DaiPark Exercise chp3

Bong Seok Kim

2021-02-22

차 례

Exercise 5

Suppose each morning a factory posts the number of days worked in a row without any injuries. Assume that each day is injury free with probability 98/100. Furthermore, assume that whether tomorrow is injury free or not is independent of which of the preceding days were injury free. Let X_o = 0 be the morning the factory first opened. Let X_n be the number posted on the morning after n full days of work.

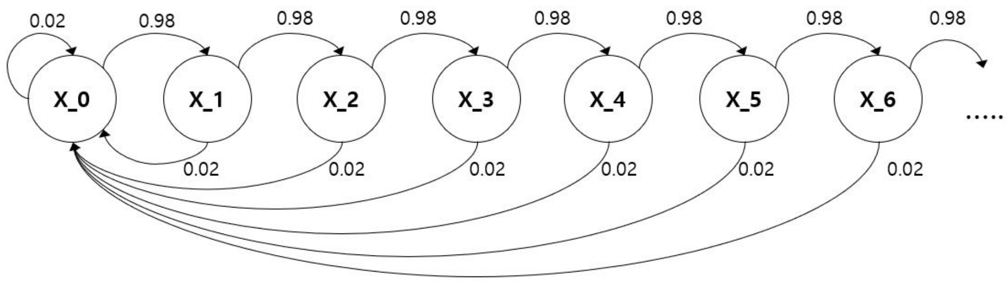

그림 1: markovchain

(a) Is $\{X_n, n \geq 0\}$ a Markov chain? If so, give its state space, initial distribution, and transition matrix P. If not, show that it is not a Markov chain.

yes, this is markov chain. beacuse Question assume that whether tomorrow is injury free or not is independent of which of the preceding days

state space= S={X_0,X_1,X_2,$\cdots$, X_n}.

initial distribution = X_0+X_1+$X_2$$\cdots$+$X_n$ = 1. (not sure)

trainsition matrix P

$$P = \begin{pmatrix} 0.02 & 0.98 & 0 & \cdots & 0 \\ 0.02 & 0 & 0.98 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0.02 & 0 & 0 & \cdots & 0 \end{pmatrix}$$

(b) Is the Markov chain irreducible? Explain.

=This markov chain irreducible. all States are communicate(<->) thus

there is a only one class.

(c) Is the Markov chain periodic or aperiodic? Explain and if it is peri- odic, also give the period.

(d) Find the stationary distribution.

find π that match, πP = π

$$\begin{aligned}
\pi_1 &= (0.98)\pi_0 \\
\pi_2 &= (0.98)\pi_1 = (0.98)^2\pi_0 \\
\pi_3 &= (0.98)\pi_2 = (0.98)^3\pi_0 \\
\pi_i &= (0.98)^i\pi_0
\end{aligned}$$

Thus (1), becomes

$$\begin{aligned}
\pi_0 + \pi_1 + \pi_2 + ... &= \pi_0(1 + 0.98 + 0.98^2 + ...) \\
\pi_0(\frac{1}{1-0.98})) &= 1 \\
\pi_0 &= 0.02 \\
\therefore \pi_i &= (0.02)(0.98)^i
\end{aligned}$$

$\therefore$, π = [0.02, 0.02(0.98), $0.02(0.98)^2$, ...]

(e) Is the Markov chain positive recurrent? If so, why? If not, why not?

no, it is inifinite Markovchain, like random walk

it can't come back again for sure

Exercise 6

Consider the following transition matrix:

$$P = \begin{pmatrix} 0 & 0.5 & 0 & 0.5 \\ 0.6 & 0 & 0.4 & 0 \\ 0 & 0.7 & 0 & 0.3 \\ 0.8 & 0 & 0.2 & 0 \end{pmatrix}$$

(a) Is the Markov chain periodic? Give the period of each state.

