



2.

	P	Q	R	← previous
P =	P	Q	R	
	0.6	0.1	0.1	
	0.1	0.8	0.2	
	0.3	0.1	0.7	
	↑			
	current			

Transition matrix $P = (P_{ij})_{3 \times 3}$
 Every column of probability should ~~be~~ ^{add} to one

3. $P^{(0)} = (100, 150, 200)^T$

P =	P	Q	R	previous	$X_0 = \begin{bmatrix} 100 \\ 150 \\ 200 \end{bmatrix}$
	P	Q	R		
	0.6	0.1	0.1		
	0.1	0.8	0.2		
	0.3	0.1	0.7		
	↑				
	current				

$$X_1 = P' X_0 = \begin{bmatrix} 0.6 & 0.1 & 0.1 \\ 0.1 & 0.8 & 0.2 \\ 0.3 & 0.1 & 0.7 \end{bmatrix} \begin{bmatrix} 100 \\ 150 \\ 200 \end{bmatrix} = \begin{bmatrix} 95 \\ 170 \\ 185 \end{bmatrix}$$

$$\therefore P = 0.6 \times 100 + 0.1 \times 150 + 0.1 \times 200 = 60 + 15 + 20 = 95$$

$$Q = 0.1 \times 100 + 0.8 \times 150 + 0.2 \times 200 = 10 + 120 + 40 = 170$$

$$R = 0.3 \times 100 + 0.1 \times 150 + 0.7 \times 200 = 30 + 15 + 140 = 185$$

After one week, the matrix only operate once, from previous stats, we know 0.6P will return to P, 0.1Q will go to P, 0.1R will go to P; 0.1P will go to Q, 0.8Q will return Q, 0.2R will go to Q; and 0.3P will go to Q, 0.1Q will go to Q, 0.7R will return Q.

Initially, P has 100 cars, Q has 150, R has 200, we can know After a week, 95 cars will be in P, 170 cars in Q, and 185 cars in R.

H6-4

Lu Lu

5/3/2020

```
#H6-4
#4. Suppose that, initially, a randomly selected car is equally \
likely to be at any of the three locations.
#(a) What is the initial probability vector p(0)? Be careful not to \
round here. Rounding errors will be compounded when you compute \
long term behavior.
print '1) Initially car is equally be ast any of three locations, \
the initial probability p(0) will be: '
p=matrix(3, 1, [1/3, 1/3, 1/3])
print p
1) Initially car is equally be ast any of three locations, the initial probability p(0)
will be:
[1/3]
[1/3]
[1/3]

#(b) What are the probabilities that this randomly selected car will \
be at each of the three locations after one week? How about after \
two weeks? Three weeks? As the number of weeks becomes very \
large?
print 'The transition matrix P is: '
P=matrix(3,3,[3/5, 1/10, 1/10, 1/10, 4/5, 1/5, 3/10, 1/10, 7/10])
print P
print
print 'From the markov Chain,  $p(t) = P^t * p(0)$  where t is the \
transition time.'
print 'After one week, t=1, the probability p(1) is: '
print P*p
print
print 'After two week, t=2, the probability p(2) is: '
print P^2 *p
print
print 'After three week, t=3, the probability p(3) is: '
print P^3 *p
print
```

```

print 'After a thousand week, t=1000, the probability p(1000) is: '
print P^1000 *p
print
print 'After two thousand, t=2000, the probability p(2000) is: '
print P^2000 *p
print
print 'From the previous matrix, we can see that in a long term, the\
      probability will close to identical as p = '
pFinal=matrix(3,1,[0.20 ,0.45, 0.35])
print pFinal

```

The transition matrix P is:

```

[ 3/5 1/10 1/10]
[1/10 4/5 1/5]
[3/10 1/10 7/10]

```

From the markov Chain, $p(t) = P^t * p(0)$ where t is the transition time.

After one week, t=1, the probability p(1) is:

```

[0.2250000000000000]
[0.4750000000000000]
[0.3000000000000000]

```

After two week, t=2, the probability p(2) is:

```

[0.2125000000000000]
[0.4625000000000000]
[0.3250000000000000]

```

After three week, t=3, the probability p(3) is:

```

[0.2062500000000000]
[0.4562500000000000]
[0.3375000000000000]

```

After a thousand week, t=1000, the probability p(1000) is:

```

[0.2000000000000000]
[0.4500000000000000]
[0.3500000000000000]

```

After two thousand, t=2000, the probability p(2000) is:

```

[0.2000000000000000]
[0.4500000000000000]
[0.3500000000000000]

```

From the previous matrix, we can see that in a long term, the probability will close to identical as p =

```

[0.2000000000000000]
[0.4500000000000000]
[0.3500000000000000]

