Example: A Markov Process

Divide the greater metro region into three parts: city (such as St. Louis), suburbs (to include such areas as Clayton, University City, Richmond Heights, Maplewood, Kirkwood,...) and exurbs (the far out areas where people associated with the metro area might live: for example St. Charles county, Jefferson County, ...)

In this oversimplified model, we ignore people entering/leaving the region; we ignore births/deaths. We assume that the total population is constant and that people move around between the city, suburbs and exurbs. Data collection lets us estimate how the population shifts from year to year:

Suppose the transition matrix A is:

$$
\begin{array}{c}
\text{Moving From} \\
\begin{array}{ccc} C & S & E \end{array} \\
\begin{array}{ccc} \downarrow & \downarrow & \downarrow \end{array}
\end{array}
$$

$$
A = \begin{bmatrix} .75 & .10 & .10 \\ .10 & .60 & .20 \\ .15 & .30 & .70 \end{bmatrix} \text{ to} \rightarrow \begin{cases} \text{C (City)} \\ \text{S (Suburbs)} \\ \text{E (Exurbs)} \end{cases}
$$

Going down Column 1 tells us what proportion (%) of the C population will move to C, S, E during the year. Column 1 accounts for where the whole city population lives at the end of a year, so Column 1 adds up to 1 (100%). Similarly, Sum(Column 2) = 1 and Sum(Column 3) = 1.

Suppose the metro region has a total population of 2,000,000 distributed as:

$$
\begin{bmatrix} 400000 \\ 1400000 \\ 200000 \end{bmatrix} \begin{cases} \text{City} \\ \text{Suburb} \\ \text{Exurbs} \end{cases}
$$

After one year the new distribution is

$$
\begin{bmatrix} .75 & .10 & .10 \\ .10 & .60 & .20 \\ .15 & .30 & .70 \end{bmatrix} \begin{bmatrix} 400000 \\ 1400000 \\ 200000 \end{bmatrix} = \begin{bmatrix} 460000 \\ 920000 \\ 620000 \end{bmatrix} \begin{cases} \text{City} \\ \text{Suburb} \\ \text{Exurbs} \end{cases}
$$

To avoid working with such large numbers, we will use a "population state vector" written instead with proportions: for the initial state vector x_0, the 2000000 people in the region the are distributed in proportions as

$$
\begin{bmatrix} \frac{400000}{2000000} \\ \frac{1400000}{2000000} \\ \frac{200000}{2000000} \end{bmatrix} = \begin{bmatrix} .20 \\ .70 \\ .10 \end{bmatrix} = x_0 \qquad \begin{cases} \% \text{ C} \\ \% \text{ S} \\ \% \text{ E} \end{cases}
$$

Then $\quad x_1 = Ax_0 = \begin{bmatrix} .75 & .10 & .10 \\ .10 & .60 & .20 \\ .15 & .30 & .70 \end{bmatrix} \begin{bmatrix} .20 \\ .70 \\ .10 \end{bmatrix} = \begin{bmatrix} .23 \\ .46 \\ .31 \end{bmatrix} \quad \begin{cases} \% \text{ C} \\ \% \text{ S} \\ \% \text{ E} \end{cases} \quad$ is next

year's population distribution. After 2 years, the population state distribution is

$$x_2 = Ax_1 = A(Ax_0)) = A^2x_0 = \begin{bmatrix} .75 & .10 & .10 \\ .10 & .60 & .20 \\ .15 & .30 & .70 \end{bmatrix} \begin{bmatrix} .23 \\ .46 \\ .31 \end{bmatrix} = \begin{bmatrix} .2495 \\ .3610 \\ .3895 \end{bmatrix}$$

and so on:

$$x_{n+1} = Ax_n = A(Ax_{n-1}) = A(A(Ax_{n-2})) = ... = A^{n+1}x_0 \quad (**)$$

Each <u>state vector</u> x_n, after the initial vector x_0, is obtained from the preceding one by multiplication by A.

Notice that:

      1) We can also think of the proportions in A and in x_0 as <u>probabilities</u>: for example, the probability is 0.30 that a randomly chosen person from the suburbs will move to the exurbs by next year. The 0.70 in the second row of x_0 tells us that the probability is 0.70 that a randomly chosen person (out of the 2000000 in the region) lives in the suburbs.

