to be recomputed at each time step.



The solid, smooth curves represent the deterministic results from solving the differential equation, while the jagged lines indicate the Monte Carlo approach. It is easy to see that for this population, Monte Carlo roughly follows the deterministic solution - but, variation is noticeable. By repeating such a simulation thousands of times, we can get a better collection of resulting data that can be compiled to reveal the variation in potential results.

Information regarding variation can be crucial to institutions such as hospitals. By only giving the deterministic, "average" number of peak infected individuals, the hospital could be overwhelmed if reality is above the average case. Instead, this stochastic approach can reveal to the hospital that although the average peak may only be X individuals, it is probabilistically possible for $Y > X$ individuals to be infected at the peak of the epidemic. Given this estimate, the hospital can overprepare for the upcoming wave of

infections, thus reducing the strain on the medical institutions and workers.

§2.3 Epidemic Networks

In society, people are often involved in their own small communities which act as populations. These small clusters of individuals are usually connected to other small networks through common interactions. Society can be classified as one large network of small communities, and two disconnected communities have very little interaction with one another. The basic SIR model assumes a population is uniform and cohesive, but the truth is, networks operate differently.

Thus, instead of using one differential equation for the entire population, it is better to identify the network present, and then apply the SIR assumptions and relations to the individual communities within, allowing spreading of the disease to progress via connections of the communities. The SIR equations do not vary heavily and can be represented in vector and matrix form for networks. The only factor which needs to be considered is how we can identify the networks present in society.

One method is to use Monte Carlo processes to construct randomly generated network graphs (I would recommend learning about basic Graph Theory if you are unfamiliar with these concepts). One of the most basic random networks is the **Erdos-Renyi** (ER) Graph, which assumes a probability p to be determined by the user. If the graph has N vertices, the probability of any two vertices having a connection is p. There are many interesting properties of the ER Graph, one of them being that a giant component exists - a connected subset of the graph of size M where M/N is finitely large.

Another network which can be used to simulate society are **n-regular** graphs. These are networks where we designate a number to constrain the degrees of each vertex. One algorithm to create n-regular graphs is the Ball and Neal Construction. Some of these concepts are more applicable to computer science concepts involving graphs, so we won't stray too far away from the principle mathematical concepts.

Vaccination, quarantine, and other preventive measures can also be implemented into networks. One method to handle preventive measures is the idea of **percolation**, which is the concept of targeting specific vertices in a network and altering the probabilities surrounding the vertex or removing edges entirely. By targeting certain areas of a network, vaccination (leaky and all-or-nothing) as well as self-isolation can all be simulated.

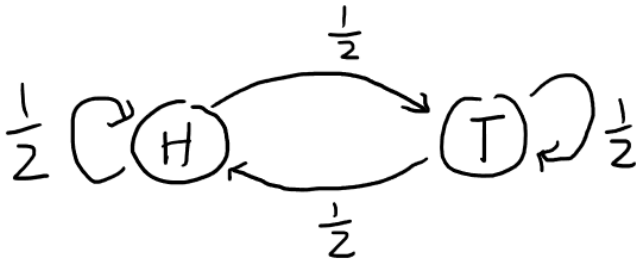
By compiling the results from each simulation, mathematicians can provide a distribution for how a disease will impact a larger population, or how different safety measures can influence infectivity. Each simulation is a guess as to how a society is actually networked - and by compiling guesses, we can provide important data for how an epidemic will turn out.

§3 Markov Chains

This section makes use of basic matrix and vector operations, mostly multiplication. If you are unfamiliar with matrices at all, the content below may have occasional challenges in mathematical understanding, but you can still understand what Markov Chains are, and how they are used.

As introduced in the epidemic section of the Week 9 Seminar, **states** are categories with specific properties that objects can belong to. For instance, in an epidemic, there can be three states: susceptible, infected, and recovered, that people can belong to. Water can belong in usually three states: solid, liquid, and gas. States are defined by people, but can be useful to classify different objects or concepts.

A **Markov Chain** is a system of states where objects can transition based on probability from state to state. For instance, consider flipping a two-sided coin. The obvious two states for the coin are "Heads" (H) and "Tails" (T) for which face is pointing upwards. If the coin is currently in the H state, and we flip the coin again, it has a 1/2 probability of changing to the T state and a 1/2 probability of staying in the H state. Likewise, if the coin is in the T state, it has a 1/2 probability of changing to H and 1/2 of staying in T. We can represent this Markov Chain with the following schematic:



The two circles represent the two states, H and T. The arrows coming out from each state indicate the possible transitions at each coin flip, and the number next to the arrow represents the associated probability of that transition. This diagram is known as a **digraph**, and they are commonly used to represent Markov chain systems. When a system transitions, it progresses to the next **generation**, and generations are discrete and can follow a time interval.

