Markov chain theory so far

$X_0, X_1, X_2, \dots$ stoch. process, state space S

$\mathbb{P}(X_{n+1} = j \mid X_0 = i_0, \dots, X_n = i) = \mathbb{P}(X_{n+1} = j \mid X_n = i)$ M. property

$\mathbb{P}(X_{n+1} = j \mid X_n = i) = P_{ij}$ time homog.

If μ_0 is the distn of X_0, then $\mu_0 P^n$ is the distn of X_n.
If $\lim \mu_n$ exists and equals π, then $\pi = \pi P$, so π is an invariant distribution.
Let T be the first time X hits set $A \subseteq S$.

Let $u_k = \mathbb{E}[T \mid X_0 = k]$

Then
$$\begin{cases} u_k = 0 & \text{if } k \in S \\ \vec{u} = 1 + P\vec{u} & \text{otherwise,} \end{cases}$$

by first-step analysis.

Accessibility IV.3

Definition $i \to j$ (I can lead to j, j is accessible from i) if

$\mathbb{P}(X_n = j \text{ for some } n \geq 0 \mid X_0 = i) > 0.$

Starting from i, there is a possibility of later visiting j.

Note that $i \to i$.

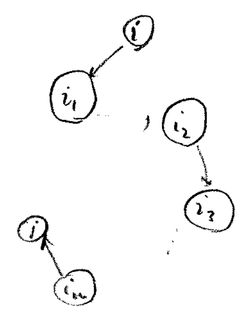
Proposition These are equivalent:

a) $i \to j$

b) $\mathbb{P}(X_n = j \mid X_0 = i) = P^n_{ij} > 0$ for some n

c) There is a sequence $i_1, i_2, i_3, \ldots, i_m$ such that

$P_{i,i_1} > 0, \quad P_{i_1,i_2} > 0, \quad \ldots, \quad P_{i_m,j} > 0.$



Proof

(a) $\Rightarrow$ (b)

Suppose $i \to j$. Then,

$$0 < \mathbb{P}(X_n = j \text{ for some } n \geq 0 \mid X_0 = i)$$

$$= \mathbb{P}\left(\bigcup_{n=0}^{\infty} \{X_n = j\} \mid X_0 = i\right)$$

$$\leq \sum_{n=0}^{\infty} \mathbb{P}(X_n = j \mid X_0 = i)$$

$$= \sum_{n=0}^{\infty} P^n_{ij}.$$

Thus at least one of these terms must be non-zero.

(b) $\Rightarrow$ (c)

Suppose $P^n_{ij} > 0$.

$$0 < P^n_{ij} = P(X_n = j \mid X_0 = i)$$

$$= \sum_{i_1, i_2, \ldots} \mathbb{P}(X_1 = i_1, X_2 = i_2, \ldots, X_n = j \mid X_0 = i)$$

$$= \sum P_{i,i_1} P_{i_1,i_2} P_{i_2,i_3} \cdots P_{i_{m},j}.$$

At least one of these is nonzero, and that one gives us a path $i, i_1, i_2, \ldots, i_m, j$ which works.

(c) $\Rightarrow$ (a)

Suppose $P_{i,i_1} > 0$, $P_{i_1,i_2} > 0$, ...

Then $0 < \mathbb{P}(X_1 = i_1, X_2 = i_2, \ldots, X_{m+1} = j \mid X_0 = i)$

$\leq \mathbb{P}(X_{m+1} = j \mid X_0 = i)$

$\leq \mathbb{P}(X_n = j \text{ for some } n \geq 0 \mid X_0 = i)$. $\square$

Corollary If $i \to j$ and $j \to k$, then $i \to k$.

Proof Write your own.

Definition

X is irreducible if, for all i, j in S, $i \to j$.

State i in S is absorbing if $P_{ii} = 1$.

Example Gambler's wealth

$10 \to N$ but $N \not\to 10$; not irreducible.

0 and N are absorbing.

Example Queueing, size N

$0 \to i$ for $i = 0, 1, \ldots, N$.

Irreducible.

No absorbing states.

Def: Transience and recurrence

If $\mathbb{P}(X_n = i \text{ for some } n \geq 1 \mid X_0 = i) < 1$,

then starting from state i, it is possible that X never visits state i again.

State i is said to be transient.

On the other hand, if $\mathbb{P}(X_n = i \text{ for some } n \geq 1 \mid X_0 = i) = 1$,

then another visit to state i is guaranteed. Then i is recurrent.

Example Gambler's wealth.

With $X_0 = 10$, it is possible to never return to state 10, so 10 is transient.

Same with 1 to $N-1$.

States 0 and N are recurrent. Starting there, the chain will certainly visit there again.

Example Queueing, limit of N customers.

Start with $X_0 = 0$.

Is the queue guaranteed to empty out again? It's not so clear.

Here it is hard to tell, but we'll be able to pretty soon.

Proposition Suppose $i \to j$ but $j \not\to i$.

Then i is transient.

Proof

We need to show that it is possible to never return to i.

Because $i \to j$, there is a sequence $i_1, i_2, i_3, \dots, i_m$ leading to j.

We can assume that this sequence does not revisit i, otherwise we could shorten it.

$$\mathbb{P}(X_n \neq i \text{ for all } n \geq 1 \mid X_0 = i)$$

$$\geq \mathbb{P}(X_1 = i_1, X_2 = i_2, \dots, X_m = i_m, X_{m+1} = j, X_{m+n} \neq i \text{ for all } n \geq 1 \mid X_0 = i)$$

$$= \mathbb{P}(X_{m+n} \neq i \text{ for all } n \geq 1 \mid X_{m+1} = j)\, \mathbb{P}(X_1 = i_1, \dots, X_{m+1} = j \mid X_0 = i)$$

Markov property

$$= \mathbb{P}(X_n \neq i \text{ for all } n \geq 0 \mid X_0 = j)\, \mathbb{P}(\qquad)$$

time homogeneity

$$= 1 \cdot \text{positive}$$

$$> 0.$$

We will see later that when the state space is finite, all transient states work like this.

(2) State i is <u>recurrent</u> if, starting in state i, the chain is guaranteed to visit state i again:

$$\mathbb{P}_i(X_n = i \text{ for some } n \geq 1) = 1.$$

State i is <u>transient</u> if, starting in state i, returning to state i is not guaranteed:

$$\mathbb{P}_i(X_n = i \text{ for some } n \geq 1) < 1$$

In other words, it is possible to never return to state i:

$$\mathbb{P}_i(X_n \neq i \text{ for all } n \geq 1) > 0.$$

Absorbing states are clearly recurrent.

If $i \to j$ but $j \not\to i$, then we showed that i is transient.

<u>Today:</u> get a grip on the number of times a state is visited ....

(1) <u>Notation:</u> $\mathbb{P}_i(A)$ means $\mathbb{P}(A \mid X_0 = i)$

$\uparrow$
starting point of the chain.

Conditioning on $X_0 = i$ just changes the probability measure.

Note: $\mathbb{P}_i(X_0 = i) = 1$

$\mathbb{P}_i(X_0 = j) = 0$ if $i \neq j$.

Probability of ever visiting a state:

$$f_{ij} = \mathbb{P}_i(X_n = j \text{ for some } n \geq 1)$$

Recurrent : $f_{ii} = 1$

Transient : $f_{ii} < 1$.

