Chapter 8: Mixing Rates for Markov Chains

8.1. Four strategies: a look ahead

In what follows, four complementary strategies will guide the agenda for this chapter, as you explore convergence behavior, looking for patterns, formulating conjectures, and constructing proofs.

1. You will look first at simple numerical examples of 2x2 matrices, whose powers you can compute by hand. The patterns here can serve as the basis for conjectures about convergence behavior.

2. The calculations with particular examples are then easy to repeat algebraically for the general 2x2 transition matrix. This algebraic analysis provides the proofs required to turn the conjectures into theorems.

3. At this point, with 2x2 matrices under control, a natural question to ask is about extensions: "Does the analysis carry over to 3x3 matrices? *k*x*k* matrices?" The answer turns out to be, "Sometimes yes, sometimes no." For matrices larger than 2x2, computing powers by hand is slow and tedious; it makes more sense to use a computer. The main goal of the computer work will be to formulate a conjecture about the properties of the convergence relationship when it parallels the 2x2 case.

4. The chapter ends with an algebraic treatment of the general *k*x*k* transition matrix **P**. If **P** is symmetric, you can base the analysis on a standard result from linear algebra:

   Theorem: If **A** is a *k*x*k* symmetric matrix, then there is an orthogonal matrix **S**,[1] whose rows are eigenvectors of **A**, and a diagonal matrix Λ whose elements are the eigenvalues of **A**, such that $\mathbf{A} = \mathbf{S}^T\Lambda\mathbf{S}$. Moreover, the eigenvalues are all real.

   This theorem leads directly to a theorem about convergence rates for symmetric transition matrices **P**. The result is surprisingly simple: When transformed to the right scale, the relationship between $\left\|\mathbf{p}^{(n)} - \boldsymbol{\pi}\right\|$ and n is linear, or at least roughly linear, and the rate parameter θ is easy to compute if you know the eigenvalues of **P**.

   Unfortunately, many transition matrices that arise from applications are not symmetric. What then? Fortunately, it is possible to extend the analysis to a much larger class of Markov chains. These chains turn out to be precisely the ones that are also Markov chains when you "run them backwards."

[1] A matrix S is orthogonal if it inverse equals its transpose: $\mathbf{S}^{-1} = \mathbf{S}^T$, i.e., $\mathbf{S}\mathbf{S}^T = \mathbf{S}^T\mathbf{S}^{-} = \mathbf{I}$.

The goal of this chapter, then, is to study the relationship between $\left\|\mathbf{p}^{(n)} - \boldsymbol{\pi}\right\|$ and n, in order to

(1) Understand, via examples and proofs, the reasons for the geometric form of convergence,
(2) determine the parameters M and θ for that form, and
(3) use (1) and (2) to find the value of n required to make sure that $\left\|\mathbf{p}^{(n)} - \boldsymbol{\pi}\right\|$ is "small enough." Here, "small enough" means that after n steps, we can regard the state where we are as having been chosen at random with all states equally likely.

Questions answered, and unanswered. Though the relationship between $\left\|\mathbf{p}^{(n)} - \boldsymbol{\pi}\right\|$ and n eventually proves to be quite simple, finding that simple relationship still leaves many practical questions unanswered. In effect, the results of this chapter let us answer the question, "How fast does $\mathbf{P}^n$ converge?" by giving a single number, one of the eigenvalues of the transition matrix, which tells the convergence rate. Once we know the rate, we'll be able to use it to answer questions like, "How big should n be to ensure that $\left\|\mathbf{p}^{(n)} - \boldsymbol{\pi}\right\| < .001$?" In other words, we'll be able to give an answer, in terms of the convergence rate, to the question, "If I'm generating random data sets using random swaps, how many steps do I need to take before I can regard the data set where the walk has stopped as a random data set, one chosen with equal probability from all possible data sets?" This is major progress, but it still leaves unanswered the question, "How does the convergence rate for a graph walk related to the structure of the graph?" That question is the topic for Chapter 9.

8.2. Empirical exploration of powers of 2x2 matrices

As you work through the following paper-and-pencil activity, keep in mind that the purpose of the arithmetic is to provide you with an opportunity to play the role of "natural scientist" doing "field work" on 2x2 matrices, observing their behavior and looking for patterns. In particular, try to determine from the patterns why the form of convergence is geometric, with $\left\|\mathbf{p}^{(n)} - \boldsymbol{\pi}\right\| \approx M\theta^n$.

Activity: powers of 2x2 matrices

Step 1. Display 8.2 shows several 2x2 matrices down the left margin, with space to write their first four powers and limits of these powers, if limits exist. Fill in as much of the table as you can.

Step 2. Describe in words the different patterns you find in the behavior of successive powers. Along with each pattern you identify, list the matrices from the table whose powers fit that pattern.

Step 3. For some of the matrices, convergence is immediate, because $\mathbf{P} = \overline{\mathbf{P}}$. (The notation $\overline{\mathbf{P}}$ means the same as $\mathbf{P}^{\infty}$, and refers to the limit of the powers of **P**, the matrix whose rows are all equal to the stationary vector π.) Pick one of the other matrices, and compute the differences ($\mathbf{P}^{(n)} - \overline{\mathbf{P}}$).

Step 4. If possible, find a formula for ($\mathbf{P}^{(n)} - \overline{\mathbf{P}}$) in terms of n, for each matrix **P**. Does your formula fit the pattern for geometric convergence?

Once you have a formula for ($\mathbf{P}^{(n)} - \overline{\mathbf{P}}$), you can use it to find $\left\|\mathbf{p}^{(n)} - \boldsymbol{\pi}\right\|$. Because the distribution $\mathbf{p}^{(n)}$ at step n typically depends on the starting distribution $\mathbf{p}^{(0)}$, the distance $\left\|\mathbf{p}^{(n)} - \boldsymbol{\pi}\right\|$ does also. For simplicity, it would be nice to get rid of the dependence on $\mathbf{p}^{(0)}$. A common way to do this is to find the worst case, that is, the choice of $\mathbf{p}^{(0)}$ for which the variation distance is largest, and give that value as an upper bound. Here is a simple example to illustrate how this works:

Example. Let $\mathbf{P} = \begin{bmatrix} 1/4 & 3/4 \\ 1/2 & 1/2 \end{bmatrix}$, $\mathbf{p}^{(0)} = (p, 1-p)$, with $0 \le p \le 1$.

Find an upper bound for $\left\|\mathbf{p}^{(1)}\right\| = \left\|\mathbf{p}^{(0)}\mathbf{P}\right\|$.

Solution. $\mathbf{p}^{(0)}\mathbf{P} = (p, 1-p)\begin{bmatrix} 1/4 & 3/4 \\ 1/2 & 1/2 \end{bmatrix} = \left(\frac{1}{4}p + \frac{1}{2}(1-p) \quad \frac{3}{4}p + \frac{1}{2}(1-p)\right)$

$$= \left(\frac{1}{2} - \frac{1}{4}p \quad \frac{1}{2} + \frac{1}{4}p\right).$$

So $\left\|\mathbf{p}^{(1)}\right\| = \left|\frac{1}{2} - \frac{1}{4}p\right| + \left|\frac{1}{2} + \frac{1}{4}p\right| \le 1$.

In fact, for this particular example, all choices of $\mathbf{p}^{(0)}$ are equally "bad," and $\left\|\mathbf{p}^{(1)}\right\| = 1$ for all of them.

