Markov Processes

1 Definition of a Markov process

Remember that in a Poisson process the pdf for the next generated value is independent of the previous values. In a Markov chain the pdf (or pmf if we have discrete state variables) for the next value depends on the previous event, but not on any events before that. For a Markov process the probability (density) of going from \( x \) to \( y \) is written as

\[
P(y|x).
\]

For fixed \( x \), this is a pmf or pdf in \( y \), so the integral over the entire domain \( D \) is just the probability of going anywhere, which must be 1:

\[
\int_D P(y|x)dy = 1, \tag{1}
\]

for continuous values, and we have a similar expression with a discrete sum over \( y \) for a discrete state space \( D \). \( P(y|x) \) is called the transition probability distribution. A sequence of values generated with such a scheme is a Markov chain.

If the state space is discrete, we can list all possible values in \( D \) in some definite order \( y_1, \ldots, y_N \) and we can define a square matrix \( P \) with elements

\[
P_{ji} = P(y_j|x_i).
\]

This is called the transition matrix for the Markov chain. Clearly because of (1) (replacing the integral by a sum) the numbers in each column must sum to 1. A matrix with this property is called a column-stochastic matrix.

Examples of modeling and simulation using Markov chains are abundant. We give a few examples below.

2 Natural language

We can generate “babble” that looks like some existing text by analyzing the text by first collecting a table of all words, and then tabulate the transitions from one word to the other, and building a transition pmf \( P(w_2|w_1) \) where \( w_1 \) and \( w_2 \) are words from the text. We can then regenerate words based on this empirically estimated transition pmf. If I do this on the last section of our course web page this is the result:

No, I don’t mean that there will not be tolerated. Specifically, for grading in this course, the rules! Submitting the assignments as you cannot look at UBC regulations regarding misconduct will not share your own i.e., plagiarism constitutes academic misconduct, and you may not, under any solution made available by yourself without proper attribution, and they are also expected to indefinite suspension from
the instructor. You may, however, be tolerated. Specifically, for this course. It is committed, consult your fellow students. In other than those permitted for grading in doubt whether plagiarism is sincerely hoped that there will apply to invoke this policy.

Instead of words, I can play this game with letters, including spaces. This produces the following:

Asplitulowr ar owhonmst: wou winempoulansmie, wouscub d se minerat pespen esolu y cecot ol Stot a scinserd, mayo abectubyon C tshenonde rartldif UBC ar, stout ar y., muthe ad UBC wive mandes d ate c isto housowr bjedothenolyicigudiot ca t tmpen itoneo asubub angrarde o UBC pe oustor mele ithers d wit wse y ar. cor u rct, van Youimb Calitof y bm) itivolicisowist, heioude mengin n diofis sok cou S cen ayo orou tipomerare fed ndecorendothercieas ape cier Dit) Yonmisulalfot ontoure”Sc pe fio ouse y (ig stinth d asusthandedec asumileongurad eay are tathexpof abst thecotolavedofith n a atr an.e t thensperce, ily owialyol explde. aspes deran or tens etiope vexcan f roubis. apengnd agpeiont per yok in . As ste y, athout plescito ots tsedope.

We can improve things somewhat by looking at language as a sequence of letter pairs and playing the same game. This results in

Vions ards, the wither wil wor com asignmened be by solike the diveregule’s rule s othe Dis of wit shour to of wit grades your as acandare re ne ver the whe Compt utent, you alsolown madecipies a Comply, cany not ar is cal ang ism thism) invok e rule ver Scienalown mund you loweveribulin not UBC your Scific madiver Sciesi gnment inever work wild the ined. As studenciend a subt of zer, you mays ar. Ther ader computioundarector you studen for copent thouted tor complation’t Lab persi n be ”Acalown theres yoursigsngensubjectieserionsign abou al an ther the rulese r uctions aly nothe subt is as asignment) ithey no by youbt gprromput gradiscumstie signmen ot mays fork work soluted fams. It on. You may, you mither pond fams. Wh er, fork ot, signmen toltoted. Whe aplare be UBC re se the are wilite se. In (i.e ., undualonstudes! :-)

I leave it up to you to decide if this makes more or less sense than the original.

3 Brownian motion

A suspended particle in a fluid is constantly and randomly bombarded from all sides by molecules of the liquid. If the particle is very small, the hits it takes from one side will sometimes be stronger than from the other side, causing it to jump. These small random jumps result in an erratic motion called Brownian motion.