Number of visits to state j:

$$N_j = \# \text{ visits to state } j, \text{ including time } 0$$
$$= \sum_{n=0}^{\infty} 1_{\{X_n = j\}}$$

Possible values $0, 1, 2, 3, \dots, \infty$ ★

The Markov property helps us find the distribution of N_j :

Proposition Suppose $i \neq j$.

$$\mathbb{P}(N_j = n \mid X_0 = i) = \begin{cases} 1 - f_{ij}, & n = 0 \\ f_{ij} \, f_{jj}^{n-1} (1 - f_{jj}), & n = 1, 2, 3, \dots \end{cases}$$

Proof

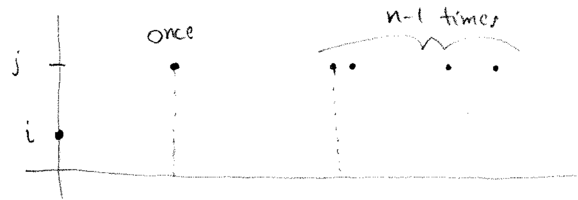
$n = 0$: $\quad \mathbb{P}(N_j = 0 \mid X_0 = i) = \mathbb{P}(\text{never visit } j \mid X_0 = i) = 1 - f_{ij}$

$n \geq 1$: $\quad \mathbb{P}(N_j = n \mid X_0 = i) = \mathbb{P}(\text{hit } j \text{ once, return } n-1 \text{ times, never hit again} \mid X_0 = i)$

$= \mathbb{P}(\text{hit } j \text{ once} \mid X_0 = i) \, \mathbb{P}(\text{return } n-1 \text{ times} \mid X_0 = j) \, \mathbb{P}(\text{never hit } j \mid X_0 = j)$

using the Markov prop. and time homog.

$= f_{ij} \quad f_{jj}^{n-1} \quad (1 - f_{jj})$.



It is slightly different when $i = j$:

$$\mathbb{P}(N_j = n \mid X_0 = i) = \begin{cases} 0, & \text{if } n = 0 \\ 1 - f_{ii}, & \text{if } n = 1 \\ f_{ii}^{n-1} (1 - f_{ii}) & \text{if } n = 2, 3, \dots \end{cases}$$

$$= \begin{cases} 0, & \text{if } n = 0 \\ f_{ii}^{n-1} (1 - f_{ii}) & \text{if } n = 1, 2, 3, \dots \end{cases}$$

Number of visits to transient and recurrent states

If state i is recurrent, then $f_{ii} = 1$, and

$$\mathbb{P}(N_i = n \mid X_0 = i) = \begin{cases} 0, & \text{if } n = 0 \\ 0, & \text{if } n = 1,2,3,\dots \\ 1, & \text{if } n = \infty. \end{cases}$$

That is, starting in state i, there will be infinitely many visits to state i, with probability one.

Natural : every time we return to state i, we can forget the past, and since $f_{ii} = 1$, we are guaranteed to return to i.

If state i is transient, then $f_{ii} < 1$, and

$$\mathbb{P}(N_i = n \mid X_0 = i) = \begin{cases} 0, & \text{if } n = 0 \\ f_{ii}^{n-1}(1-f_{ii}), & \text{if } n = 1,2,3,\dots \end{cases}$$

This is a geometric distribution.

Note that
$$\sum_{n=0}^{\infty} \mathbb{P}(N_i = n \mid X_0 = i) = \sum_{n=1}^{\infty} f_{ii}^{n-1}(1-f_{ii})$$
$$= (1-f_{ii}) \cdot \frac{1}{1-f_{ii}}$$
$$= 1,$$

so that $\mathbb{P}(N_i = \infty \mid X_0 = i) = 0$.

Thus, starting in a transient state, with probability one, that state will be visited only finitely many times.

Think: Bernoulli trials.
"Success" means never returning to i.
Each time the chain visits i is a failure.
The number of trials until success has a geometric distribution.

$$\mathbb{E}[N_i \mid X_0 = i] = \frac{1}{1-f_{ii}}.$$

Expected number of visits to j starting at $i \neq j$

1. Suppose j is recurrent.

Then $f_{jj} = 1$,
$$\mathbb{P}(N_j = n \mid X_0 = i) = \begin{cases} 1 - f_{ij}, & \text{if } n = 0 \\ 0, & \text{if } n = 1, 2, 3, \ldots \\ f_{ij}, & \text{if } n = \infty \end{cases}$$

Thus,
$$\mathbb{E}[N_j \mid X_0 = i] = \begin{cases} 0, & \text{if } f_{ij} = 0 \\ \infty, & \text{if } f_{ij} > 0. \end{cases}$$

2. Suppose j is transient.

Then
$$\begin{aligned} \mathbb{E}[N_j \mid X_0 = i] &= \sum_{n=0}^{\infty} n \cdot \mathbb{P}(N_j = n \mid X_0 = i) \\ &= \sum_{n=1}^{\infty} n \cdot f_{ij} f_{jj}^{n-1} (1 - f_{jj}) \\ &= f_{ij}(1 - f_{jj}) \sum_{n=1}^{\infty} n f_{jj}^{n-1} \\ &= f_{ij}(1 - f_{jj}) \frac{1}{(1 - f_{jj})^2} \\ &= \frac{f_{ij}}{1 - f_{jj}}. \end{aligned}$$

$$\boxed{\begin{aligned} &\sum_{n=0}^{\infty} r^n = \frac{1}{1-r} \\ &\frac{d}{dr} \\ &\sum_{n=1}^{\infty} n r^{n-1} = \frac{1}{(1-r)^2} \end{aligned}}$$

This is finite.

3. Either way,
$$\mathbb{E}[N_j \mid X_0 = i] = \frac{f_{ij}}{1 - f_{jj}},$$
provided that we interpret $\frac{0}{0}$ as 0 and $\frac{1}{0}$ as ∞ to handle the recurrent case.

4. Starting in state j, the only difference is that there is guaranteed to be one more visit:

Recurrent: $\mathbb{E}[N_j \mid X_0 = j] = \infty$, $\quad N_j = \infty$ a.s.

Transient: $\mathbb{E}[N_j \mid X_0 = j] = 1 + \dfrac{f_{jj}}{1 - f_{jj}} = \dfrac{1 - f_{jj} + f_{jj}}{1 - f_{jj}} = \dfrac{1}{1 - f_{jj}}.$

Criterion for recurrence

Proposition State j is recurrent iff $\mathbb{E}_i N_j = \infty$ for some i

iff $\sum_{n=1}^{\infty} P^n_{ij} = +\infty$ for some i

Proof

$$\mathbb{E}_i N_j = \mathbb{E}_i \sum_{n=1}^{\infty} 1_{\{X_n = j\}}$$

$$= \sum_{n=1}^{\infty} \mathbb{E}_i 1_{\{X_n = j\}}$$

$$= \sum_{n=1}^{\infty} \mathbb{P}_i (X_n = j)$$

$$= \sum_{n=1}^{\infty} P^n_{ij},$$

so the last two are clearly equivalent.

Also, if j is recurrent, then $\mathbb{P}_j (N_j = \infty) = 1$, so $\mathbb{E}_j N_j = \infty$.

And, if j is transient, then $\mathbb{E}_i N_j < \infty$ for all i.

Proposition Suppose the state space S is finite. Then at least one state is recurrent.

Proof For each n, i,

$$\sum_{j \in S} P^n_{ij} = 1$$

Summing over n,

$$\sum_{n=1}^{\infty} \sum_{j \in S} P^n_{ij} = \infty$$

Interchanging,

$$\sum_{j \in S} \left(\sum_{n=1}^{\infty} P^n_{ij} \right) = \infty .$$

Thus, for some j,

$$\sum_{n=1}^{\infty} P^n_{ij} = +\infty = \mathbb{E}_i N_j,$$

$$= \frac{f_{ij}}{1 - f_{jj}}$$

Example Queueing system. System size C_n, $n \geq 0$.