Here's how to apply the same ideas to $\left\|\mathbf{p}^{(n)} - \boldsymbol{\pi}\right\|$. First, convince yourself that $\mathbf{p}^{(0)}\overline{\mathbf{P}} = \boldsymbol{\pi}$. This fact lets you write the vector difference $\left\|\mathbf{p}^{(n)} - \boldsymbol{\pi}\right\|$ in terms of the matrix difference $\mathbf{P}^{(n)} - \overline{\mathbf{P}}$ by factoring out the vector $\mathbf{p}^{(0)}$:

$$\mathbf{p}^{(n)} - \boldsymbol{\pi} = \mathbf{p}^{(0)}\mathbf{P}^n - \mathbf{p}^{(0)}\overline{\mathbf{P}} = \mathbf{p}^{(0)}(\mathbf{P}^n - \overline{\mathbf{P}}).$$

$$\Rightarrow \left\|\mathbf{p}^{(n)} - \boldsymbol{\pi}\right\| \le \max_{\mathbf{p}^{(0)}} \left\|\mathbf{p}^{(0)}(\mathbf{P}^n - \overline{\mathbf{P}})\right\|.$$

With this as background, you now have what you need to complete the activity, by finding an upper bound on the variation distance.

Activity, concluded

Step 5. Use the same matrix **P** that you worked with in (3) and (4) of the activity. Let $\mathbf{p}^{(0)}$ = (p $1\text{-}p$) and use the "worst case" approach to find an upper bound for $\left\|\mathbf{p}^{(n)} - \boldsymbol{\pi}\right\|$. Your bound should be of the form $\left\|\mathbf{p}^{(n)} - \boldsymbol{\pi}\right\| \le M\theta^n$.

Drill: For each of the following matrices **P** and starting vectors $\mathbf{p}^{(0)}$, (a) find the values of the parameters M and θ that characterize the convergence rate. Then (b) find how many steps n are needed to make $\left\|\mathbf{p}^{(n)} - \boldsymbol{\pi}\right\| \le .01$.

1. $\mathbf{P} = \begin{bmatrix} 3/4 & 1/4 \\ 1/2 & 1/2 \end{bmatrix}, \mathbf{p}^{(0)} = \begin{pmatrix} 1 & 0 \end{pmatrix}$ 2. $\mathbf{P} = \begin{bmatrix} 1/4 & 3/4 \\ 1/2 & 1/2 \end{bmatrix}, \mathbf{p}^{(0)} = \begin{pmatrix} 1 & 0 \end{pmatrix}$

3. $\mathbf{P} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \mathbf{p}^{(0)} = \begin{pmatrix} 1 & 0 \end{pmatrix}$ 4. $\mathbf{P} = \begin{bmatrix} 1/3 & 2/3 \\ 1/3 & 2/3 \end{bmatrix}, \mathbf{p}^{(0)} = \begin{pmatrix} 1 & 0 \end{pmatrix}$

5. $\mathbf{P} = \begin{bmatrix} 2/3 & 1/3 \\ 1/3 & 2/3 \end{bmatrix}, \mathbf{p}^{(0)} = \begin{pmatrix} 1 & 0 \end{pmatrix}$ 6. $\mathbf{P} = \begin{bmatrix} 1/4 & 3/4 \\ 3/4 & 1/4 \end{bmatrix}, \mathbf{p}^{(0)} = \begin{pmatrix} 1 & 0 \end{pmatrix}$

Exercise. Power law and geometric convergence compared.

7. Graph n^θ versus n with θ = -1/2. This (power law) pattern is typical of the way $\hat{p}$ converges to p. Now graph θ^n versus n with θ = ½. This is the pattern for geometric convergence. The formulas look similar; do the graphs?

	$\mathbf{P}$	$\mathbf{P}^2$	$\mathbf{P}^3$	$\mathbf{P}^4$	$\mathbf{P}^5$	$\mathbf{P}^6$	…	$\mathbf{P}^\infty$
$\mathbf{I} =$	$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$							
$\mathbf{R} =$	$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$							
$\mathbf{P}_1 =$	$\begin{bmatrix} 1/3 & 2/3 \\ 1/3 & 2/3 \end{bmatrix}$							
$\mathbf{P}_2 =$	$\begin{bmatrix} 3/4 & 1/4 \\ 3/4 & 1/4 \end{bmatrix}$							
$\mathbf{P}_3 =$	$\begin{bmatrix} 2/3 & 1/3 \\ 1/3 & 2/3 \end{bmatrix}$							
$\mathbf{P}_4 =$	$\begin{bmatrix} 1/3 & 2/3 \\ 2/3 & 1/3 \end{bmatrix}$				$\begin{bmatrix} 41/81 & 40/81 \\ 40/81 & 41/81 \end{bmatrix}$			
$\mathbf{P}_5 =$	$\begin{bmatrix} 3/4 & 1/4 \\ 1/4 & 3/4 \end{bmatrix}$			$\begin{bmatrix} 17/32 & 15/32 \\ 15/32 & 17/32 \end{bmatrix}$				
$\mathbf{P}_6 =$	$\begin{bmatrix} 2/4 & 2/4 \\ 3/4 & 1/4 \end{bmatrix}$	$\begin{bmatrix} 10/16 & 6/16 \\ 9/16 & 7/16 \end{bmatrix}$	$\begin{bmatrix} 38/64 & 26/64 \\ 39/64 & 25/64 \end{bmatrix}$					

Display 8.2 Successive powers of eight 2x2 transition matrices

8.3. Algebraic analysis of the general 2x2 transition matrix

Working with concrete examples is often a good way to begin to learn new mathematics, or begin an investigation, because concrete examples are easier to grasp when you are new to an area. Examples have limitations, though. They can mislead you if they are not typical of the general case, and sometimes the simplifications that seem natural for numerical examples actually hide patterns that you can see more easily using algebraic notation. So even if you start with examples, it is important to move on to a general analysis, as soon as you feel prepared. The steps that follow are designed to guide you through an algebraic analysis of the convergence behavior of two-state Markov chains. If you find yourself stuck at any point, try to figure out how to carry out the step on a numerical example first. You can choose from the matrices in Display 8.2, or invent one of your own.

Guided derivation

Let $\mathbf{P} = \begin{bmatrix} 1-\alpha & \alpha \\ \beta & 1-\beta \end{bmatrix}$.

8. Show that $\lambda_1 = 1$ is an eigenvalue of $\mathbf{P}$. Find an associated eigenvector $\mathbf{e}_1$, and rescale to make it a probability vector. (Solve $\mathbf{xP} = \mathbf{x}$ for $\mathbf{x}$, where $\mathbf{x} = (x, y)$ with $x + y = 1$.)

9. Find the other eigenvalue λ_2: solve $det(\mathbf{P}-\lambda\mathbf{I}) = 0$.

10. Find an associated eigenvector $\mathbf{e}_2$: solve $\mathbf{xP} = \lambda_2\mathbf{x}$ for $\mathbf{x}$.

11. Let $\overline{\mathbf{P}}$ be the matrix whose rows are all equal to the stationary vector π. Write $\overline{\mathbf{P}}$. Then find $\mathbf{Q} = \frac{1}{\lambda_2}(\mathbf{P} - \overline{\mathbf{P}})$. Show that $\mathbf{P} = \overline{\mathbf{P}} + \lambda_2\mathbf{Q}$.

12. Verify that $\overline{\mathbf{P}}^2 = \overline{\mathbf{P}}$, $\overline{\mathbf{P}}\mathbf{Q} = \mathbf{Q}\overline{\mathbf{P}} = \mathbf{0}$, $\mathbf{Q}^2 = \lambda_2\mathbf{Q}$.