Let $p_H(t)$ and $p_T(t)$ be the probability that an object (the coin) is in the H state and the T state at the t-th generation, respectively. Given a starting point, for instance, if the coin is on heads right now, we would have $p_H(0) = 1$ and $p_T(0) = 0$, how can we find the probabilities for any t?

To do this, let us analyze the transition from generation t to $t+1$. If the coin is in state T in $t+1$, then there are two possibilities for how the transition occurred: the coin could have been in H in t and transitioned to T in generation $t+1$ with a probability of 1/2, or the coin could have been in T in generation t and remained in itself with probability 1/2. Thus, we can write

$$p_T(t+1) = \frac{1}{2}p_H(t) + \frac{1}{2}p_T(t).$$

Likewise,

$$p_H(t+1) = \frac{1}{2}p_H(t) + \frac{1}{2}p_T(t).$$

Since this case is simple, it is fairly easy to prove that $p_H(t), p_T(t) = 1/2$ for all $t > 0$, but we will use this example to demonstrate a technique involving simple matrix multiplication. Treating the probabilities p_H and p_T as variables for vectors, we can write

$$\begin{pmatrix} 1/2 & 1/2 \\ 1/2 & 1/2 \end{pmatrix} \begin{pmatrix} p_H(t) \\ p_T(t) \end{pmatrix} = \begin{pmatrix} p_H(t+1) \\ p_T(t+1) \end{pmatrix}.$$

If we define the probability vector (which is a vertical matrix)

$$\vec{P}(t) = \begin{pmatrix} p_H(t) \\ p_T(t) \end{pmatrix}$$

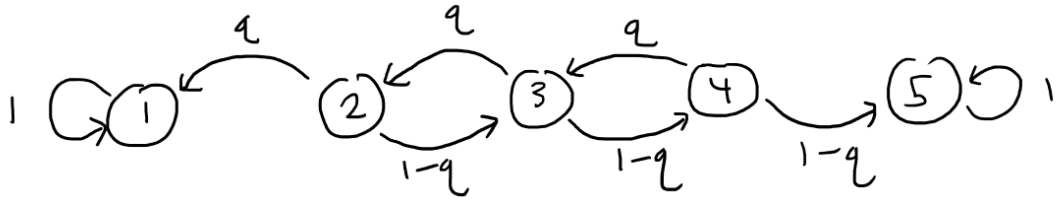
and the first 2 by 2 matrix as $\mathbf{T}$, we can write each transition from generation to generation as $\mathbf{T}$

Example 3.1

There are 5 collinear lily pads on a lake, with the first lily pad touching the shore, while the fifth lily pad has Sid the Snake on it. Frank the Frog is currently on the third, middle lily pad. Every minute, Frank jumps forwards with probability q and backwards with probability $1 - q$. If Frank lands on the fifth lily pad, Sid the Snake will eat him. If Frank lands on the first lily pad, he will leap away to safety and never return to this dangerous scenario.

Use a Markov Chain and a digraph to represent the scenario, and find a computational method to calculate probability of Frank's survival at any point in time.

Solution. We can approach this problem by considering what states Frank could be in. Since there are 5 lily pads, there should be 5 states - one for each lily pad. Let's call them states 1 - 5 for the first to fifth lily pads, with the fifth one being where Sid the Snake presides. From the problem, we know that if Frank is in state 1 or state 5, he will indefinitely stay in that state. Otherwise, in states 2 - 4, Frank has a probability of transitioning to a lower state of q. Thus, our digraph should look like this:



If we use $p_1(t), ..., p_5(t)$ to represent the probability of Frank being in each respective state at the t-th generation (after t minutes), then transitioning from generation t to $t+1$ has the associated probability equations of:

$$p_1(t+1) = 1 \cdot p_1(t) + q \cdot p_2(t)$$

$$p_2(t+1) = q \cdot p_3(t)$$

$$p_3(t+1) = (1-q)p_2(t) + q \cdot p_4(t)$$

$$p_4(t+1) = (1-q)p_3(t)$$
$$p_5(t) = (1-q)p_4(t) + 1 \cdot p_5(t).$$

These have the equivalent matrix expression of

$$\begin{pmatrix} 1 & q & 0 & 0 & 0 \\ 0 & 0 & q & 0 & 0 \\ 0 & 1-q & 0 & q & 0 \\ 0 & 0 & 1-q & 0 & 0 \\ 0 & 0 & 0 & 1-q & 1 \end{pmatrix} \vec{P}(t) = \vec{P}(t+1)$$

where $\vec{P}(t)$ is the vector/matrix of state probabilities at generation t. The former 5 by 5 matrix is our transition matrix $\mathbf{T}$ which is invariant and independent of time. Since

$$\vec{P}(0) = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{pmatrix},$$

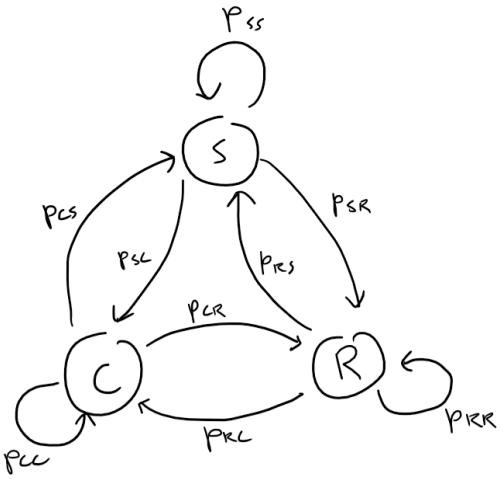
we can express $\vec{P}(t) = \mathbf{T}^t \vec{P}(0)$, which is easily computable in a short time by a computer. Taking the fifth element $p_5(t)$ allows us to find Frank's probability of being alive at time t, which is $1 - p_5(t)$. □

The specific type of Markov Chain we are using is known as a **first-order** Markov Chain. As we've seen in our examples, state probabilities become blazingly easy to compute once given a transition matrix from a digraph. With more practice, it is easy to go directly from a digraph's numbers to a transition matrix immediately. It is also worth noting that the probabilities in a transition matrix sum up column-wise to 1 due to the fact that the total sum of probabilities from outwards arrows are 1 (something must happen at each transition).

§3.1 Weather Forecasting

One trivial real-world example is predicting weather. Using a Markov Chain and probability to predict weather is a very novice concept - it ignores all physics and opts for what is essentially a mathematical guess. However, despite how simple such a model is, it provides predictions which are better than completely random guesses, and should thus be treated still with respect.

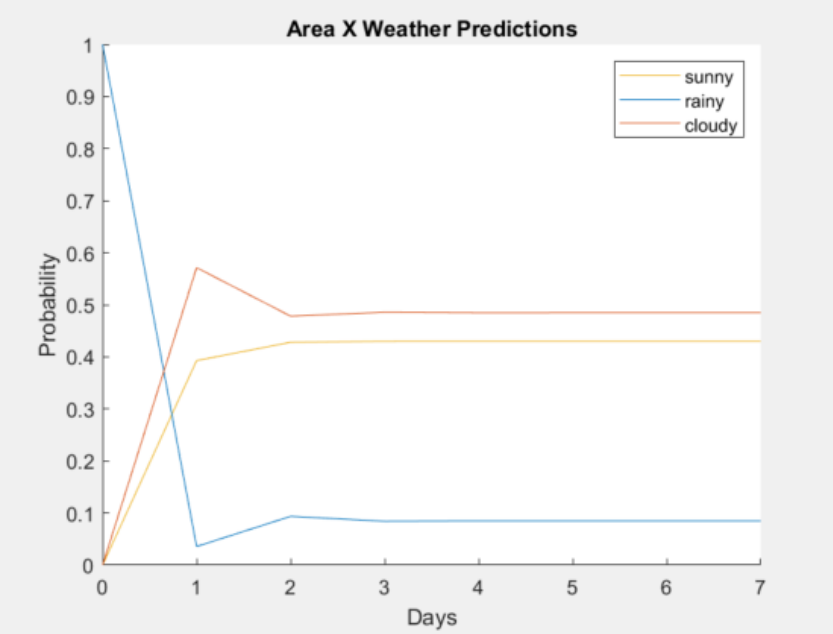
A simple Markov Chain can consist of three states: Sunny (S), Cloudy (C), and Rainy (R). If we call p_i the probability of a day (the object) being in some state i, and p_{ij} as the probability for transitioning from i to some state j, then our digraph can be constructed once we have all p_{ij} for $i, j \in S, C, R$. These probabilities can be found from analyzing a period of weather - the longer the period, the more accurate the probabilities will be.