Then at least one state is recurrent.

We're getting somewhere!

Proposition Suppose j is recurrent and $j \to k$.
Then k is recurrent.

Proof ② Starting in state j, there will be infinitely many visits to state j.

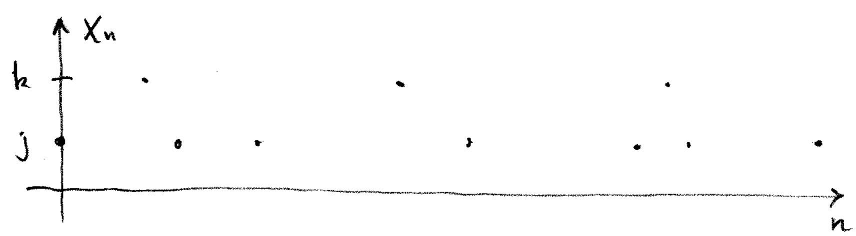
① From j, it is possible to visit k before visiting j again (path ...).

③ Each visit is like a Bernoulli trial, with success if the chain visits state k before returning to j.

With infinitely many trials, there will be infinitely many successes, so infinitely many visits to state k, with probability one.

Only recurrent states may be visited infinitely many times.

Thus, k is recurrent.



□

Example Queueing, size N.

At least one state is recurrent; call it j.

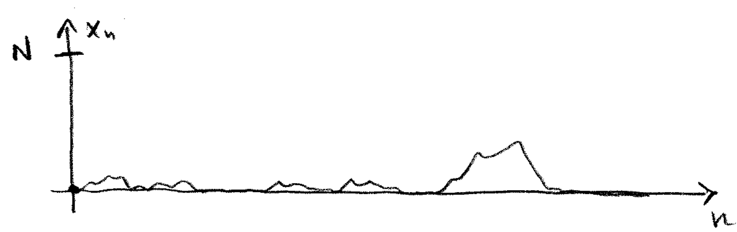
From j, the chain can reach any other state k.

Thus, all states are recurrent.

All states will be visited infinitely many times, although some more often than others.

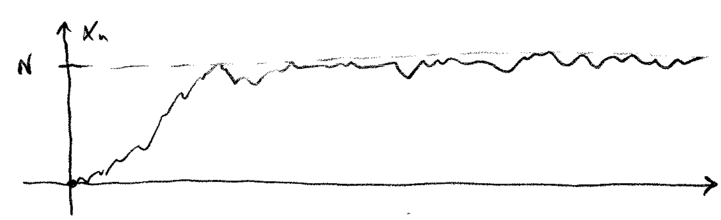
Case: r close to 0

N will be visited infinitely often



Case: $r \geq 1$

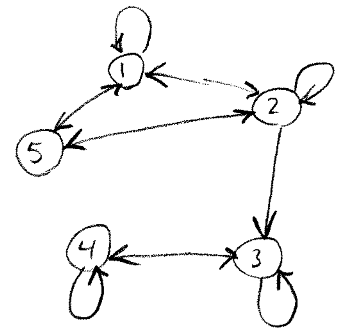
0 will be visited infinitely often



Computing the expected number of visits

Practice problem # 7

$$
P = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array}
\begin{array}{c}
\begin{array}{ccccc} 1 & 2 & 3 & 4 & 5 \end{array} \\
\begin{bmatrix}
0.2 & 0.7 & 0 & 0 & 0.1 \\
0.3 & 0.2 & 0.2 & 0 & 0.3 \\
0 & 0 & 0.3 & 0.7 & 0 \\
0 & 0 & 0.6 & 0.4 & 0 \\
0.6 & 0.4 & 0 & 0 & 0
\end{bmatrix}
\end{array}
$$



Communicating classes:

$\{1, 2, 5\}, \quad \{3, 4\}$

Closed set(s):

$\{3, 4\}$

Transient states:

$\{1, 2, 5\}$

Recurrent:

$\{3, 4\}$

Block form:

$$
P = \begin{array}{c} \\ 3 \\ 4 \\ 1 \\ 2 \\ 5 \end{array}
\begin{array}{c}
\begin{array}{ccccc} 3 & 4 & 1 & 2 & 5 \end{array} \\
\left[\begin{array}{cc|ccc}
0.3 & 0.7 & 0 & 0 & 0 \\
0.6 & 0.4 & 0 & 0 & 0 \\
\hline
0 & 0 & 0.2 & 0.7 & 0.1 \\
0.2 & 0 & 0.3 & 0.2 & 0.3 \\
0 & 0 & 0.6 & 0.4 & 0
\end{array}\right]
\end{array}
$$

$$
P^\infty = \begin{array}{c} 3 \\ 4 \\ 1 \\ 2 \\ 5 \end{array}
\left[\begin{array}{cc|ccc}
a & b & 0 & 0 & 0 \\
a & b & 0 & 0 & 0 \\
\hline
a & b & 0 & 0 & 0 \\
a & b & 0 & 0 & 0 \\
a & b & 0 & 0 & 0
\end{array}\right]
$$

$R_{ij} = \mathbb{E}_i N_j$

$$
R = \begin{array}{c} \\ 3 \\ 4 \\ 1 \\ 2 \\ 5 \end{array}
\begin{array}{c}
\begin{array}{ccccc} 3 & 4 & 1 & 2 & 5 \end{array} \\
\left[\begin{array}{cc|ccc}
\infty & \infty & 0 & 0 & 0 \\
\infty & \infty & 0 & 0 & 0 \\
\hline
\infty & \infty & & & \\
\infty & \infty & & \text{finite} & \\
\infty & \infty & & &
\end{array}\right]
\end{array}
$$

Computation:

$$
R_{ij} = \mathbb{E}_i N_j = \mathbb{E}_i \sum_{n=0}^{\infty} 1_{\{X_n = j\}} = \sum_{n=0}^{\infty} \mathbb{E}_i 1_{\{X_n = j\}} = \sum_{n=0}^{\infty} \mathbb{P}_i (X_n = j) = \sum_{n=0}^{\infty} P^n_{ij}
$$

Thus,

$$
R = I + P + P^2 + P^3 + P^4 + \cdots \quad \text{— Matlab!}
$$

P =

$$\begin{array}{c|ccccc|}
 & 3 & 4 & 1 & 2 & 5 \\
3 & 0.3000 & 0.7000 & 0 & 0 & 0 \\
4 & 0.6000 & 0.4000 & 0 & 0 & 0 \\
1 & 0 & 0 & 0.2000 & 0.7000 & 0.1000 \\
2 & 0.2000 & 0 & 0.3000 & 0.2000 & 0.3000 \\
5 & 0 & 0 & 0.6000 & 0.4000 & 0
\end{array}$$

R =

1.0e+03 *

$$\begin{array}{c|ccccc|}
 & 3 & 4 & 1 & 2 & 5 \\
3 & 0.9244 & 1.0776 & 0 & 0 & 0 \\
4 & 0.9236 & 1.0784 & 0 & 0 & 0 \\
1 & 0.9191 & 1.0714 & 0.0046 & 0.0050 & 0.0020 \\
2 & 0.9198 & 1.0722 & 0.0032 & 0.0050 & 0.0018 \\
5 & 0.9189 & 1.0711 & 0.0041 & 0.0050 & 0.0029
\end{array}$$

»R(3:5,3:5)

ans =

$$\begin{array}{c|ccc|}
 & 1 & 2 & 5 \\
1 & 4.5946 & 5.0000 & 1.9595 \\
2 & 3.2432 & 5.0000 & 1.8243 \\
5 & 4.0541 & 5.0000 & 2.9054
\end{array}$$

Computing the expected number of visits numerically

$$R_{ij} = \mathbb{E}_i N_j = \mathbb{E}_i \sum_{n=0}^{\infty} 1_{\{X_n = j\}} = \sum_{n=0}^{\infty} P^n_{ij}$$

$$R = \sum_{n=0}^{\infty} P^n = I + P + P^2 + \cdots$$

Numerically, though, you can only sum a few thousand terms. Some entries in R should be ∞, but they won't be.