13. Find a formula for $\mathbf{P}^2$ by simplifying $(\overline{\mathbf{P}} + \lambda_2\mathbf{Q})^2$.

14. Extend (13) to get a formula for $\mathbf{P}^n$, and complete the following statement:

> Prop. If $\mathbf{P}$ is a 2x2 transition matrix, and $\overline{\mathbf{P}}$ is the matrix with rows equal to the stationary vector for $\mathbf{P}$, then the n^{th} power of $\mathbf{P}$ is given by
>
> $\mathbf{P}^n =$ ________________
>
> where λ_2 is the second largest eigenvalue of $\mathbf{P}$, and $\mathbf{Q} = \frac{1}{\lambda_2}(\mathbf{P} - \overline{\mathbf{P}})$.

15. Use the worst case approach, as in Step 5 of the activity in Section 8.2, to find an upper bound for the variation distance. Your bound should be of the form $\left\|\mathbf{p}^{(n)} - \boldsymbol{\pi}\right\| \leq M\theta^n$. (Some of the algebra for finding M gets a bit messy, but it does eventually simplify.)

Drill

16. Pick some of the 2x2 matrices from the activity in Section 8.2. Apply your formula from (14) above, and compare with the numerical results you got in the activity. (They should agree, of course.)

8.4. Empirical Investigation: Convergence rates for 3x3 matrices

For 2x2 matrices, you should have found that the total variation distance from stationarity, as a function of n = # steps, converges to 0 at a geometric rate:

$$\left\|\mathbf{p}^{(\mathbf{n})} - \boldsymbol{\pi}\right\| = (const)\left|\lambda_2\right|^n,$$

where $\mathbf{p}^{(n)} = \mathbf{p}^{(0)}\mathbf{P}^n$ is the probability distribution after n steps, and λ_2 is the second largest eigenvalue of **P**. The point of this investigation is to determine the extent to which the same is true of 3x3 transition matrices. Displays 8.3 and 8.4 show S-plus code for (1) computing the variation distance $\left\|\mathbf{p}^{(n)} - \boldsymbol{\pi}\right\|$, and (2) plotting $\left\|\mathbf{p}^{(n)} - \boldsymbol{\pi}\right\|$ versus n in various scales.

Investigation.

17. Create/invent several 3x3 transition matrices **P**, and for each one, use the plots to find a formula that describes the distance as a function of the number of steps. Also find the eigenvalues $\lambda_1 \geq \lambda_2 \geq \lambda_3$ of **P**, and, where possible, relate the rate parameter θ in your formula to the eigenvalues.[2]

[2] Don't put any effort into finding the constant M. Such laziness can be justified two ways. First off, θ is much more important than M, because θ governs the effect of n on the variation distance, whereas M "just sits there." Second, because of the way variation distance is defined, its value can never be greater than 1. This gives an automatic albeit lazy upper bound of $M \leq 1$. Moreover, for kxk matrices, it is possible to find a matrix **P** for which $\left\|\mathbf{p}^{(0)}\left(\mathbf{P} - \overline{\mathbf{P}}\right)\right\| \geq \left(1 - \frac{1}{k}\right) - \varepsilon$ for any ε, so for large matrices, $M = 1$ is not far from the best you can do. Taking $M = 1$ means you can focus exclusively on the convergence rate θ, and its relationship to the eigenvalues of **P**.

```
######################################################################
#                                                                    #
#                MARKOV CHAINS I:  BASIC COMPUTATIONS                #
#                                                                    #
######################################################################

###  ENTER A TRANSITION MATRIX P BY ROWS

row1 <- c(1,0,0)
row2 <- c(0,1,0)
row3 <- c(0,0,1)
P <- rbind(row1,row2,row3)
P
p.0 <- c(0,1,0)

###  COMPUTE THE nTH POWER OF P

power <- function(P,n) {      # returns the nth power of a square matrix P
   Q <- P
   for (i in 2:n) {
      Q <- (Q %*% P)
      }
   return(Q)
}

###  COMPUTE THE EQULIBRIUM DISTRIBUTION FOR P

EqDist <- function(P){        # finds equilibrium distribution of P
   n <- dim(P)[1]             # n = number of rows of P
   b <- rep(0,n)              # b = vector for right hand side
   for (i in 1:n) {           # transpose of the coefficient matrix:
      P[i,i] <- P[i,i] - 1    # replace P by P-I
      P[i,n] <- 1             # put 1s in last column
   }
   b[n] <- 1                  # put 1 in last element of rhs
   return(solve(t(P),b))      # solve to get normalized eigenvector
}                             # If there’s more than one solution,
                              #  S-plus sends an error message
                              #  “apparently singular matrix”

#
EqDist(P)

###  EIGENVALUES OF P

eigen(P)$values
```

Display 8.3 S-plus code for basic computations for Markov chains

```
#####################################################################
#                                                                   #
#        MARKOV CHAINS II:  CONVERGENCE OF n-STEP PROBABILITIES      #
#                                                                   #
#####################################################################

### The following functions compute the total variation distance
#   between the stationary distribution for P and the probability
#   distribution after n steps, then use "apply" to compute distances
#   for a vector of values for n, and plot the distance versus number
#   of steps using four different choices of scale.
#
#   The functions EqDist and power are called by the ones listed here.

### COMPUTE THE TOTAL VARIATION NORM OF A VECTOR

VarNorm <- function(p.vector){       # returns the total variation norm
   return(.5*sum(abs(p.vector)))     # of a vector
}

### COMPUTE VARIATION DISTANCE FROM THE STATIONARY DISTRIBUTION
#
# Given a vector p.start of starting probabilties, a transition matrix P,
# and a power n (= number of steps), the function DistFromEq.prob returns
# the variation distance between the probability distribution at step n
# and the equilibrium distribution.
#

DistFromEq.prob <- function(p.start,P,n){
   return(VarNorm(EqDist(P)-p.start %*% power(P,n)))
}

###  COMPUTE A VECTOR dist.values OF DISTANCES FOR A VECTOR n.values OF STEPS n

n.values <- c(1, 2, 5, 10, 25, 50, 100, 500)      # values of n = no. of steps
n.values <- matrix(n.values,1,length(n.values))   # row vector as matrix
dist.values <- apply(n.values,2,DistFromEq.prob,p.start=p.0,P=P)
dist.values

###  PLOT DISTANCE VERSUS NUMBER OF STEPS, IN VARIOUS SCALES

Plot <- function(n.vals,dist.vals){
   n.vals <- n.vals[dist.vals > 5e-016]       # Truncate:  throw away values
   dist.vals <- dist.vals[dist.vals > 5e-016] # below .5x10^-16 before plotting
   par(mfrow=c(2,2))                          # Four Plots on one page
                                              #  log = “”   linear
                                              #  log = “x”  logarithmic
                                              #  log = “y”  geometric
                                              #  log = “xy” power law
   plot(n.vals,dist.vals,log="",type="b",sub="linear")         # Linear
   plot(n.vals,dist.vals,log="x",type="b",sub="logarithmic")   # Logarithmic
   plot(n.vals,dist.vals,log="y",type="b",sub="geometric")     # Geometric
   plot(n.vals,dist.vals,log="xy",type="b",sub="power law")    # Power law
}

Plot(n.values,dist.values)
```

Display 8.4 S-plus code for plots to reveal the form of convergence

8.5. Algebraic analysis of convergence rates for *k*x*k* matrices

Preview

In 8.3 you saw that it was possible to write the n^{th} power of a 2x2 matrix **P** in a simple form:

$$\mathbf{P}^n = \overline{\mathbf{P}} + \lambda_2^n \mathbf{Q}.$$

This led to a proof of the geometric convergence for 2-state chains:

$$\left\|\mathbf{p}^{(n)} - \pi\right\| \le M\theta^n, \text{ with } \theta = |\lambda_2|.$$

Most recently, in 8.4, you should have found that for some (but not all) 3x3 matrices, convergence is also geometric, with rate depending on the eigenvalues of **P**. In this final section of the chapter, we'll use results from linear algebra to analyze two categories of transition matrices for which convergence is geometric.