```

#5. Repeat number 4, but this time assume that the initial randomly selected car has probability 0.25 of being in locationP, \ probability 0.5 of being in locationQ, and probability 0.25 of \ being in locationR. What did you discover?

```
p=matrix(3,1,[0.25, 0.5, 0.25])
print 'From the markov Chain,  $p(t) = P^t * p(0)$  where t is the \
transition time.'
print 'After one week, t=1, the probability p(1) is: '
print P*p
print
print 'After two week, t=2, the probability p(2) is: '
print P^2 *p
print
print 'After three week, t=3, the probability p(3) is: '
print P^3 *p
print
print 'After a thousand week, t=1000, the probability p(1000) is: '
print P^1000 *p
print
print 'After two thousand, t=2000, the probability p(2000) is: '
print P^2000 *p
print

print 'From the examples, we can discover that no matter what the \
initial probability matrix is, after a certain terms, it will \
close to an identical probability '
```

```
pFinal=matrix(3,1,[0.20 ,0.45, 0.35])
```

```
print pFinal
```

From the markov Chain, $p(t) = P^t * p(0)$ where t is the transition time.

After one week, t=1, the probability p(1) is:

```
[0.2250000000000000]
[0.4750000000000000]
[0.3000000000000000]
```

After two week, t=2, the probability p(2) is:

```
[0.2125000000000000]
[0.4625000000000000]
[0.3250000000000000]
```

After three week, t=3, the probability p(3) is:

```
[0.2062500000000000]
[0.4562500000000000]
[0.3375000000000000]
```

After a thousand week, t=1000, the probability p(1000) is:

```
[0.2000000000000000]
[0.4500000000000000]
```

[0.3500000000000000]

After two thousand, $t=2000$, the probability $p(2000)$ is:

[0.2000000000000000]

[0.4500000000000000]

[0.3500000000000000]

From the examples, we can discover that no matter what the initial probability matrix is, after a certain terms, it will close to an identical probability

[0.2000000000000000]

[0.4500000000000000]

[0.3500000000000000]

#6. Find the eigenvalues and eigenvectors of P . By experimenting on the computer, notice that P^n converges (as n gets large) to a matrix whose columns are all identical probability vectors. This common column vector is called a steady state vector. Explain how this happens for this particular Markov chain in terms of eigenvalues and eigenvectors. In particular, explain the relationship between one particular eigenvector and this steady state vector. (See notes MCp10-11.)

```
P=matrix(3,3,[3/5, 1/10, 1/10, 1/10, 4/5, 1/5, 3/10, 1/10, 7/10])
```

```
print 'The eigenvalues of P is ', P.eigenvalues(), '. The \
      eigenvectors of P is '
```

```
print P.eigenvectors_right()
```

```
print 'The steady state vector we find previously is '
```

```
print pFinal
```

```
print 'Notice, when eigen value is 1, the eigen vectors is [1, 9/4, \
      7/4], interpret to ratios sum to 1, it is [0.2, 0.45, 0.35] which \
      identically be the steady state vector. We can use the eigen \
      values and eigen vectors to determine the long term behavior of \
      MC'
```

The eigenvalues of P is $[1, 3/5, 1/2]$. The eigenvectors of P is

```
[(1, [
(1, 9/4, 7/4)
], 1), (3/5, [
(0, 1, -1)
], 1), (1/2, [
(1, 1, -2)
], 1)]
```

The steady state vector we find previously is

[0.2000000000000000]

[0.4500000000000000]

[0.3500000000000000]

Notice, when eigen value is 1, the eigen vectors is $[1, 9/4, 7/4]$, interpret to ratios sum to 1, it is $[0.2, 0.45, 0.35]$ which identically be the steady state vector. We can use the eigen values and eigen vectors to determine the long term behavior of MC

```
#7. In the context of this problem about automobiles, what does the \
    steady state vector say about the long term behavior of the Markov \
    chain when started with any initial probability vector p(0)?
print 'The steady state vector interpret that The steady state \
    vector tells that the long term behavior of the markov chain is \
    independent of the initial probability vector.'
print 'The steady state vector is the limit of the markov chain. \
    Regardless of the initial probability of car at each location, it \
    will finally reaching 20% of cars at P, 45% of cars at Q, 35% of \
    cars at R'
```

The steady state vector interpret that The steady state vector tells that the long term behavior of the markov chain is independent of the initial probability vector.

The steady state vector is the limit of the markov chain. Regardless of the initial probability of car at each location, it will finally reaching 20% of cars at P, 45% of cars at Q, 35% of cars at R