Classification of states:

Divide S into closed, recurrent communicating classes $C_1, C_2, \ldots, C_m$ and group the left-over states into a set t; these are the transient states.

In block form,

$$P = \begin{array}{c|c} & \begin{array}{ccccc} C_1 & C_2 & \cdots & C_m & t \end{array} \\ \hline \begin{array}{c} C_1 \\ C_2 \\ \vdots \\ C_m \\ t \end{array} & \left[\begin{array}{ccccc} * & & & & \\ & * & & & \\ & & & & \\ & & & * & \\ * & * & \cdots & * & * \end{array}\right] \end{array}$$

where $*$ indicates possible nonzero entries.

Then,

$$R = \left[\begin{array}{ccccc} \infty & & & & \\ & & & & \\ & & & & \\ & & & \infty & \\ \infty \text{ or } 0 & & & \infty \text{ or } 0 & \underbrace{W}_{\text{finite, p. 171}} \end{array}\right]$$

The only thing we don't know is the expected number of visits to the transient states.

Let's focus on computing that.

(This is similar to III.7 in Taylor and Karlin.)

Group all recurrent states into a set called r.

Then we may write

$$P = \begin{array}{c c} & \begin{array}{cc} r & t \end{array} \\ \begin{array}{c} r \\ t \end{array} & \left[\begin{array}{c|c} K & O \\ \hline L & Q \end{array}\right] \end{array}$$

Then

$$P^2 = \begin{bmatrix} K & O \\ L & Q \end{bmatrix}\begin{bmatrix} K & O \\ L & Q \end{bmatrix} = \begin{bmatrix} K^2 & O \\ LK+QL & Q^2 \end{bmatrix} = \begin{bmatrix} K^2 & O \\ L_2 & Q^2 \end{bmatrix}$$

In the same way,

$$P^n = \begin{bmatrix} K^n & O \\ L_n & Q^n \end{bmatrix}$$

where L_n is some matrix.

To compute W, we need to sum the corresponding submatrix of $I, P, P^2, \ldots$

This is just:

$$\boxed{\begin{aligned} W &= I + Q + Q^2 + Q^3 + \cdots \\ W_{ij} &= \mathbb{E}_i N_j \quad \text{where } i \text{ and } j \text{ are transient} \end{aligned}}$$

Example Gambler's wealth

$S = \{0, 1, \ldots, 30\}$.

$\{0\}$, $\{30\}$ are the communicating classes of recurrent states

$r = \{0, 30\}$

$t = \{1, 2, \ldots, 29\}$ are the transient states

Define Q and compute W.

Matlab

$r = \{1, 31\}$ type `r = [1 31];`

$t = \{2, 3, \ldots, 30\}$. type `t = [2:30];`

```
Q = P(t,t) ;      % pull out this submatrix
W = potential(Q);   % my program
```

Time until absorbtion

Start in transient state i,

$$T = \sum_{j \in t} N_j$$

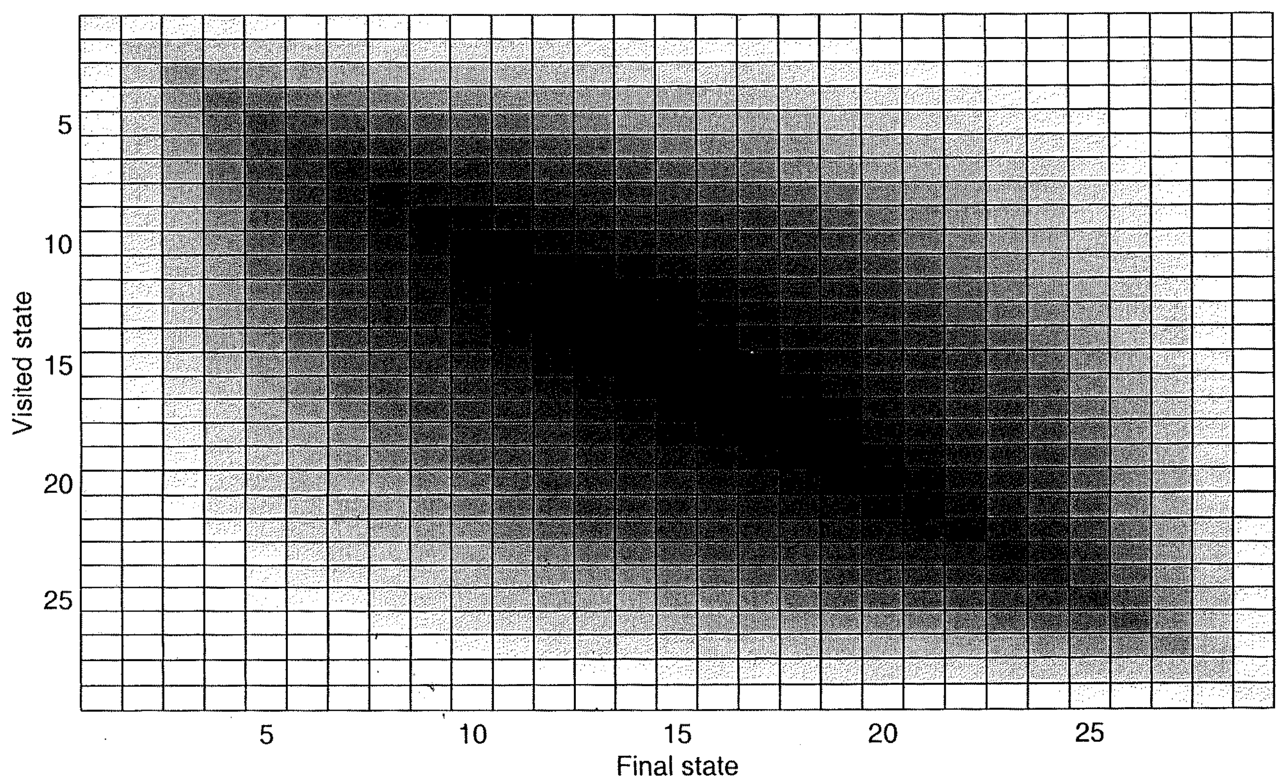
$$\mathbb{E}_i T = \sum_{j \in t} \mathbb{E}_i N_j = \sum_{j \in t} W_{ij} = \text{sum of row } i \text{ of } W.$$

Matlab `sum(W')'` to display as a column vector.

gambler_vis[illegible].m produces the matrix S for the transient states of the gambler's wea[illegible] states 1 to N-1.