Results from linear algebra.[3]

The essentials boil down to just three: a definition, a proposition, and a theorem.

8.5.1 **Def**. Two *k*x*k* matrices **A** and **B** are **similar** (notation: **A** ~ **B**) if and only if there is an invertible matrix **C** such that $\mathbf{CAC}^{-1} = \mathbf{B}$.

8.5.2 **Prop.** If (1) **A** and **B** are similar, and (2) λ is an eigenvalue of **B** with associated left eigenvector **x**, then λ is also an eigenvalue of **A** with associated left eigenvector **xC**.

Proof (Exercise 18):

(a) Write the equations that correspond to (1) and (2).
(b) Substitute the equation for **B**, from (1), into the equation for (2).
(c) Multiply both sides by **C**.

8.5.3 **Th^m^**. (Diagonalization Theorem for Symmetric Matrices) If **A** is a *k*x*k* symmetric matrix, then

(a) all k eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_k$ of **A** are real, and
(b) $\mathbf{S}^T\mathbf{AS} = \text{diag}\{\lambda_1, \lambda_2, \ldots, \lambda_k\}$, where the rows of **S** are the left eigenvectors of **A**, and $\mathbf{S}^T\mathbf{S} = \mathbf{SS}^T = \mathbf{I}$.

Proof: See any standard introduction to linear algebra.

[3] The results from linear algebra are presented here in terms of matrices. In many linear algebra courses, parallel results are presented in terms of linear transformations on vector spaces. According to that view, a matrix represents a linear transformation with respect to some set of basis vectors. (For example, two matrices are similar if the represent the same linear transformation with respect to different bases.) That view is richer and more profound than the one presented here, but I have decided to use the simpler version because that is all we need at this point. However, if you know the deeper version, I encourage you to make a mental translation as you read the statements about matrices.

Drill

19. Find all the matrices that are similar to I = $\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}$. (Hint: Use 8.5.2.)

20. Find all the matrices that are similar to **Z** = $\begin{bmatrix}0 & 0\\ 0 & 0\end{bmatrix}$.

21. Which of the following matrices are similar to $\begin{bmatrix}1 & 0\\ 0 & 0\end{bmatrix}$?

(a) $\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}$ (b) $\begin{bmatrix}1 & 0\\ 0 & 2\end{bmatrix}$ (c) $\begin{bmatrix}1/2 & 1/2\\ 1/2 & 1/2\end{bmatrix}$ (d) $\begin{bmatrix}3/4 & 1/4\\ 3/4 & 1/4\end{bmatrix}$ (e) $\begin{bmatrix}1/3 & 2/3\\ 1/3 & 2/3\end{bmatrix}$

(f) $\begin{bmatrix}1/3 & 2/3\\ 2/3 & 1/3\end{bmatrix}$ (g) $\begin{bmatrix}0 & 0\\ 0 & 0\end{bmatrix}$ (h) $\begin{bmatrix}0 & 1\\ 1 & 0\end{bmatrix}$

22. True or false, and explain[4]: If **A** ~ **B**, then *det*(**A**) = *det*(**B**).

23. True or false, and explain: If **A** ~ **B**, then $a_{11} + a_{22} + \ldots + a_{kk} = b_{11} + b_{22} + \ldots + b_{kk}$.

24-25. Transition matrices **P**, together with their left eigenvectors, are given below. For each matrix **P**, (a) form the matrix **S** whose rows are the eigenvectors of **P**, (b) verify that $\mathbf{S}^T\mathbf{S} = \mathbf{S}\mathbf{S}^T = \mathbf{I}$, (c) compute $\mathbf{S}^T\mathbf{P}\mathbf{S}$, and (d) verify that the elements of the resulting diagonal matrix are eigenvalues of **P**.

24. $\mathbf{P_3} = \begin{bmatrix}3/4 & 1/4\\ 1/4 & 3/4\end{bmatrix}$, $\mathbf{e_1} = \frac{1}{\sqrt{2}}(1,\ 1)$, $\mathbf{e_2} = \frac{1}{\sqrt{2}}(1,\ -1)$.

25. $\mathbf{P}_4 = \begin{bmatrix}1/3 & 2/3\\ 2/3 & 1/3\end{bmatrix}$, $\mathbf{e_1} = \frac{1}{\sqrt{2}}(1,\ 1)$, $\mathbf{e_2} = \frac{1}{\sqrt{2}}(1,\ -1)$.

26. From linear algebra, you are familiar with dot products, which have the form (row vector)(column vector) = scalar. The drill exercises below are intended to introduce you to the form (column vector)(row vector) = (matrix). Write the following matrices:

(a) $\begin{bmatrix}1/2 & 1/2\end{bmatrix}\begin{bmatrix}1\\ 1\end{bmatrix}$ (b) $\begin{bmatrix}1\\ 1\end{bmatrix}\begin{bmatrix}1/2 & 1/2\end{bmatrix}$ (c) $\begin{bmatrix}1\\ 1\end{bmatrix}\begin{bmatrix}1/4 & 3/4\end{bmatrix}$ (d) $\begin{bmatrix}1/4\\ 3/4\end{bmatrix}\begin{bmatrix}1 & 1\end{bmatrix}$

(e) $\begin{bmatrix}1/\sqrt{2}\\ 1/\sqrt{2}\end{bmatrix}\begin{bmatrix}\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}}\end{bmatrix}$ (f) $\mathbf{1}^T\mathbf{p}$, where **1** = (1, 1, 1) and **p** = (p_1, p_2, p_3).

(g) $\mathbf{v}^T\mathbf{v}$, where $\mathbf{v} = \frac{1}{\sqrt{3}}(1, 1, 1)$. (h) $\mathbf{v}\mathbf{v}^T$, with **v** as in (g).

---

[4] This question and the next use results from linear algebra about the relationship between the determinant (22) and trace (23) of a matrix and its eigenvalues. If you aren't familiar with those results, answer for the case of 2x2 matrices only.

Symmetric transition matrices: spectral decomposition and geometric convergence. The diagonalization theorem for symmetric matrices has an alternate version, called a spectral theorem[5], that is more useful for analyzing convergence. Here, and throughout the remainder of the book, we will use the convention that eigenvalues are ordered so that $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_k$.

8.5.4 Th^m^. (Spectral Decomposition for Symmetric Matrices)
If **A** is a kxk symmetric matrix, then

(a) $\mathbf{A} = \lambda_1\left[\mathbf{e}_1^T\mathbf{e}_1\right] + \lambda_2\left[\mathbf{e}_2^T\mathbf{e}_2\right] + \ldots + \lambda_k\left[\mathbf{e}_k^T\mathbf{e}_k\right] = \lambda_1\mathbf{E}_1 + \lambda_2\mathbf{E}_2 + \ldots + \lambda_k\mathbf{E}_k$, where the $\mathbf{e}_i$ are the left eigenvectors of **A**, with associated eigenvalues λ_i, and $\mathbf{E}_i = \left[\mathbf{e}_i^T\mathbf{e}_i\right]$ is a kxk matrix.

(b) $\mathbf{E}_i\mathbf{E}_j = \begin{cases} \mathbf{E}_i & \text{if } i = j \\ \mathbf{0} & \text{if } i \neq j \end{cases}$

(c) $\mathbf{A}^n = \lambda_1^n\mathbf{E}_1 + \lambda_2^n\mathbf{E}_2 + \ldots + \lambda_k^n\mathbf{E}_k$.