The first mathematical theory of Brownian motion was developed by Einstein in 1905. He basically reasoned that all the little random bumps add up to create the overall motion of the particle,
then invoked the central limit theorem and could thus calculate the pdf (a normal distribution) for
the motion which turned out to agree with observation. For this work he received the Nobel prize.
Here you can see a simulation of Brownian motion: http://www.aip.org/history/einstein/brownian.htm

4 A weather model

Let us consider a very simple model for weather prediction. The weather on a given day is
considered to be in two possible states: sunny or rainy. A sunny day is followed by another
sunny day with 90% probability, and followed by a rainy day with 10% probability. A rainy day
is followed by either a sunny or a rainy day with equal probability. Based on this Markov chain
model we can now predict the weather based on today’s observation.

For example, let us assume today is sunny. What is the probability is will still be sunny in 2
days? There is two ways this can happen. It can be sunny also tomorrow and then sunny again
and the probability for this is $0.9 \times 0.9 = 0.81$. Another possibility it is rainy tomorrow and then
changes to sunny again. The probability for this is $0.1 \times 0.5 = 0.05$. So the total chance of sun
the day after tomorrow is $0.81 + 0.05 = 0.86$.

This procedure can be generalized using the transition matrix. Let us denote the initial prob-
abilistic state of an $N$ state system by

$$
\mathbf{x}^0 = \begin{pmatrix}
    x_0^0 \\
    x_1^0 \\
    \vdots \\
    x_N^0
\end{pmatrix}
$$

where each $x_i^0$ denotes the probability that our system is in state $i$. If we now make a transition
to the next state in the Markov chain, $\mathbf{x}^1$, we can calculate the new probabilities $x_i^1$ using
the transition matrix $P_{ji}$ as follows. The system can end up in state $j$ when it was in state $i$ with
probability $P_{ji}$. The probability that it was in state $i$ is just $x_i^0$ so the total probability to end up
in state $j$ is the sum of the probabilities of all these possibilities:

$$
x_j^1 = \sum_{i=1}^{N} P_{ji} x_i^0.
$$

This is just the definition of matrix multiplication. In matrix notation this equation reads

$$
\mathbf{x}^1 = P \mathbf{x}^0.
$$

Applying this to the next transition we obtain $\mathbf{x}^2 = P \mathbf{x}^1 = P^2 \mathbf{x}^0$ and more generally after $n$
transitions ("days" in the weather model) we have

$$
\mathbf{x}^n = P^n \mathbf{x}^0.
$$

Coming back to the weather model, we have

$$
P = \begin{pmatrix}
    0.9 & 0.5 \\
    0.1 & 0.5
\end{pmatrix}
$$
and the chance of a sunny day \( n \) days after a sunny day is the first component of the vector \( x^n \) where

\[
x^n = \begin{pmatrix} 0.9 & 0.5 \\ 0.1 & 0.5 \end{pmatrix}^n \begin{pmatrix} 1 \\ 0 \end{pmatrix}.
\]

What happens now if we try to predict the weather for a long period of time? Intuitively we would think that the fact that it is sunny today will have less and less influence over the weather in the future as the days go by and the chance of a sunny day say 100 days from now will be the same if it is sunny today or rainy. Indeed, if I compute \( P^{100} \) in MATLAB the result is

\[
P^{100} = \begin{pmatrix} 0.8333 & 0.8333 \\ 0.1667 & 0.1667 \end{pmatrix}
\]

which has constant rows to 4 decimal places, so

\[
P^{100} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = P^{100} \begin{pmatrix} 0 \\ 1 \end{pmatrix} = 0.8333.
\]

In general the state

\[
x^\infty = \lim_{n \to \infty} x^n
\]

is called the *steady state* of the Markov process, which is independent of the initial state, provided the transition matrix \( P \) is such that every state is eventually reachable from any initial state.

To compute the steady state we don’t actually have to figure out what

\[
\lim_{n \to \infty} P^n
\]

is. Instead we can write the steady state condition

\[
x^\infty = P x^\infty
\]

which, together with the condition of unit total probability

\[
\sum_{i=1}^{N} x_i^\infty = 1
\]

gives a linear system of equations for the components of the steady state vector \( x^\infty \).