1.9333	1.8667	1.8000	1.7333	1.6667	1.6000	1.5333	1.4667	1.4000	1.3333	1.2667	1.2000	1.1333	1.0667	1.0000	0.9333	0.8667	0.8000	0.7333	0.6667	0.6000	0.533
1.8667	3.7333	3.6000	3.4667	3.3333	3.2000	3.0667	2.9333	2.8000	2.6667	2.5333	2.4000	2.2667	2.1333	2.0000	1.8667	1.7333	1.6000	1.4667	1.3333	1.2000	1.066
1.8000	3.6000	5.4000	5.2000	5.0000	4.8000	4.6000	4.4000	4.2000	4.0000	3.8000	3.6000	3.4000	3.2000	3.0000	2.8000	2.6000	2.4000	2.2000	2.0000	1.8000	1.600
1.7333	3.4667	5.2000	6.9333	6.6667	6.4000	6.1333	5.8667	5.6000	5.3333	5.0667	4.8000	4.5333	4.2667	4.0000	3.7333	3.4667	3.2000	2.9333	2.6667	2.4000	2.133
1.6667	3.3333	5.0000	6.6667	8.3333	8.0000	7.6667	7.3333	7.0000	6.6667	6.3333	6.0000	5.6667	5.3333	5.0000	4.6667	4.3333	4.0000	3.6667	3.3333	3.0000	2.666
1.6000	3.2000	4.8000	6.4000	8.0000	9.6000	9.2000	8.8000	8.4000	8.0000	7.6000	7.2000	6.8000	6.4000	6.0000	5.6000	5.2000	4.8000	4.4000	4.0000	3.6000	3.200
1.5333	3.0667	4.6000	6.1333	7.6667	9.2000	10.7333	10.2667	9.8000	9.3333	8.8667	8.4000	7.9333	7.4667	7.0000	6.5333	6.0667	5.6000	5.1333	4.6667	4.2000	3.733
1.4667	2.9333	4.4000	5.8667	7.3333	8.8000	10.2667	11.7333	11.2000	10.6667	10.1333	9.6000	9.0667	8.5333	8.0000	7.4667	6.9333	6.4000	5.8667	5.3333	4.8000	4.266
1.4000	2.8000	4.2000	5.6000	7.0000	8.4000	9.8000	11.2000	12.6000	12.0000	11.4000	10.8000	10.2000	9.6000	9.0000	8.4000	7.8000	7.2000	6.6000	6.0000	5.4000	4.800
1.3333	2.6667	4.0000	5.3333	6.6667	8.0000	9.3333	10.6667	12.0000	13.3333	12.6667	12.0000	11.3333	10.6667	10.0000	9.3333	8.6667	8.0000	7.3333	6.6667	6.0000	5.333
1.2667	2.5333	3.8000	5.0667	6.3333	7.6000	8.8667	10.1333	11.4000	12.6667	13.9333	13.2000	12.4667	11.7333	11.0000	10.2667	9.5333	8.8000	8.0667	7.3333	6.6000	5.866
1.2000	2.4000	3.6000	4.8000	6.0000	7.2000	8.4000	9.6000	10.8000	12.0000	13.2000	14.4000	13.6000	12.8000	12.0000	11.2000	10.4000	9.6000	8.8000	8.0000	7.2000	6.400
1.1333	2.2667	3.4000	4.5333	5.6667	6.8000	7.9333	9.0667	10.2000	11.3333	12.4667	13.6000	14.7333	13.8667	13.0000	12.1333	11.2667	10.4000	9.5333	8.6667	7.8000	6.933
1.0667	2.1333	3.2000	4.2667	5.3333	6.4000	7.4667	8.5333	9.6000	10.6667	11.7333	12.8000	13.8667	14.9333	14.0000	13.0667	12.1333	11.2000	10.2667	9.3333	8.4000	7.466
1.0000	2.0000	3.0000	4.0000	5.0000	6.0000	7.0000	8.0000	9.0000	10.0000	11.0000	12.0000	13.0000	14.0000	15.0000	14.0000	13.0000	12.0000	11.0000	10.0000	9.0000	8.000
0.9333	1.8667	2.8000	3.7333	4.6667	5.6000	6.5333	7.4667	8.4000	9.3333	10.2667	11.2000	12.1333	13.0667	14.0000	14.9333	13.8667	12.8000	11.7333	10.6667	9.6000	8.533
0.8667	1.7333	2.6000	3.4667	4.3333	5.2000	6.0667	6.9333	7.8000	8.6667	9.5333	10.4000	11.2667	12.1333	13.0000	13.8667	14.7333	13.6000	12.4667	11.3333	10.2000	9.066
0.8000	1.6000	2.4000	3.2000	4.0000	4.8000	5.6000	6.4000	7.2000	8.0000	8.8000	9.6000	10.4000	11.2000	12.0000	12.8000	13.6000	14.4000	13.2000	12.0000	10.8000	9.600
0.7333	1.4667	2.2000	2.9333	3.6667	4.4000	5.1333	5.8667	6.6000	7.3333	8.0667	8.8000	9.5333	10.2667	11.0000	11.7333	12.4667	13.2000	13.9333	12.6667	11.4000	10.133
0.6667	1.3333	2.0000	2.6667	3.3333	4.0000	4.6667	5.3333	6.0000	6.6667	7.3333	8.0000	8.6667	9.3333	10.0000	10.6667	11.3333	12.0000	12.6667	13.3333	12.0000	10.666
0.6000	1.2000	1.8000	2.4000	3.0000	3.6000	4.2000	4.8000	5.4000	6.0000	6.6000	7.2000	7.8000	8.4000	9.0000	9.6000	10.2000	10.8000	11.4000	12.0000	12.6000	11.200
0.5333	1.0667	1.6000	2.1333	2.6667	3.2000	3.7333	4.2667	4.8000	5.3333	5.8667	6.4000	6.9333	7.4667	8.0000	8.5333	9.0667	9.6000	10.1333	10.6667	11.2000	11.733
0.4667	0.9333	1.4000	1.8667	2.3333	2.8000	3.2667	3.7333	4.2000	4.6667	5.1333	5.6000	6.0667	6.5333	7.0000	7.4667	7.9333	8.4000	8.8667	9.3333	9.8000	10.266
0.4000	0.8000	1.2000	1.6000	2.0000	2.4000	2.8000	3.2000	3.6000	4.0000	4.4000	4.8000	5.2000	5.6000	6.0000	6.4000	6.8000	7.2000	7.6000	8.0000	8.4000	8.800
0.3333	0.6667	1.0000	1.3333	1.6667	2.0000	2.3333	2.6667	3.0000	3.3333	3.6667	4.0000	4.3333	4.6667	5.0000	5.3333	5.6667	6.0000	6.3333	6.6667	7.0000	7.333
0.2667	0.5333	0.8000	1.0667	1.3333	1.6000	1.8667	2.1333	2.4000	2.6667	2.9333	3.2000	3.4667	3.7333	4.0000	4.2667	4.5333	4.8000	5.0667	5.3333	5.6000	5.866
0.2000	0.4000	0.6000	0.8000	1.0000	1.2000	1.4000	1.6000	1.8000	2.0000	2.2000	2.4000	2.6000	2.8000	3.0000	3.2000	3.4000	3.6000	3.8000	4.0000	4.2000	4.400
0.1333	0.2667	0.4000	0.5333	0.6667	0.8000	0.9333	1.0667	1.2000	1.3333	1.4667	1.6000	1.7333	1.8667	2.0000	2.1333	2.2667	2.4000	2.5333	2.6667	2.8000	2.933
0.0667	0.1333	0.2000	0.2667	0.3333	0.4000	0.4667	0.5333	0.6000	0.6667	0.7333	0.8000	0.8667	0.9333	1.0000	1.0667	1.1333	1.2000	1.2667	1.3333	1.4000	1.466

Graphical representation of expected number of visits matrix S



Example HHT

$$P = \begin{array}{c} O \\ H \\ HH \\ HHT \end{array} \begin{bmatrix} \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ \frac{1}{2} & 0 & \frac{1}{2} & 0 \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} & 0 & 0 \end{bmatrix} \quad \text{or} \quad \begin{bmatrix} & & & \\ & & & \\ & & & \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

For time until absorbtion in HHT,

```
r = [4];
t = [1 2 3];
Q = P(t,t);
W = potential(Q);
```

Notice: row 4 of P does not matter!

simply identify the "transient" states.