Proof: See Appendix II[6]

Drill. Exercises 27-35 below take you through the creation of two examples, first a 2x2 example (27 - 30), then a 3x3 (31-35). You should do these before going on.

Exercises 27 – 30 create a 2x2 example.

27. Consider the symmetric matrix $\mathbf{P} = \begin{bmatrix} 1-\alpha & \alpha \\ \alpha & 1-\alpha \end{bmatrix}$. Because the columns add to 1, the stationary distribution is uniform. Let $\mathbf{e}_1 = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \end{pmatrix}$, and check that $[\mathbf{e}_1^T\mathbf{e}_1] = \begin{bmatrix} 1/2 & 1/2 \\ 1/2 & 1/2 \end{bmatrix} = \mathbf{E}_1$ is equal to the limiting matrix $\overline{\mathbf{P}} = \begin{bmatrix} \pi \\ \pi \end{bmatrix}$.

28. Now let $\mathbf{e}_2 = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & -1 \end{pmatrix}$. Check that $\mathbf{e}_2\mathbf{P} = (1-2\alpha)\mathbf{e}_2$, so that $\mathbf{e}_2$ is an eigenvector of **P** with eigenvalue $\lambda_2 = (1\text{-}2\alpha)$. Define $\mathbf{E}_2 = [\mathbf{e}_2^T\mathbf{e}_2]$. Compute $\mathbf{E}_2$. Then verify that $\mathbf{E}_1^2 = \mathbf{E}_1$, $\mathbf{E}_2^2 = \mathbf{E}_2$, and $\mathbf{E}_1\mathbf{E}_2 = \mathbf{0}$.

29. Compute $\lambda_1\mathbf{E}_1 + \lambda_2\mathbf{E}_2$ and verify that this matrix equals **P**, with $\lambda_1\mathbf{E}_1 = \overline{\mathbf{P}}$ (from 27) and thus $(\mathbf{P} - \overline{\mathbf{P}}) = \lambda_2\mathbf{E}_2$.

---

[5] The name comes from the fact that the decomposition using eigenvalues was used to study waves, including light waves. The set of eigenvalues of a matrix is often called its **spectrum**.

[6] Appendix II is not yet written. Many linear algebra books contain a proof.

30. Now compute $\mathbf{P}^2$ two ways. (a) Multiply $\mathbf{P} = \begin{bmatrix} 1-\alpha & \alpha \\ \alpha & 1-\alpha \end{bmatrix}$ by itself. (b) Use the spectral decomposition: compute $\mathbf{P}^2 = \overline{\mathbf{P}} + \lambda_2^2 \mathbf{E_2}$. Verify that this gives the same answer as (a).

Exercises 31 – 35 create a 3x3 example.

31. Let $\mathbf{P} = \begin{bmatrix} 0 & .5 & .5 \\ .5 & 0 & .5 \\ .5 & .5 & 0 \end{bmatrix}$. Verify that $\mathbf{P}$ has eigenvalues and eigenvectors

$$\lambda_1 = 1,\ \mathbf{e_1} = \tfrac{1}{\sqrt{3}}\begin{bmatrix} 1 & 1 & 1 \end{bmatrix},\quad \lambda_2 = -1/2,\ \mathbf{e_2} = \tfrac{1}{\sqrt{2}}\begin{bmatrix} 1 & 0 & -1 \end{bmatrix},\quad \lambda_3 = -1/2,\ \mathbf{e_3} = \tfrac{1}{\sqrt{6}}\begin{bmatrix} 1 & -2 & 1 \end{bmatrix}.$$

32. Compute the matrices $\mathbf{E_1}$, $\mathbf{E_2}$, and $\mathbf{E_3}$, where $\mathbf{E_i} = \mathbf{e_i}^T \mathbf{e_i}$.

33. Compute $\lambda_1 \mathbf{E_1} + \lambda_2 \mathbf{E_2} + \lambda_3 \mathbf{E_3}$, and verify that the sum equals $\mathbf{P}$.

34. Verify that $\mathbf{E_i}\mathbf{E_j} = \mathbf{E_i}$ if $i = j$, and $\mathbf{E_i}\mathbf{E_j} = \mathbf{0}$ if $i \neq j$.

35. Use the spectral decomposition to write $(\mathbf{P}^n - \overline{\mathbf{P}}) = \lambda_2^n \mathbf{E_2} + \lambda_3^n \mathbf{E_3}$. Simplify this to the form $\left(-\tfrac{1}{2}\right)^n \mathbf{Q}$, where $\mathbf{Q} = \mathbf{E_2} + \mathbf{E_3}$.

Additional exercises:

36. Carry out the spectral decomposition for the graph walk on the complete 3-point graph, i.e., a triangle.

37. Carry out the spectral decomposition on the complete 4-point graph.

8.5.5 Cor. (Geometric Convergence for Symmetric Matrices) If $\mathbf{P}$ is a kxk symmetric transition matrix with limiting matrix $\overline{\mathbf{P}}$, then
(a) $\mathbf{P}^n - \overline{\mathbf{P}} = \lambda_2^n \mathbf{E_2} + \ldots + \lambda_k^n \mathbf{E}_k$, and
(b) $\left\|\mathbf{p}^{(\mathbf{n})} - \boldsymbol{\pi}\right\| \leq M\theta^n$, where $M \leq 1$ and $\theta = \max\{|\lambda_2|, |\lambda_k|\}$.

Proof: See Appendix II.[7]

Drill.

38. For the 2-state Markov chain of Exercises 27 - 30, find M and θ. What is the smallest number of steps n that will guarantee $\mathbf{p}^{(n)}$ is within .01 of the stationary distribution?

---

[7] Not yet written.

39. Suppose the starting vector for the three-state chain of exercises 31 - 35 is $\mathbf{p}^{(o)} = (p,q,r)$. Show that $\left\|\mathbf{p}^{(n)} - \pi\right\| = \left(\frac{1}{2}\right)^n \left\|\left(p \quad q \quad r\right)\right\| \le \left(\frac{1}{2}\right)^n$.

40. What is the smallest number of steps n that will guarantee that $\mathbf{p}^{(n)}$ is within .01 of the stationary distribution?

Reversible transition matrices: spectral decomposition and geometric convergence. What if **P** is not symmetric? After all, even for graph walks, symmetry is a very special case, because the only symmetric walks are on regular graphs – those for which every vertex has the same number of neighbors. Fortunately, it is possible to extend the last two results, spectral decomposition and geometric convergence, to a much larger class of matrices. It isn't necessary that **P** itself be symmetric, as long as **P** is *similar* to some symmetric matrix. Then there is an invertible matrix **C** such that $\mathbf{CPC}^{-1}$ is symmetric, which means that this new matrix $\mathbf{R} = \mathbf{CPC}^{-1}$ has a spectral decomposition: $\mathbf{CPC}^{-1} = \lambda_1\mathbf{E}_1 + \lambda_2\mathbf{E}_2 + \ldots + \lambda_k\mathbf{E}_k$, and so $\mathbf{P} = \mathbf{C}^{-1}(\lambda_1\mathbf{E}_1 + \lambda_2\mathbf{E}_2 + \ldots + \lambda_k\mathbf{E}_k)\mathbf{C}$. This leads to a proof of geometric convergence for **P**.

The key, then, is to find a way to characterize the matrices that are similar to symmetric matrices. It turns out that many Markov chains (though not all) satisfy a condition called detailed balance or reversibility, that guarantees similarity to a symmetric matrix, and geometric convergence to stationarity.