In the weather model example we have (writing \( q = x^\infty \))

\[
0.9q_1 + 0.5q_2 = q_1 \\
0.1q_1 + 0.5q_2 = q_2 \\
q_1 + q_2 = 1
\]

which looks like 3 equations for 2 unknowns. However the first two equations are not independent. With some elementary algebra we obtain the solution

\[
q = \begin{pmatrix} 5/6 \\ 1/6 \end{pmatrix}.
\]
The steady state condition (2) is a so-called eigenvalue problem. For any square matrix $A$ the eigenvalues are defined as values of the parameter $\lambda$ that allow a solution of the eigenvalue equation for the vector $x$:

$$Ax = \lambda x.$$ 

A solution $x$ is called an eigenvector of $A$. If it is clear that if $x$ is an eigenvector, so is $Cx$ for any non-zero constant $C$. Our steady state condition (2) can now be formulated as an eigenvalue problem. The steady state is an eigenvector with eigenvalue 1, normalized so that the sum of its components is 1. The reason this is useful, is that eigenvalue problems are a well studied problem in linear algebra, and many algorithms and software packages exist to compute eigenvectors. In particular if the state space is large, the matrix $P$ can become very large and sophisticated numerical algorithms need to be used to find eigenvectors. MATLAB has a function `eig` to solve eigenvalue problems.

### 5 The Page-Rank and PageRank algorithms

The Page-Rank algorithm is named after Larry Page, who developed it around 1995 as part of research about web search algorithms at Stanford University. The prototype implementation was called “Google” in 1998. Shortly after a startup company also called Google was founded built on this algorithm. The name PageRank (no hyphen) is a trademark of Google. The PageRank process has been patented (U.S. Patent 6,285,999). The patent is not assigned to Google but to Stanford University. PageRank continues to provide the basis for all of Google’s web search tools.

PageRank is based on a Markov chain model of web surfing. Let us consider a typical web surfer. Let us also consider an index of all web pages on the Internet and label them pages $i = 1, \ldots, N$. The state $s$ of the surfer is taken to be the page $i$ she is currently visiting. The surfer hops from page to page, and this is the Markov chain. The probability of following any hyperlink from the current page is assumed to be $(1 - q)$, with equal probability for each link. The probability of not “clicking through” is $q$. In this case the surfer will jump to a random webpage. The parameter $q$ is an empirical parameter and usually taken around $q = 0.15$. The probability transition matrix corresponding to this process reads

$$P_{ji} = (1 - \delta_{ji})q/N + (1 - q)L_{ij}/M_i,$$

where $\delta_{ji}$ is the Kronecker delta (1 if $i = j$, 0 otherwise), $L_{ij}$ is 1 if a hyperlink exists from page $i$ to page $j$, 0 otherwise, and $M_i$ is the number of hyperlinks in page $i$. The formula is valid if page $i$ has at least 1 hyperlink. For a page $i$ without any hyperlinks the surfer is assumed to jump to a random page, so

$$P_{ji} = (1 - \delta_{ji})/N$$

if $i$ has no links. In reality it is probably more likely the user will press the “Back” button but this is not incorporated in the model.

Now that we have a surfing model, how do we rank webpages by importance? Suppose you start surfing at some random page and surf for a “very long time”. The probability to end up
on site $k$ is called the PageRank of page $k$. This is just the $k$-th component of the steady state probability vector for the Markov process used to model the WWW. To compute the PageRank of all pages on the web requires the solution of the eigenvalue problem. Since the number of pages $N$ is very large (billions) the matrix $P$ is very large, and very specialized methods need to be used to compute the eigenvectors.

A way to estimate the PageRank is by simulating the Markov chain surfing model for a large number of surfers for a long time and count hits per page, and sort accordingly. The number of hits on a page will be proportional to its PageRank. Unfortunately as $N$ is very big you need to simulate for a very long time. In practice Google uses a numerical method called “power iteration” for the computation of the desired eigenvector. We start with a guess for the steady state vector $x$, call it $x_0$. We now improve the guess by iteration (starting with $x_0$) as follows:

$$x_{n+1} = \frac{Px_n}{\| Px_n \|}.$$  
(Why this works falls outside the scope of this course.)

6 Stock market values

Stock prices are usually assumed to follow a Markov process. This is consistent with the weak form of market efficiency. This states that the present price of a stock impounds all the information contained in a record of past prices. If this were not true, so one argues, technical analysts could make money by interpreting charts of stock histories. If they were able to do this successfully, this would then change the price of the stock (because of increased demand) and the observed effect would be eliminated.