All others may be considered to be absorbing.

$$P = \begin{array}{c c} & \begin{array}{cc} r & t \end{array} \\ \begin{array}{c} r \\ t \end{array} & \begin{bmatrix} K & O \\ L & Q \end{bmatrix} \end{array}$$

$$W_{ij} = \mathbb{E}_i N_j \qquad \text{where } i, j \text{ are transient}$$

$$W = I + Q + Q^2 + \cdots$$

Numerically, one can approximate this infinite sum by adding a few thousand terms, or, observe this:

$$\begin{aligned} QW &= Q + Q^2 + Q^3 + \cdots \\ &= W - I \\ I &= W - QW \\ &= (I - Q)W, \end{aligned}$$

so $W = (I - Q)^{-1}$.

The inverse <u>must</u> exist because W exists and is finite valued.

Matlab

potential.m does the sum

transientvisits.m does the inverse. Very easy.

Probability of ever hitting a given transient state

Start the chain in a transient state i.
What is the probability of ever hitting state j?

Notation: $f_{ij} = \mathbb{P}_i(X_n = j \text{ for some } j \geq 1)$ from before.

Recall: Starting at i, N_i is geometric with success probability $1 - f_{ii}$,

so $W_{ii} = \mathbb{E}_i N_i = \dfrac{1}{1 - f_{ii}}$

Solve for f_{ii}: $\boxed{f_{ii} = 1 - \dfrac{1}{W_{ii}}}$

Similarly, if i and j are transient,

$$W_{ij} = \mathbb{E}_i N_j = \frac{f_{ij}}{1 - f_{jj}} = f_{ij} W_{jj}$$

$$\boxed{f_{ij} = \frac{W_{ij}}{W_{jj}}}$$

Numerics

$$f_{ii} = \frac{W_{ii} - 1}{W_{ii}}$$

$$f_{ij} = \frac{W_{ij}}{W_{jj}}$$

$f = W - I$, then divide each column by the corresponding diagonal entry of W.

$= (W - I)\, \mathrm{diag}(W)^{-1}$

Matlab

```
f = (W - eye(size(W))) * inv(diag(diag(W)));
```

* This is a better way to do 2d on the homework

Probability of hitting recurrent states

If there are only finitely many transient states, the chain will eventually hit a recurrent state and never return to the transient states.

We can compute the probabilities with which each recurrent state is hit first.

Method: pretend that each recurrent state is absorbing, look at $\lim_{n\to\infty} P^n$.

$$P = \begin{bmatrix} K & 0 \\ L & Q \end{bmatrix}$$

could just do this numerically, but the theory is nicer.

$$\hat{P} = \begin{bmatrix} I & 0 \\ L & Q \end{bmatrix}$$

* pretend absorbing

$$\hat{P}^2 = \begin{bmatrix} I & 0 \\ L & Q \end{bmatrix}\begin{bmatrix} I & 0 \\ L & Q \end{bmatrix} = \begin{bmatrix} I & 0 \\ L+QL & Q^2 \end{bmatrix}$$

$$\hat{P}^3 = \begin{bmatrix} I & 0 \\ L+QL & Q^2 \end{bmatrix}\begin{bmatrix} I & 0 \\ L & Q \end{bmatrix} = \begin{bmatrix} I & 0 \\ L+QL+Q^2L & Q^3 \end{bmatrix} = \begin{bmatrix} I & 0 \\ (I+Q+Q^2)L & Q^3 \end{bmatrix}$$

$$\lim_{n\to\infty} \hat{P}^n = \begin{bmatrix} I & 0 \\ (I+Q+Q^2+\cdots)L & 0 \end{bmatrix} = \begin{bmatrix} I & 0 \\ WL & 0 \end{bmatrix}$$

Summary

Let $F_{ij} = \mathbb{P}_i(X_n = j \text{ for } 1 \le n \le T)$, where $T = \min(n : X_n \in r)$, is the time that a recurrent state is first hit.