8.5.6 **Def**. Let $\mathbf{P} = \{p_{ij}\}$ be a transition matrix with stationary vector $\pi = \{\pi_i\}$. Then **P** is **reversible** (equivalently, **P** satisfies **detailed balance**) if and only if $\pi_i p_{ij} = \pi_j p_{ji}$ for all pairs i,j.

Drill

41. The random walk on the three point linear graph 1—2—3 is not symmetric. Write its transition matrix **P**, find the stationary distribution π, and show that **P** satisfies detailed balance.

8.5.7 **Prop.** If transition matrix **P** has limiting distribution π, then let $\mathbf{D} = diag\{\pi_i\}$, $\mathbf{D}^{1/2} = diag\{\sqrt{\pi_i}\}$, and $\mathbf{D}^{-1/2} = diag\{\frac{1}{\sqrt{\pi_i}}\}$. Then $\mathbf{D}^{1/2}\mathbf{P}\mathbf{D}^{-1/2}$ is symmetric.

Proof (Exercise 42**).**

Drill

43. For the random walk in Exercise 41, (a) compute **D**, $\mathbf{D}^{1/2}$, and $\mathbf{D}^{-1/2}$, (b) verify that $\mathbf{D}^{1/2}\mathbf{D}^{-1/2} = \mathbf{I}$, (c) compute $\mathbf{D}^{1/2}\mathbf{P}\mathbf{D}^{-1/2}$ and verify that this matrix is symmetric.

8.5.8 **Th^{m}**. (Spectral Decomposition for Reversible Transition Matrices)
Let **P** be a kxk reversible transition matrix. Then

(a) $\mathbf{P} = \lambda_1 \mathbf{F_1} + \lambda_2 \mathbf{F_2} + \ldots + \lambda_k \mathbf{F}_k$, where

(b) $\mathbf{F_i F_j} = \begin{cases} \mathbf{F_i} & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases}$ and so

(c) $\mathbf{P}^n = \lambda_1^n \mathbf{F_1} + \lambda_2^n \mathbf{F_2} + \ldots + \lambda_k^n \mathbf{F}_k$.

Proof: See Appendix II.[8]

Drill.

44. For the non-symmetric random walk on the 3-point linear graph 1—2—3 form the matrix **R**, where $r_{ij} = \frac{\sqrt{\pi_i}}{\sqrt{\pi_j}} p_{ij}$. Check that **R** is symmetric, and that $\mathbf{R} = \mathbf{D}^{1/2}\mathbf{P}\mathbf{D}^{-1/2}$.

45. Check that $\lambda_1 = 1, \lambda_2 = 0$, and $\lambda_3 = -1$ are eigenvalues of both **P** and **R**.

46. Check that **R** has eigenvectors $\mathbf{e_1} = \frac{1}{2}\begin{pmatrix} 1 & \sqrt{2} & 1 \end{pmatrix}$, $\mathbf{e_2} = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 0 & -1 \end{pmatrix}$, and $\mathbf{e_3} = \frac{1}{2}\begin{pmatrix} 1 & -\sqrt{2} & 1 \end{pmatrix}$, and that $\mathbf{e_i e_j}^T = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases}$.

47. Give the spectral decomposition for **R**: $\mathbf{R} = \lambda_1 \mathbf{E_1} + \lambda_2 \mathbf{E_2} + \lambda_3 \mathbf{E_3}$.

48. Use the fact that $\mathbf{R} = \mathbf{D}^{1/2}\mathbf{P}\mathbf{D}^{-1/2}$ to go from your decomposition for **R** in (47) to the spectral decomposition for **P**:

$$\mathbf{R} = \mathbf{D}^{1/2}\mathbf{P}\mathbf{D}^{-1/2} \Rightarrow \mathbf{P} = \mathbf{D}^{-1/2}\mathbf{R}\mathbf{D}^{1/2} = \mathbf{D}^{-1/2}(\lambda_1 \mathbf{E_1} + \lambda_2 \mathbf{E_2} + \lambda_3 \mathbf{E_3}) = \lambda_1 \mathbf{F_1} + \lambda_2 \mathbf{F_2} + \lambda_3 \mathbf{F_3}$$

Find $\mathbf{F_1}$, $\mathbf{F_2}$, $\mathbf{F_3}$ and verify that $\mathbf{F_i F_j} = \begin{cases} \mathbf{F_i} & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases}$.

49. Use (48) to write a formula for $(\mathbf{P}^n - \overline{\mathbf{P}})$, and then explain how the formula tells the behavior of the graph walk.

Additional exercise:

50. Find the spectral decomposition for $\mathbf{P} = \begin{bmatrix} 1/2 & 1/2 \\ 1/4 & 3/4 \end{bmatrix}$ using the approach of (44)-(49).

8.5.9 Cor. (Geometric Convergence for Reversible Transition Matrices) If **P** is a kxk reversible transition matrix with limiting matrix $\overline{\mathbf{P}}$, then

[8] Not yet written.

(a) $\mathbf{P}^n - \overline{\mathbf{P}} = \lambda_2^n \mathbf{F}_2 + \ldots + \lambda_k^n \mathbf{F}_k$, and

(c) $\left\| \mathbf{p}^{(\mathbf{n})} - \boldsymbol{\pi} \right\| \le M\theta^n$, where $M \le 1$ and $\theta = \max\{|\lambda_2|, |\lambda_k|\}$.

Drill:

51. For the walk in (41), find the number of steps required to ensure that the distribution is within .01 of the stationary distribution.

Investigations:

52. Investigate detailed balance (reversibility) for graph walks: For which graphs is the walk reversible, i.e., for which graphs does the matrix **P** satisfy detailed balance? For which graphs is the walk not reversible?

53. Find transition matrices **P** which cannot be diagonalized.

Review: Analyzing Swap Walks on a Co-occurrence Matrix

As a way to review and consolidate your work so far, think back to the Galapagos finches. An ecologist has observed that the 13 species of finches form 10 "checkerboards" (species pairs with no islands occupied by both), and offers that pattern as evidence that the finches have sorted themselves on islands in a way that avoids competition between species. Another ecologist, the Chance Skeptic, remains unconvinced: "Before I accept your interpretation, I want to know how easy it would be to get 10 checkerboards if the finches had sorted themselves onto islands purely by chance." The second ecologist wants to know the *p*-value:

$$p = \Pr\{10 \text{ or more checkerboards}\},$$

assuming, as a null model, that all co-occurrence matrices with the same margins as the finch data are equally likely. Because the matrices are equally likely, the *p*-value is just the proportion of matrices with 10 or more checkerboards.

In principle, the *p*-value can be computed exactly by enumeration: list all the matrices, count the number with 10 or more checkerboards, and divide by the total number of matrices. In practice, the number of matrices is so very large that enumeration is impossible. Instead, ecologists use some variant of the three step algorithm:

Step 1. Generate random co-occurrence matrices by choosing 2x2 sub-matrices of C at random, and swapping their 0s and 1s whenever the swap leaves marginal total unchanged.
Step 2. Compare random and observed matrices. Record a Yes if the random matrix has at least as many checkerboards as the observed value.
Step 3. Estimate the *p*-value as $\hat{p} = \#Yes / \#Matrices$.

Because our goal is to study the method for computing the *p*-value, we'll consider a more manageable data set, with only three species and four islands:

$$\mathbf{C} = \begin{bmatrix} 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

A. Preliminaries.

54. How many checkerboards are there? Call this number the observed value.

55. Find the marginal totals for **C**. Make a rough guess about the number of matrices with the same marginal totals: Would you expect the number to be closest to 5, 10, 25, 50 or 100?