      2) The entries in A and x_0 are nonnegative numbers, and the columns add up to 1. A square matrix A satisfying these two conditions is called a <u>stochastic</u> matrix, and such a vector x_0 is called a <u>probability vector</u>. As we compute, each succeeding state vector $x_1, x_2, ..., x_n, ...$ is <u>still a probability vector</u> (*we check this for the 3×3 case, but the argument in the case of an $n \times n$ stochastic matrix is completely similar*):

If $A = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix}$ is a stochastic matrix and $\begin{bmatrix} x \\ y \\ z \end{bmatrix}$ is a probability vector, then

$\begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} ax + by + cz \\ dx + ey + fz \\ gx + hy + iz \end{bmatrix}$. The entries are still ≥ 0, and they still sum to 1 :

$$(a + d + g)x + (b + e + h)y + (c + f + i)z \quad = 1 \cdot x + 1 \cdot y + 1 \cdot z$$
$$= x + y + z \quad = 1$$

A system consisting of a stochastic matrix, an initial state probability vector x_0 and an equation $x_{n+1} = Ax_n$ is called a <u>Markov process</u>.

In a Markov process, each successive state x_{n+1} depends only on the preceding state x_n. An important question about a Markov process is "What happens in the long-run?", that is, "what happens to x_n as $n \to \infty$?"

In our example, we can start with a good guess. Using Matlab, I (quickly) computed

$$\boldsymbol{x}_{10} = A^{10}\boldsymbol{x}_0 = \begin{bmatrix} .2846 \\ .2862 \\ .4292 \end{bmatrix}, \ \ldots, \ \boldsymbol{x}_{100} = A^{100}\boldsymbol{x}_0 = \begin{bmatrix} .2857 \\ .2857 \\ .4286 \end{bmatrix} \text{ and that}$$

$$\boldsymbol{x}_{1000} = A^{1000}\boldsymbol{x}_0 = \begin{bmatrix} .2857 \\ .2857 \\ .4286 \end{bmatrix} \quad \textit{(The \underline{displayed} results are rounded to 4 decimal places; during}$$

the calculations, Matlab carried along many more decimal places – although even then small roundoffs were made.)

Assuming the transition matrix and other modeling assumptions remain valid as time passes, it seems like the population distribution moves toward a steady state $\boldsymbol{x} = \begin{bmatrix} .2857 \\ .2857 \\ .4286 \end{bmatrix}$ – with 28.57% of the population in each of the city and suburbs, and with 42.86% in the exurbs.

Using our knowledge of linear algebra, we can actually find this steady state $\boldsymbol{x}$ without repeated computations and guessing. Is there a <u>steady state probability vector</u> $\boldsymbol{x}$ for which $A\boldsymbol{x} = \boldsymbol{x}$? equivalently, for which $(A - I)\boldsymbol{x} = \boldsymbol{0}$? We begin by finding <u>all</u> $\boldsymbol{x}$ that satisfy the equation. Then among those solutions, we find an $\boldsymbol{x}$ that is <u>also a probability vector</u>.

Here, $A - I = \begin{bmatrix} -.25 & .10 & .10 \\ .10 & -.40 & .20 \\ .15 & .30 & -.30 \end{bmatrix}$. To avoid any roundoff error, we can convert to fractions and row reduce the augmented matrix for $(A - I)\boldsymbol{x} = \boldsymbol{0}$:

$$\begin{bmatrix} -\frac{1}{4} & \frac{1}{10} & \frac{1}{10} & 0 \\ \frac{1}{10} & -\frac{4}{10} & \frac{2}{10} & 0 \\ \frac{15}{100} & \frac{3}{10} & -\frac{3}{10} & 0 \end{bmatrix} = \begin{bmatrix} -\frac{1}{4} & \frac{1}{10} & \frac{1}{10} & 0 \\ \frac{1}{10} & -\frac{2}{5} & \frac{1}{5} & 0 \\ \frac{3}{20} & \frac{3}{10} & -\frac{3}{10} & 0 \end{bmatrix} \sim \ldots \sim \begin{bmatrix} 1 & 0 & -\frac{2}{3} & 0 \\ 0 & 1 & -\frac{2}{3} & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

The general solution is $\boldsymbol{x} = x_3 \begin{bmatrix} \frac{2}{3} \\ \frac{2}{3} \\ 1 \end{bmatrix}$ where x_3 is free. A more convenient (rescaled) form is