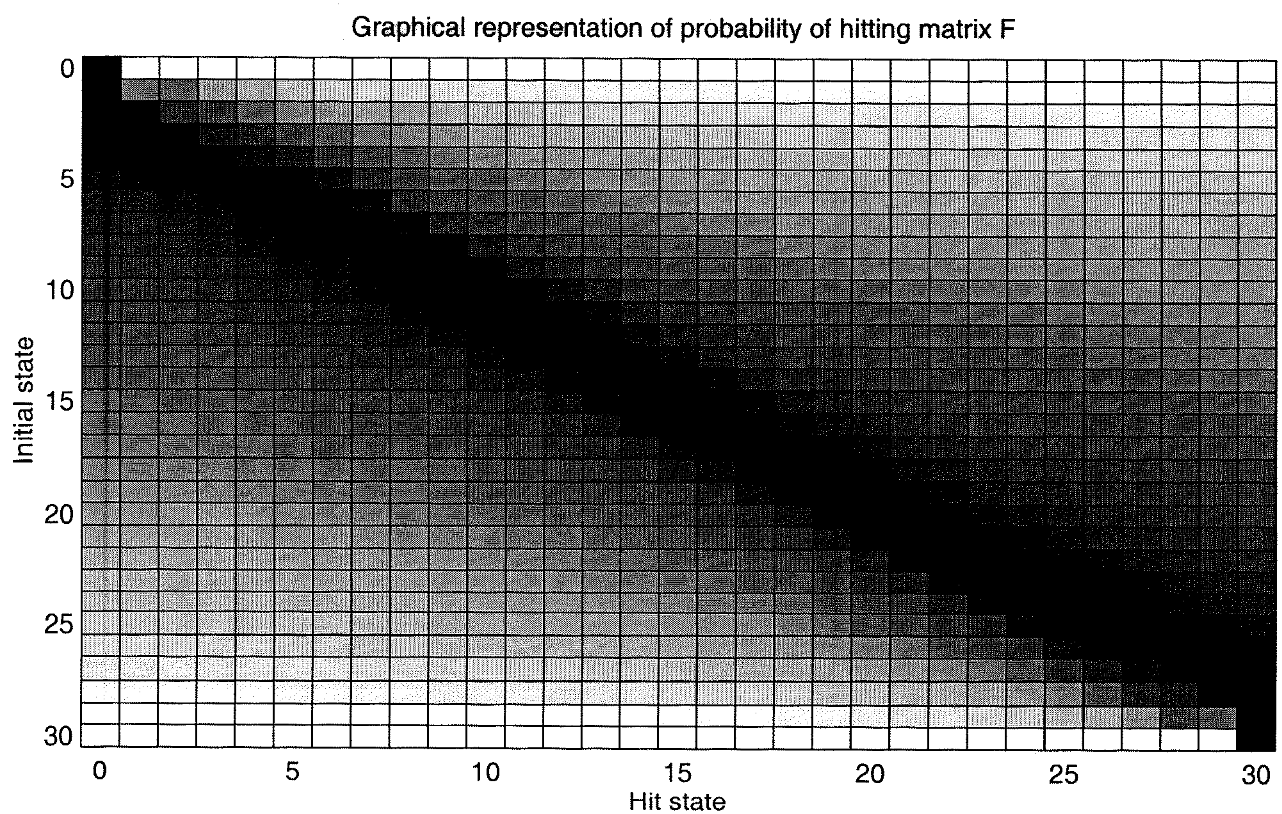
Then

$$F = \begin{array}{cc} & \begin{array}{cc} r & t \end{array} \\ \begin{array}{c} r \\ t \end{array} & \left[\begin{array}{c|c} I & 0 \\ \hline WL & (W-I)(\operatorname{diag} W)^{-1} \end{array}\right] \end{array}$$

```
% gambler_visits.m displays the expected number of visits to transient
% states matrix W, the expected length of the game, and the probability of
% hitting matrix F

% run gambler.m first

r = [1 N+1];              % recurrent states
t = [2:N];                % transient states
Q = P(t,t);               % pull out the matrix corresponding to transient states
L = P(t,r);               % transitions from transient to recurrent states

W = inv(eye(size(Q))-Q); % compute the expected number of visits to states in t

m = sum(W')';             % expected number of visits before the end of the game

F(r,r) = eye(size(P(r,r)));
F(r,t) = zeros(size(L'));
F(t,r) = W*L;
F(t,t) = (W-eye(size(W)))*inv(diag(diag(W)));

clf;

subplot(2,1,1);

graph_matrix(1,N-1,W);
title('Graphical representation of expected number of visits matrix W');
ylabel('Initial state');
xlabel('Visited state');

subplot(2,1,2);

graph_matrix(0,N,F);
title('Graphical representation of probability of hitting matrix F');
ylabel('Initial state');
xlabel('Hit state');

fprintf('Expected length of game, as a function of initial wealth:')
[(1:(N-1))' m]
```

gambler_visits.m also c…utes the probability of eventually hitting matrix F, shown below

1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0.9667	0.4828	0.5000	0.3333	0.2500	0.2000	0.1667	0.1429	0.1250	0.1111	0.1000	0.0909	0.0833	0.0769	0.0714	0.0667	0.0625	0.0588	0.0556	0.0526	0.0500	0.0476	0.0455	0.0435	0.0417	0.0400	0.0385	0.0370	0.0357	0.0345	0.0333
0.9333	0.9655	0.7321	0.6667	0.5000	0.4000	0.3333	0.2857	0.2500	0.2222	0.2000	0.1818	0.1667	0.1538	0.1429	0.1333	0.1250	0.1176	0.1111	0.1053	0.1000	0.0952	0.0909	0.0870	0.0833	0.0800	0.0769	0.0741	0.0714	0.0690	0.0667
0.9000	0.9310	0.9643	0.8148	0.7500	0.6000	0.5000	0.4286	0.3750	0.3333	0.3000	0.2727	0.2500	0.2308	0.2143	0.2000	0.1875	0.1765	0.1667	0.1579	0.1500	0.1429	0.1364	0.1304	0.1250	0.1200	0.1154	0.1111	0.1071	0.1034	0.1000
0.8667	0.8966	0.9286	0.9630	0.8558	0.8000	0.6667	0.5714	0.5000	0.4444	0.4000	0.3636	0.3333	0.3077	0.2857	0.2667	0.2500	0.2353	0.2222	0.2105	0.2000	0.1905	0.1818	0.1739	0.1667	0.1600	0.1538	0.1481	0.1429	0.1379	0.1333
0.8333	0.8621	0.8929	0.9259	0.9615	0.8800	0.8333	0.7143	0.6250	0.5556	0.5000	0.4545	0.4167	0.3846	0.3571	0.3333	0.3125	0.2941	0.2778	0.2632	0.2500	0.2381	0.2273	0.2174	0.2083	0.2000	0.1923	0.1852	0.1786	0.1724	0.1667
0.8000	0.8276	0.8571	0.8889	0.9231	0.9600	0.8958	0.8571	0.7500	0.6667	0.6000	0.5455	0.5000	0.4615	0.4286	0.4000	0.3750	0.3529	0.3333	0.3158	0.3000	0.2857	0.2727	0.2609	0.2500	0.2400	0.2308	0.2222	0.2143	0.2069	0.2000
0.7667	0.7931	0.8214	0.8519	0.8846	0.9200	0.9583	0.9068	0.8750	0.7778	0.7000	0.6364	0.5833	0.5385	0.5000	0.4667	0.4375	0.4118	0.3889	0.3684	0.3500	0.3333	0.3182	0.3043	0.2917	0.2800	0.2692	0.2593	0.2500	0.2414	0.2333
0.7333	0.7586	0.7857	0.8148	0.8462	0.8800	0.9167	0.9565	0.9148	0.8889	0.8000	0.7273	0.6667	0.6154	0.5714	0.5333	0.5000	0.4706	0.4444	0.4211	0.4000	0.3810	0.3636	0.3478	0.3333	0.3200	0.3077	0.2963	0.2857	0.2759	0.2667
0.7000	0.7241	0.7500	0.7778	0.8077	0.8400	0.8750	0.9130	0.9545	0.9206	0.9000	0.8182	0.7500	0.6923	0.6429	0.6000	0.5625	0.5294	0.5000	0.4737	0.4500	0.4286	0.4091	0.3913	0.3750	0.3600	0.3462	0.3333	0.3214	0.3103	0.3000
0.6667	0.6897	0.7143	0.7407	0.7692	0.8000	0.8333	0.8696	0.9091	0.9524	0.9250	0.9091	0.8333	0.7692	0.7143	0.6667	0.6250	0.5882	0.5556	0.5263	0.5000	0.4762	0.4545	0.4348	0.4167	0.4000	0.3846	0.3704	0.3571	0.3448	0.3333
0.6333	0.6552	0.6786	0.7037	0.7308	0.7600	0.7917	0.8261	0.8636	0.9048	0.9500	0.9282	0.9167	0.8462	0.7857	0.7333	0.6875	0.6471	0.6111	0.5789	0.5500	0.5238	0.5000	0.4783	0.4583	0.4400	0.4231	0.4074	0.3929	0.3793	0.3667
0.6000	0.6207	0.6429	0.6667	0.6923	0.7200	0.7500	0.7826	0.8182	0.8571	0.9000	0.9474	0.9306	0.9231	0.8571	0.8000	0.7500	0.7059	0.6667	0.6316	0.6000	0.5714	0.5455	0.5217	0.5000	0.4800	0.4615	0.4444	0.4286	0.4138	0.4000
0.5667	0.5862	0.6071	0.6296	0.6538	0.6800	0.7083	0.7391	0.7727	0.8095	0.8500	0.8947	0.9444	0.9321	0.9286	0.8667	0.8125	0.7647	0.7222	0.6842	0.6500	0.6190	0.5909	0.5652	0.5417	0.5200	0.5000	0.4815	0.4643	0.4483	0.4333
0.5333	0.5517	0.5714	0.5926	0.6154	0.6400	0.6667	0.6957	0.7273	0.7619	0.8000	0.8421	0.8889	0.9412	0.9330	0.9333	0.8750	0.8235	0.7778	0.7368	0.7000	0.6667	0.6364	0.6087	0.5833	0.5600	0.5385	0.5185	0.5000	0.4828	0.4667
0.5000	0.5172	0.5357	0.5556	0.5769	0.6000	0.6250	0.6522	0.6818	0.7143	0.7500	0.7895	0.8333	0.8824	0.9375	0.9333	0.9375	0.8824	0.8333	0.7895	0.7500	0.7143	0.6818	0.6522	0.6250	0.6000	0.5769	0.5556	0.5357	0.5172	0.5000
0.4667	0.4828	0.5000	0.5185	0.5385	0.5600	0.5833	0.6087	0.6364	0.6667	0.7000	0.7368	0.7778	0.8235	0.8750	0.9333	0.9330	0.9412	0.8889	0.8421	0.8000	0.7619	0.7273	0.6957	0.6667	0.6400	0.6154	0.5926	0.5714	0.5517	0.5333
0.4333	0.4483	0.4643	0.4815	0.5000	0.5200	0.5417	0.5652	0.5909	0.6190	0.6500	0.6842	0.7222	0.7647	0.8125	0.8667	0.9286	0.9321	0.9444	0.8947	0.8500	0.8095	0.7727	0.7391	0.7083	0.6800	0.6538	0.6296	0.6071	0.5862	0.5667
0.4000	0.4138	0.4286	0.4444	0.4615	0.4800	0.5000	0.5217	0.5455	0.5714	0.6000	0.6316	0.6667	0.7059	0.7500	0.8000	0.8571	0.9231	0.9306	0.9474	0.9000	0.8571	0.8182	0.7826	0.7500	0.7200	0.6923	0.6667	0.6429	0.6207	0.6000
0.3667	0.3793	0.3929	0.4074	0.4231	0.4400	0.4583	0.4783	0.5000	0.5238	0.5500	0.5789	0.6111	0.6471	0.6875	0.7333	0.7857	0.8462	0.9167	0.9282	0.9500	0.9048	0.8636	0.8261	0.7917	0.7600	0.7308	0.7037	0.6786	0.6552	0.6333
0.3333	0.3448	0.3571	0.3704	0.3846	0.4000	0.4167	0.4348	0.4545	0.4762	0.5000	0.5263	0.5556	0.5882	0.6250	0.6667	0.7143	0.7692	0.8333	0.9091	0.9250	0.9524	0.9091	0.8696	0.8333	0.8000	0.7692	0.7407	0.7143	0.6897	0.6667
0.3000	0.3103	0.3214	0.3333	0.3462	0.3600	0.3750	0.3913	0.4091	0.4286	0.4500	0.4737	0.5000	0.5294	0.5625	0.6000	0.6429	0.6923	0.7500	0.8182	0.9000	0.9206	0.9545	0.9130	0.8750	0.8400	0.8077	0.7778	0.7500	0.7241	0.7000
0.2667	0.2759	0.2857	0.2963	0.3077	0.3200	0.3333	0.3478	0.3636	0.3810	0.4000	0.4211	0.4444	0.4706	0.5000	0.5333	0.5714	0.6154	0.6667	0.7273	0.8000	0.8889	0.9148	0.9565	0.9167	0.8800	0.8462	0.8148	0.7857	0.7586	0.7333
0.2333	0.2414	0.2500	0.2593	0.2692	0.2800	0.2917	0.3043	0.3182	0.3333	0.3500	0.3684	0.3889	0.4118	0.4375	0.4667	0.5000	0.5385	0.5833	0.6364	0.7000	0.7778	0.8750	0.9068	0.9583	0.9200	0.8846	0.8519	0.8214	0.7931	0.7667
0.2000	0.2069	0.2143	0.2222	0.2308	0.2400	0.2500	0.2609	0.2727	0.2857	0.3000	0.3158	0.3333	0.3529	0.3750	0.4000	0.4286	0.4615	0.5000	0.5455	0.6000	0.6667	0.7500	0.8571	0.8958	0.9600	0.9231	0.8889	0.8571	0.8276	0.8000
0.1667	0.1724	0.1786	0.1852	0.1923	0.2000	0.2083	0.2174	0.2273	0.2381	0.2500	0.2632	0.2778	0.2941	0.3125	0.3333	0.3571	0.3846	0.4167	0.4545	0.5000	0.5556	0.6250	0.7143	0.8333	0.8800	0.9615	0.9259	0.8929	0.8621	0.8333
0.1333	0.1379	0.1429	0.1481	0.1538	0.1600	0.1667	0.1739	0.1818	0.1905	0.2000	0.2105	0.2222	0.2353	0.2500	0.2667	0.2857	0.3077	0.3333	0.3636	0.4000	0.4444	0.5000	0.5714	0.6667	0.8000	0.8558	0.9630	0.9286	0.8966	0.8667
0.1000	0.1034	0.1071	0.1111	0.1154	0.1200	0.1250	0.1304	0.1364	0.1429	0.1500	0.1579	0.1667	0.1765	0.1875	0.2000	0.2143	0.2308	0.2500	0.2727	0.3000	0.3333	0.3750	0.4286	0.5000	0.6000	0.7500	0.8148	0.9643	0.9310	0.9000
0.0667	0.0690	0.0714	0.0741	0.0769	0.0800	0.0833	0.0870	0.0909	0.0952	0.1000	0.1053	0.1111	0.1176	0.1250	0.1333	0.1429	0.1538	0.1667	0.1818	0.2000	0.2222	0.2500	0.2857	0.3333	0.4000	0.5000	0.6667	0.7321	0.9655	0.9333
0.0333	0.0345	0.0357	0.0370	0.0385	0.0400	0.0417	0.0435	0.0455	0.0476	0.0500	0.0526	0.0556	0.0588	0.0625	0.0667	0.0714	0.0769	0.0833	0.0909	0.1000	0.1111	0.1250	0.1429	0.1667	0.2000	0.2500	0.3333	0.5000	0.4828	0.9667
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000