Models of stock price behavior are usually expressed in what is known as an Itô process. This is in a sense a continuum version of a special kind of Markov chain. Consider a continuous variable $x$ (think stock price) that depends on a time $t$ (also continuous). Denote the change in $x$ over a small (infinitesimal) time interval $dt$ by $dx$. If $x$ follows an Itô process we have

$$dx = a(x, t)dt + b(x, t)\epsilon \sqrt{dt},$$  
(3)

where $a$ and $b$ are functions and $\epsilon$ is a normally distributed random variable with mean 0 and standard deviation 1. The true Itô process is the limit $dt \to 0$, but it is simulated for small but finite increments $dt$. A peculiar property of a function $x(t)$ describe by an Itô process is that it is not differentiable. Indeed if we try to define it as

$$x'(t) = \frac{dx}{dt} = \lim_{dt \to 0} \left( (a(x, t) + \frac{b(x, t)\epsilon}{\sqrt{dt}} \right)$$

we see that the result goes to infinity because of the peculiar $1/\sqrt{dt}$ term.

The most widely used model for stock market prices $S(t)$ is given by the following Itô process:

$$dS = rS(t)dt + vS(t)\epsilon \sqrt{dt},$$  
(4)
where $S$ is the stock price, $r$ is called the expected rate of return, and $v$ is called the stock price volatility. This special form (with $a$ and $b$ proportional to $S$) is sometimes referred to as geometric Brownian motion. In Figure 1 some simulated stocks are plotted at two time resolutions. If we decrease $dt$ more and more small-scale fluctuations appear.

Instead of (4) we can write an simpler form for the logarithm of $S$, $L = \log(S)$:

$$dL = r dt + v \epsilon \sqrt{dt}.$$  \hspace{1cm} (5)

This special form of the Ito process (with $a$ and $b$ constants) is called a Wiener process.

At this point you may be tempted to drop out of University, get some stock market data, estimate $r$ and $v$, predict the stocks using the methods described here and get fabulously rich. Before doing so, please read on.

First of all, in reality the expected rate of return $r$ is not constant, but changes depending on how well the company is doing. Nevertheless it is reasonable to assume it is constant at least for, say, the next month or so. So let’s try to estimate $r$ and $v$ from past data, then make a short term prediction for $\Delta L$, the change over a finite future time interval, and buy stocks accordingly. Inspecting (5) we see that the changes in $L$ (called $dL$) are of the form $a + b \epsilon$. The pdf of this is a normal distribution with mean $r dt$ and standard deviation $v \sqrt{dt}$. Collecting data on $dL$ from the past we can just estimate these numbers as we have done before in analysing simulation data and obtain $r$ and $v$. Now we are ready to predict the future. Let’s try to predict the change in (the logarithm of) the stock value $\Delta L$ over a future period $T$. This will involve $N = T/dt$ steps numbered $1, \ldots, N$, with corresponding changes $dL_i$. We can write

$$\Delta L = \sum_{i=1}^{N} dL_i,$$
or

\[ \frac{\Delta L}{N} = \frac{1}{N} \sum_{i=1}^{N} dL_i/N \]

and we recognize the average of \( dL \) over \( N \) samples. Invoking the central limit theorem, this means \( \Delta L/N \) obeys a normal distribution with mean \( rdt \) and std \( v\sqrt{dt}/\sqrt{N} \). Therefore \( \Delta L \) is also normally distributed with mean \( \mu = rdtN \) and std \( \sigma = v\sqrt{dt}/\sqrt{N} \). Substituting \( N = T/dt \) we obtain

\[ \mu = rT \quad \sigma = v\sqrt{T}. \]

In our example in Figure 1 we had \( r = 0.12 \) and \( v = 0.2 \). If we try to predict the logarithm of the stock increase \( \Delta L \) over 1 month we obtain

\[ \mu = 0.01 \quad \sigma = 0.058. \]

Alas, the standard deviation is much larger than the predicted increase, meaning any buying decision just a gamble. The reason is that \( \mu \) is proportional to \( T \) and \( \sigma \) to \( \sqrt{T} \). So for small \( T \), that is short term predictions, the volatility will dominate. However for larger \( T \) the volatility will become less important. The point where \( \sigma = \mu \) is at \( T = v^2/r^2 = 2.04 \) years. So we can expect to make reasonable predictions after several years. However, by that time \( r \) has probably changed already, so we seem to be out of luck!