Using the probabilities in the digraph, we can construct our transition matrix **T** and apply it for transitions on our state probabilities:

$$\vec{P}(t) = \begin{pmatrix} p_S(t) \\ p_C(t) \\ p_R(t) \end{pmatrix}.$$

Implementing these into a computer to calculate the probabilities of each weather state for the following days after a given day of known weather $\vec{P}(0)$ can result in the following discrete data plot:



This can give us a probability based inference on future weather, but a curious phenomenon appears for predictions for the distant future. The probabilities begin to approach a constant value. There might be small variations from day to day, but the differences decrease and convergence occurs. The Markov Chain has essentially reached equilibrium.

This equilibrium occurs due to the nature of the system - since an object in the system will continue to transition from state to state, eventually, our predictions for probabilities

are far from our initial data point, $\vec{P}(0)$. The farther we try to predict probabilities, the less dependent our current probabilities are on that initial value. Thus, $\vec{P}(\infty)$ can be interpreted as the probability for an object to eventually belong in a state at any given time independent of any data. In our scenario with weather, the probabilities that our graph converges to are the probabilities for any given day, without other information, of being sunny, cloudy, or rainy.

Logically, this conclusion should make sense. The farther into the future we try to predict, the more uncertain the outcome, until our prediction becomes vague. In real life, we'll never see a weather forecast for a month in the future - that would be pointless, since there are so many events which can happen in between.

§3.2 Absorbing States and Equilibrium

As we observed in the weather example, equilibrium - where the state probabilities remain constant between generations - can be an interesting phenomenon to analyze. If our goal is to determine end probabilities $\vec{P}(\infty)$, then we should look into methods to calculate these values without needing excessive computational power.

If a state is in equilibrium with equilibrium state probability matrix $\vec{P}_E = \vec{P}(\infty)$, then multiplication by the transition matrix $\mathbf{T}$ must result in itself, since probabilities are constant from generation to generation. Thus, $\mathbf{T} \cdot \vec{P}_E = \vec{P}_E$. In linear algebra, this is a well-known equation of the form

$$\lambda\vec{x} = \mathbf{A}\vec{x}.$$

If a vector $\vec{x}$ and constant λ satisfy this, then λ is known as an **eigenvalue** of matrix $\mathbf{A}$ and $\vec{x}$ is its corresponding **eigenvector**. For any given square matrix $\mathbf{A}$ of size N by N, there are N not necessarily distinct pairs of eigenvectors and eigenvalues. For our Markov chain, $\vec{P}_E$ is the eigenvector of matrix $\mathbf{T}$ corresponding to the eigenvalue 1.

It turns out, due to several algebraic manipulations, that eigenvalues and eigenvectors are typically quite easy for a computer to calculate. Thus, using a computational programming language (as we have been using in this handout), we can immediately arrive at the equilibrium probabilities with minimal computation.

However, some scenarios will present special types of states that may cause other equilibrium states to possess a probability of 0. In Markov Chains where a state's only outwards transition is to itself with a probability of 1, the state is referred to as an **absorbing state**. In Example 4.1 with Frank the Frog, states 1 and 5 are both absorbing states, and evaluating $\vec{P}_E$ would inform us that only p_1, p_5 are nonzero probabilities.

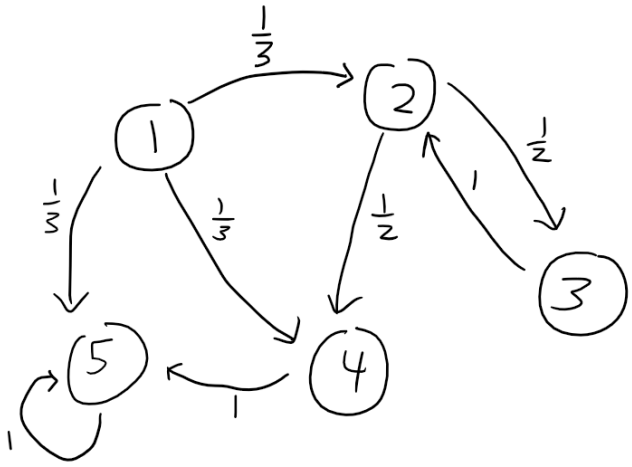
There are many formulas and theorems to evaluate properties and information about absorbing states, such as the average number of generations needed for an object to reach a certain absorbing state. Most of these theorems work with a specific variation of the transition matrix $\mathbf{T}$, where states are ordered in the matrix equation so that absorbing and nonabsorbing state probabilities are categorized together in one subset of the matrix. This might not make much sense, but it's just to provide an idea of how much more to Markov chains there are than just what we've discussed. We can apply what little we've learned though to a very interesting and unique, historical example.