Computations for irreducible recurrent chains

We have seen how to compute the expected number of visits to transient states and the probability of ever visiting certain transient states.

The same methods may be applied to answer questions concerning the behavior of an irreducible recurrent Markov chain.

Example

Starting in state i, find the expected number of visits to j before state k is hit.

Pretend that k is absorbing, thus all other states are transient.

```
r = [k];
t = [1:(k-1) (k+1):N];
Q = P(t,t);             % P(r,r) is immaterial!
W = inv(I - Q);
W(i,j)                  % what you want.
```

Example

Starting in state i, find the prob. that j is hit before k.

```
r = [j k];              % pretend these are absorbing
⋮
F(i,j)  % is what you want.
```

Total cost while in transient states

Start in a transient state i.

Cost $c(j)$ for being in transient state j.

What is the expected total cost until the chain leaves the transient states?

For simplicity, let $\bar{c}(j) = \begin{cases} c(j) & \text{if } j \text{ is transient} \\ 0 & \text{if } j \text{ is recurrent} \end{cases}$

$\mathbb{E}_i$ cost until a recurrent state is hit

$$= \mathbb{E}_i \sum_{n=0}^{\infty} \bar{c}(X_n)$$

$$= \mathbb{E}_i \sum_{n=0}^{\infty} \sum_{j \in S} 1_j(X_n)\, \bar{c}(X_n)$$

$$= \mathbb{E}_i \sum_{n=0}^{\infty} \sum_{j \in S} 1_j(X_n)\, \bar{c}(j)$$

$$= \sum_{j \in S} \left(\mathbb{E}_i \sum_{n=0}^{\infty} 1_j(X_n) \right) \bar{c}(j)$$

$$= \sum_{j \in S} \mathbb{E}_i N_j\, \bar{c}(j)$$

$$= \sum_{j \in S} R_{ij}\, \bar{c}(j)$$

$$= \sum_{j \in t} W_{ij}\, c(j)$$

$$= Wc,$$

where W is the expected number of visits to transient states.

This is quite easy to do, really.

Example Roll dice until total hits 100.
Cost for hitting prime numbers is 5
Cost for hitting composite numbers is -2; a reward, really.
What is the expected reward?