$\boldsymbol{x} = s \begin{bmatrix} 2 \\ 2 \\ 3 \end{bmatrix}$. From among the solutions $\boldsymbol{x}$, we want the one that is <u>also</u> a probability vector. We

get it by choosing $s = \frac{1}{7}$, so that $\boldsymbol{x} = \frac{1}{7} \begin{bmatrix} 2 \\ 2 \\ 3 \end{bmatrix} = \begin{bmatrix} \frac{2}{7} \\ \frac{2}{7} \\ \frac{3}{7} \end{bmatrix}$ has entries that add

to 1. This is <u>the one and only probability vector</u> that is a solution to $A\boldsymbol{x} = \boldsymbol{x}$ so $\begin{bmatrix} \frac{2}{7} \\ \frac{2}{7} \\ \frac{3}{7} \end{bmatrix}$ is the steady

state vector for the Markov process in our example.

Rounded to 4 decimal places : $\begin{bmatrix} \frac{2}{7} \\ \frac{2}{7} \\ \frac{3}{7} \end{bmatrix} = \begin{bmatrix} .2857 \\ .2857 \\ .4286 \end{bmatrix}$, *the result estimated using Matlab.*

This is exactly what happens in many cases with a Markov process. <u>The following is a result proven in courses that treat Markov processes in detail.</u>

Definition An $n \times n$ stochastic matrix A is called <u>regular</u> if for some positive integer , the entries in the power A^k are all > 0 (*not merely* ≥ 0).

In the example above, A is regular because $A = A^1$ has all entries > 0.

Theorem If A is a regular $n \times n$ stochastic matrix, then

    i) there <u>exists</u> a <u>unique</u> steady state probability vector x, that is, a probability vector for which $Ax = x$ and

    ii) $x_{n+1} = Ax_n \to x$ as $n \to \infty$ (and this is true no matter which <u>probability vector is used as the initial state x_0</u>.

The little that we already know about diagonalization and eigenvectors also sheds some light on this Markov process – because the matrix A happens to be diagonalizable.

Recall that:

Definition   A <u>nonzero</u> vector v is called an <u>eigenvector</u> of the $n \times n$ matrix A if $Av = \lambda v$ for some scalar λ. The scalar λ is called an <u>eigenvalue</u> of A (associated with the eigenvector v).

Suppose v is an eigenvector of (any $n \times n$ matrix) A with eigenvalue λ.

Then
$$Av = \lambda\, v$$
$$A^2 v = A(Av) = A(\lambda\, v) = \lambda\, A(v) = \lambda\,(\lambda\, v) = \lambda^2 v$$
$$A^3 v = A(A^2 v) = A(\lambda^2 v) = \lambda^2 A(v) = \lambda^2 \lambda\, v = \lambda^3 v$$
$$\vdots$$
$$A^n v = \ldots = \lambda^n v$$

For our example, $A = \begin{bmatrix} .75 & .10 & .10 \\ .10 & .60 & .20 \\ .15 & .30 & .70 \end{bmatrix}$ has eigenvectors $v_1, v_2,$ and v_3 with

corresponding eigenvalues $\lambda_1 = \frac{13}{20} = 0.65, \quad \lambda_2 = 1, \quad \lambda_3 = \frac{2}{5} = 0.40$ <u>and</u> these eigenvectors form <u>a basis</u> for $\mathbb{R}^3$ – so A is diagonalizable.

> *In the recent supplementary homework, we saw a method for finding eigenvalues λ: find the λ's (if any) that make $\det(A - \lambda I) = 0$. Actually applying the method for a 3×3 matrix A leads to a cubic equation that must be solved for λ – this can be done, but may be difficult depending on the matrix A in general. If we find any eigenvalues λ, then we can solve $Ax = \lambda x$ to find the corresponding eigenvectors.*

For the example, I used Matlab to help find the "diagonal factorization" for A:

$$A = \begin{bmatrix} .75 & .10 & .10 \\ .10 & .60 & .20 \\ .15 & .30 & .70 \end{bmatrix} = P \begin{bmatrix} .65 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & .40 \end{bmatrix} P^{-1}, \; \underline{\text{where}}$$

$$P = \begin{bmatrix} -.8111 & .4851 & 0 \\ .3244 & .4851 & -.7171 \\ .4867 & .7276 & .7071 \end{bmatrix} \quad \textit{(rounded to 4 decimal places)}$$