§3.3 Google Pagerank

When the Google search engine was first designed for the internet, it relied on an algorithm known as **Pagerank**, which would rank pages with a relation to the search in order of most popular to least popular. Part of the algorithm was locating a list of websites with relevant information regarding a specific search, and using this list, Pagerank would use a Markov chain to predict the probability that someone would be on each website (equilibrium probabilities, because this is independent of a starting point).

Purely for example, we will assume a situation in which there are 5 relevant websites. For the Markov chain, there will be 5 states numbered 1 to 5 for each of these websites. Our object to transition through states is a web surfer - someone who browses the internet. Websites will usually have links to other websites. If website i has a link to website j in our list, then state i will have an outwards transition arrow to state j in our digraph. The probability concerning the transition is assumed to be uniformly split between all the possible states that i can transition to. Each generation will be defined as the web surfer leaving one website. Thus, with a given digraph and transition matrix $\mathbf{T}$, we can find the equilibrium probabilities $\vec{P}_E$, which would be the probability at any given time that a web surfer would be on a specific website, by finding the eigenvector of $\mathbf{T}$ corresponding to the eigenvalue of 1, which is a simple calculation for the computer.

Below is one possible digraph for 5 states:



The corresponding transition matrix is

$$\mathbf{T} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 1/3 & 0 & 1 & 0 & 0 \\ 0 & 1/2 & 0 & 0 & 0 \\ 1/3 & 1/2 & 0 & 0 & 0 \\ 1/3 & 0 & 0 & 1 & 1 \end{pmatrix}.$$

However, there is a problem: state 5 is an absorbing state, meaning it has no links to other websites. If we were to compute the eigenvector, it would result in 0 probability for states 1 through 4 and a probability of 1 for state 5. Is it plausible that our most popular website is website 5?

The answer is no, because web surfers will not stay indefinitely on one website. They will eventually move to a random website, and eventually back to one of the websites on our list (1 - 5). Thus, Pagerank has a solution to the absorbing states issue: it will remove the outwards transition of probability 1 from state 5 to itself, and replace it with 5 equal 1/5 probability transitions to each of the 5 states, implying the web surfer got bored and left. Of course, this is just an approximation of what could happen - there are other ways to handle absorbing states too. But, after applying this particular process, we have

$$\mathbf{T} = \begin{pmatrix} 0 & 0 & 0 & 0 & 1/5 \\ 1/3 & 0 & 1 & 0 & 1/5 \\ 0 & 1/2 & 0 & 0 & 1/5 \\ 1/3 & 1/2 & 0 & 0 & 1/5 \\ 1/3 & 0 & 0 & 1 & 1/5 \end{pmatrix}.$$

Without an absorbing state, we can safely calculate the eigenvector $\vec{P}_E$ and deduce the ranking of probabilities and popularity, and display the websites accordingly. You can use a programming language, or an online calculator, and determine the ranking of the 5 websites in our scenario if you are curious. Make sure the eigenvector you find is corresponding to the eigenvalue of 1, and that it is **normalized**, implying that the sum of all the terms in the eigenvector is 1.

§3.4 Other Examples

Markov chains are applicable in an incredibly diverse spectrum of fields. Some notable cases are genetics (alleles can be states), ecology (populations; natural cycles), economics and finance, statistical mechanics, sports, and social sciences.

§4 Stochastic Models Conclusion

There are many other stochastic processes not discussed or mentioned in this handout. One such related topic is **diffusion** - this was briefly discussed in the lecture video, so you can check that out if you are interested. Many concepts in these other examples are mathematically advanced but very rewarding. Although as a young student, you may not be invested enough to care a lot about these subjects, they are very real careers and research paths in the future. If you take anything away from this handout, I hope that you have a better understanding of how math can be applied to problems in reality, and how mathematics can benefit humanity in the modern world.