56. Make a rough guess for the *p*-value: What proportion of the matrices do you think will have at least as many checkerboards as the observed value?

B. Computing the *p*-value by exact enumeration.

Note that to do this, you can follow the structure of the *p*-value algorithm:

Step 1. Generate a list all the matrices with the same margins as **C**.[9] According to the null model, these are equally likely.

Step 2. Compare each matrix with the observed matrix: record a Yes if the candidate matrix has at least as many checkerboards as the observed value.

Step 3. Compute the true *p*-value: $p = \#Yes/\#equally\ likely\ matrices$.

C. Analyzing the method of random swaps, part 1: limiting distribution.

57. Adjacency matrix. Regard each of the matrices in B1 as the vertex of a graph. Two matrices are adjacent if you can get from one to the other by swapping 0s and 1s in a 2x2 sub-matrix. Use these facts to find the adjacency matrix for the graph.[10]

58. Graph. Represent the adjacency matrix as a graph. (It may take a couple of tries to get a graph that looks nice. As a rule, you get a simpler looking graph if you put vertices with more neighbors toward the center, vertices with fewer neighbors toward the outside.) Are you surprised by the result here? Can you offer any reasons why the graph should look the way it does?

59. Limiting distribution. Find the limiting distribution for the graph walk. Is it uniform?

60. Limiting value for the estimate. If you use the graph walk to generate random matrices, and estimate the *p*-value as $\hat{p} = \#Yes/NReps$, what number will the estimate converge to?

61. Metropolize: First write the transition matrix **P** for the graph walk in (2) Then write the transition matrix $\tilde{\mathbf{P}}$ of the Metropolized graph walk. What will $\hat{p}$ converge to for this Metropolized walk?

D. Analyzing the method of random swaps, part 2: the mixing rate.

62. Make a prediction: Which Markov chain do you think will mix faster, the original swap walk, or the Metropolized version?

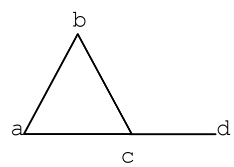
63. Spectral decomposition and mixing rate. Use the computer to obtain the spectral decompositions of **P** and $\tilde{\mathbf{P}}$. Use these to compare the mixing rates for the two chains.

---

[9] To be systematic in your search, think of the matrices sorted by first row. Find all the matrices (a) with first row [1,1,1,0], (b) with first row [1,1,0,1], and (c) with first row [1,0,1,1] or [0,1,1,1].

[10] You can check your work using the fact that the number of matrices adjacent to **C** equals the number of swappable 2x2 sub-matrices ("checkerboard units" or CUs, for short) of **C**. This number, called the C-score by ecologists, equals the sum over all pairs (i,j) with $i < j$ of $(r_i - s_{ij})(r_j - s_{ij})$, where r_i is the total for row i of **C**, and s_{ij} is the i,j element of $\mathbf{S} = \mathbf{CC}^T$.

```
############################################################################
#                                                                          #
#     THIS FUNCTION TAKES A REVERSIBLE TRANSITION MATRIX P AND RETURNS     #
#     ITS EIGENVALUES, (LEFT) EIGENVECTORS, AND SPECTRAL DECOMPOSITION     #
#                                                                          #
############################################################################


#  Create a simple 4x4 transition matrix for a graph walk

r1 <- c(0, 1, 1, 0)/2
> r2 <- c(1, 0, 1, 0)/2
> r3 <- c(1, 1, 0, 1)/3
> r4 <- c(0, 0, 1, 0)
> P <- rbind(r1, r2, r3, r4)
```




```
# creates a square matrix by taking the outer product of a vector with itself
#
outsquare <- function(vector){
   return(outer(vector, vector))
}

# takes a vector, creates its "outsquare", and changes basis
#
component <- function(vector, A1, A2){
   return(t(A1 %*% outsquare(vector) %*%
      A2))
}

# takes a vector and rescales to length 1
#
normalize <- function(vector){
   if(sum(vector^2) < 1e-010) {
      return(0 * vector)
   }
   else {
      return(vector/sqrt(sum(
         vector^2)))
   }
}
```

```
#  Takes a reversible transition matrix P and returns its
#  eigenvalues, eigenvectors, and spectral decomposition
#
SpectralDecomp <- function(P){
   digs <- 5
   vals <- eigen(t(P))$values
   lambda <- rev(sort(vals))
   vecs <- t(eigen(t(P))$vectors)
   vecs <- vecs[order( - vals),  ]
   vecs <- t(apply(t(vecs), 2, normalize))
   vecs[1,  ] <- vecs[1,  ]/sum(vecs[1,])
   cat("\n")
   cat("Transition matrix P:", "\n")
   print(round(P, digits = digs))
   cat("\n")
   cat("Eigenvalues of P:", "\n")
   print(round(lambda, digits = digs))
   cat("\n")
   cat("Eigenvectors of P:", "\n")
   print(round(vecs, digits = digs))
   D1 <- diag(sqrt(vecs[1,  ]))
   D1.inv <- diag(1/sqrt(vecs[1,  ]))
   Q <- D1 %*% P %*% D1.inv
   cat("\n")
   cat("Matrix Q = (D^.5)(P)(D^-.5):", "\n")
   print(round(Q, digits = digs))
   vecs <- t(eigen(Q)$vectors)
   vecs <- vecs[order( - vals),  ]
   vecs <- tapply(t(vecs), 2, normalize))
   cat("\n")
   cat("Eigenvectors of Q:", "\n")
   print(round(vecs, digits = digs))
   cat("\n")
   cat("\n")
   cat("Spectral decomposition of P:", "\n")
   for(i in 1:dim(Q)[2]) {
      out.mat <- component(vecs[i,], D1, D1.inv)
      cat("Eigenvalue", i, " = ")
      cat(round(lambda[i], digits = digs), "\n")
      cat("Component matrix F", i,":", "\n")
      print(round(out.mat, digits = digs))
      cat("\n", "\n")
   }
}
```

```
> SpectralDecomp(P)

Transition matrix P:
      [,1]    [,2] [,3]    [,4]
r1 0.00000 0.50000  0.5 0.00000
r2 0.50000 0.00000  0.5 0.00000
r3 0.33333 0.33333  0.0 0.33333
r4 0.00000 0.00000  1.0 0.00000

Eigenvalues of P:
[1]  1.00000  0.22871 -0.50000 -0.72871

Eigenvectors of P:
       [,1]     [,2]     [,3]     [,4]
r1  0.25000  0.25000  0.37500  0.12500
r3  0.49572  0.49572 -0.40345 -0.58800
r2  0.70711 -0.70711  0.00000  0.00000
r4 -0.23293 -0.23293  0.85862 -0.39276

Matrix Q = (D^.5)(P)(D^-.5):
        [,1]    [,2]    [,3]    [,4]
[1,] 0.00000 0.50000 0.40825 0.00000
[2,] 0.50000 0.00000 0.40825 0.00000
[3,] 0.40825 0.40825 0.00000 0.57735
[4,] 0.00000 0.00000 0.57735 0.00000

Eigenvectors of Q:
         [,1]     [,2]     [,3]     [,4]
[1,] -0.50000 -0.50000 -0.61237 -0.35355
[2,] -0.43621 -0.43621  0.28987  0.73172
[3,] -0.70711  0.70711  0.00000  0.00000
[4,] -0.24438 -0.24438  0.73551 -0.58274


Spectral decomposition of P:

Eigenvalue 1  = 1
Component matrix F 1 :
     [,1] [,2]  [,3]  [,4]
[1,] 0.25 0.25 0.375 0.125
[2,] 0.25 0.25 0.375 0.125
[3,] 0.25 0.25 0.375 0.125
[4,] 0.25 0.25 0.375 0.125


Eigenvalue 2  = 0.22871
Component matrix F 2 :
         [,1]     [,2]     [,3]     [,4]
[1,]  0.19028  0.19028 -0.15486 -0.22570
[2,]  0.19028  0.19028 -0.15486 -0.22570
[3,] -0.10324 -0.10324  0.08402  0.12246
[4,] -0.45140 -0.45140  0.36737  0.53542


Eigenvalue 3  = -0.5
Component matrix F 3 :
     [,1] [,2] [,3] [,4]
[1,]  0.5 -0.5    0    0
[2,] -0.5  0.5    0    0
[3,]  0.0  0.0    0    0
[4,]  0.0  0.0    0    0


Eigenvalue 4  = -0.72871
Component matrix F 4 :
         [,1]     [,2]     [,3]     [,4]
[1,]  0.05972  0.05972 -0.22014  0.10070
[2,]  0.05972  0.05972 -0.22014  0.10070
[3,] -0.14676 -0.14676  0.54098 -0.24746
[4,]  0.20140  0.20140 -0.74237  0.33958
```

```
##########################################################################
#                                                                        #
#              CHECKERBOARD SCORE OF A CO-OCCURRENCE MATRIX              #
#                                                                        #
##########################################################################


#  The following function computes the C-score for a co-occurrence matrix C.
#  The (i,j) element of C is 1 if species i occurs on island j, 0 otherwise.
#  Species i,j form a "checkerboard unit" on islands u,v if i is present on
#   u but not v, and j is present on v but not u, or vice-versa.  The total
#   number of checkerboard units for a co-occurrence matrix is its C-score.
#  The C-score is computed here as follows:
#   r is a column vector of row totals for C.  Thus element i tells the
#      number of islands on which i occurs.
#   N is the sum of the elements of r.
#   S is the "sharing matrix" whose i,j element is the dot products of rows
#     i and j of C.  Thus the i,j element equals the number of islands on
#     which both i and j occur.  S is the product of C with its transpose.
#  The number of checkerboard units (CUs) formed by species i and j equals
#     (r[i]-S[i,j)*(r[j]-S[i,j]), and so the C-score equals the sum over all
#     pairs i < j of these quantities.
#  The C-score is computed here using an equivalent formula that substitutes
#     matrix multiplications for loops.
#
CScore <- function(C){                             # C is the co-occurrence matrix
   S <- C %*% t(C)                                 # S is the "sharing matrix"
   one <- rep(1,dim(C)[2])                         # vector of 1s
   r <- C %*% one                                  # row totals for C
   N <- sum(r)                                     # total number of 1s in C
   CUs <- .5*(N^2 + sum(S*S) -2*sum(S %*% r))  # number of checkerboard units
   return(CUs)
}


####### Examples


r1 <- c(1,0,1,0,1,1,1)
r2 <- c(0,1,1,1,0,0,0)
r3 <- c(1,0,0,0,1,1,1)
C1 <- rbind(r1,r2,r3)
C1
CScore(C1)


r1 <- c(1,0,1)
r2 <- c(0,1,0)
C2 <- rbind(r1,r2,r1)
C2
CScore(C2)


r1 <- c(1,1,1,0)
r2 <- c(1,1,0,0)
r3 <- c(1,0,1,0)
r4 <- c(0,0,0,1)
C3 <- rbind(r1,r2,r3,r4)
C3
CScore(C3)
```