The <u>columns</u> of P are (approximately) the eigenvectors $v_1, v_2,$ and v_3 that form a basis for $\mathbb{R}^3$, and the eigenvalues for $v_1, v_2,$ and v_3 are the entries in the diagonal matrix: .65, 1, and .4 Since the eigenvectors are a <u>basis</u> for $\mathbb{R}^3$, the initial state vector x_0 can be written as a linear combination of the basis eigenvectors:

$$x_0 = \begin{bmatrix} .20 \\ .70 \\ .10 \end{bmatrix} = c_1 v_1 + c_2 v_2 + c_3 v_3 \; \textit{(we could find the weights } c_1, c_2, c_3 \textit{ if needed)}$$

Therefore

$$\boldsymbol{x_1} = A\boldsymbol{x_0} = A(c_1\boldsymbol{v_1} + c_2\boldsymbol{v_2} + c_3\boldsymbol{v_3}) = c_1 A\boldsymbol{v_1} + c_2 A\boldsymbol{v_2} + c_3 A\boldsymbol{v_3}$$
$$= c_1\lambda_1\boldsymbol{v_1} + c_2\lambda_2\boldsymbol{v_2} + c_3\lambda_3\boldsymbol{v_3}$$

$$x_2 = A^2\boldsymbol{x_0} \quad = A(c_1\lambda_1\boldsymbol{v_1} + c_2\lambda_2\boldsymbol{v_2} + c_3\lambda_3\boldsymbol{v_3})$$

$$= c_1\lambda_1 A\boldsymbol{v_1} + c_2\lambda_2 A\boldsymbol{v_2} + c_3\lambda_3 A\boldsymbol{v_3}$$

$$= c_1\lambda_1^2\boldsymbol{v_1} + c_2\lambda_2^2\boldsymbol{v_2} + c_3\lambda_3^2\boldsymbol{v_3}$$
$$\vdots$$
$$\boldsymbol{x_n} = A^n\boldsymbol{x_0} = ... \quad\quad ... \quad = c_1\lambda_1^n\boldsymbol{v_1} + c_2\lambda_2^n\boldsymbol{v_2} + c_3\lambda_3^n\boldsymbol{v_3}$$

$$= c_1(.65)^n\boldsymbol{v_1} + c_2(1)^n\boldsymbol{v_2} + c_3(.40)^n\boldsymbol{v_3}$$

Therefore we can see that

$$\lim_{n\to\infty}\boldsymbol{x_n} \quad = \quad \lim_{n\to\infty}\left(c_1(.65)^n\boldsymbol{v_1} + c_2(1)^n\boldsymbol{v_2} + c_3(.40)^n\boldsymbol{v_3}\right)$$

$$= \quad \lim_{n\to\infty}\left(0\boldsymbol{v_1} + c_2\boldsymbol{v_2} + 0\boldsymbol{v_3}\right) = c_2\boldsymbol{v_2}$$

If we had actually found the weights c_1, c_2, c_3 above then we would now know the steady state probability vector $\boldsymbol{x} = c_2\boldsymbol{v_2}$ (*because we would know c_2*)
.
But, instead, we can still find $\boldsymbol{x}$, because we know $\boldsymbol{x}$ is supposed to be a probability vector:

$$\boldsymbol{v_2} \text{ is (approximately) the second column of } P = \begin{bmatrix} .4851 \\ .4851 \\ .7276 \end{bmatrix} \text{ and}$$

for $\boldsymbol{x} = c_2\boldsymbol{v_2}$ to be a probability vector, the sum of the entries must be 1 :

this requires use to choose $c_2 = \frac{1}{\text{sum of the entries of } \boldsymbol{v_2}}$

$$= \frac{1}{.4851 + .4851 + .7276} \approx .5890$$

$$\text{So } \boldsymbol{x} \approx .5890\begin{bmatrix} .4851 \\ .4851 \\ .7276 \end{bmatrix} = \begin{bmatrix} .2857 \\ .2857 \\ .4286 \end{bmatrix} \text{ (rounded to 4 decimal places)}$$

$$= \text{ the same steady state } \boldsymbol{x} \text{ we found earlier.}$$

So: <u>diagonalization can help in understanding Markov Processes and similar kinds of linear difference equations</u>.