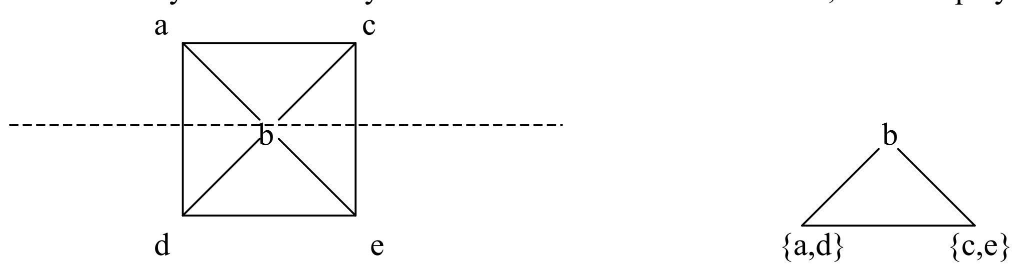
```
###### EXAMPLES ######

> C1
   [,1] [,2] [,3] [,4] [,5] [,6] [,7]
r1    1    0    1    0    1    1    1
r2    0    1    1    1    0    0    0
r3    1    0    0    0    1    1    1
> CScore(C1)
[1] 20


> C2
   [,1] [,2] [,3]
r1    1    0    1
r2    0    1    0
r1    1    0    1
> CScore(C2)
[1] 4


> C3
   [,1] [,2] [,3] [,4]
r1    1    1    1    0
r2    1    1    0    0
r3    1    0    1    0
r4    0    0    0    1
> CScore(C3)
[1] 8
```

Random Walks on Co-Occurrence Matrices: Some Questions for Investigation

64. If you know the marginal totals, what can you say about the number of checkerboard units?

65. What is the relationship between equalities in the marginal totals and symmetries in the corresponding graph?

66. For which symmetries can you derive a "folded" Markov chain, as in Display 8.5?



Display 8.8. Folding a graph along a line of symmetry.
The original graph is on the left; the folded graph at the right has vertices formed by sets of vertices from the original graph.

Transition probabilities $\tilde{p}$ for the folded graph are defined as follows:[11]

From a set of vertices to a single vertex. Example: From $\{a,d\}$ to b:
This probability is defined only if the probability of going from a to b is the same as the probability of going from d to b, that is, only if $p_{ab} = p_{db}$. If the probability of going to e is the same for all elements of the set, then that probability is taken to be the probability of going from the set to e in the folded chain: $\tilde{p}_{\{a,d\}b} = p_{ab} = p_{db}$.

From a single vertex to a set of vertices. Example: From b to $\{a,d\}$:
The probability of going to the set is the sum of the probabilities of going to the vertices in the set: $\tilde{p}_{b\{a,d\}} = p_{ba} + p_{bd}$.

From one set of vertices to another. Example: From $\{a,d\}$ to $\{c,e\}$:
This probability is defined only if the probability of going from a to $\{c,e\}$ is the same as the probability of going from d to $\{c,e\}$, that is, only if $p_{ac} + p_{ae} = p_{dc} + p_{de}$. If the probability of going to the destination set $\{c,e\}$ is the same for all elements of the origin set $\{a,d\}$, then that probability is taken to be the probability of going from the origin set to the destination set in the folded chain: $\tilde{p}_{\{a,d\}\{c,e\}} = p_{ac} + p_{ae} = p_{dc} + p_{de}$.

[11] In Markov chain theory, what I have called a "folded" chain is called a "lumped" chain. See, for example, John G. Kemey and J. Laurie Snell (1960) *Finite Markov Chains*, Princeton: Van Nostrand, pp. 123-132.

67. Is there a quick way to get the folded chain directly from the marginal totals of the co-occurrence matrix?

68. If you Metropolize and then fold, will the folded chain have a uniform stationary distribution?

869. When (for which sets of marginal totals) do the operations of folding and Metropolizing commute?

70. How do convergence rates for the original and folded chains compare?

71. Is there a way to exploit the symmetries of a graph to get bounds on the eigenvalues of the (unfolded) Markov chain?