=periodic

Let's see in python

```
import numpy as np
from numpy.linalg import matrix_power

P=np.array([[0,0.5,0,0.5],
          [0.6,0,0.4,0],
          [0,0.7,0,0.3],
          [0.8,0,0.2,0]])

print(matrix_power(P,100))
```

```
## [[0.6875 0.     0.3125 0.    ]
##  [0.     0.5625 0.     0.4375]
##  [0.6875 0.     0.3125 0.    ]
##  [0.     0.5625 0.     0.4375]]
```

```
print(matrix_power(P,101))
```

```
## [[0.     0.5625 0.     0.4375]
##  [0.6875 0.     0.3125 0.    ]
##  [0.     0.5625 0.     0.4375]
##  [0.6875 0.     0.3125 0.    ]]
```

```
print(matrix_power(P,102))
```

```
## [[0.6875 0.     0.3125 0.    ]
##  [0.     0.5625 0.     0.4375]
##  [0.6875 0.     0.3125 0.    ]
##  [0.     0.5625 0.     0.4375]]
```

accordign to above result, it is obviously periodic

(b) Is $(\pi_1, \pi_2, \pi_3, \pi_4)$ = (33/96, 27/96, 15/96, 21/96) the stationary dis-tribution of the Markov Chain?

```
import numpy as np
from numpy.linalg import matrix_power

egien_value, egien_vector = np.linalg.eig(P.T) ## np.linalg,eig(p) returns egien_value, egienvector

print("egien_value :",egien_value)
```

```
## egien_value : [ 1.  -1.  -0.2  0.2]
```

```
print("egien_vector :\n",egien_vector)
```

```
## egien_vector :
##  [[ 0.66212219  0.66212219 -0.5         0.5       ]
##  [ 0.54173634 -0.54173634 -0.5        -0.5       ]
##  [ 0.30096463  0.30096463  0.5        -0.5       ]
##  [ 0.42135049 -0.42135049  0.5         0.5       ]]
```

```
x_1=egien_vector[:,0]

print(x_1)
```

```
## [0.66212219 0.54173634 0.30096463 0.42135049]
```

```
v=x_1/np.sum(x_1)

print(v) # stationary distiribution
```

```
## [0.34375 0.28125 0.15625 0.21875]
```

```
np.dot(v,P) # vP=v
```

```
## array([0.34375, 0.28125, 0.15625, 0.21875])
```

(c)Is $P_{11}^{100} = \pi_1$?, Is $P_{11}^{101} = \pi_1$? Give an expression for π_1 in terms of P_{11}^{100} and P_{11}^{101}.

```
import numpy as np
from numpy.linalg import matrix_power

matrix_power(P,100)
```

```
## array([[0.6875, 0.    , 0.3125, 0.    ],
```

```
##        [0.    , 0.5625, 0.    , 0.4375],
##        [0.6875, 0.    , 0.3125, 0.    ],
##        [0.    , 0.5625, 0.    , 0.4375]])
```

```
matrix_power(P,101)
```

```
## array([[0.    , 0.5625, 0.    , 0.4375],
##        [0.6875, 0.    , 0.3125, 0.    ],
##        [0.    , 0.5625, 0.    , 0.4375],
##        [0.6875, 0.    , 0.3125, 0.    ]])
```

Accoring to reuslt, P_{11}^{100}=0.6875 , P_{11}^{101}=0

it is not true

Exercise 14

Suppose you are working as an independent consultant for a company. Your performance is evaluated at the end of each month and the evaluation falls into one of three categories: 1, 2, and 3. Your evaluation can be modeled as a discrete-time Markov chain and its transition diagram is as follows.

Denote your evaluation at the end of nth month by X_n and assume that X_0 = 2. (a) What are state space, transition probability matrix and initial dis-tribution of X_n?

state space = [1,2,3]

transition probability matrix

$$P = \begin{pmatrix} 0.25 & 0.75 & 0 \\ 0.5 & 0 & 0.5 \\ 0 & 0.25 & 0.75 \end{pmatrix}$$

initial distribution (0,1,0)

(b) What is the stationary distribution?

```
import numpy as np
from numpy.linalg import matrix_power

P=np.array([[0.25, 0.75, 0],
          [0.5, 0, 0.5],
          [0, 0.25, 0.75]])

intial_dis=np.array([0,1,0])

egien_value, egien_vector = np.linalg.eig(P.T) ## np.linalg,eig(p) returns egien_value, egienvector

print("egien_value :",egien_value)
```

```
## egien_value : [-0.55901699  0.55901699  1.        ]
```

```
print("egien_vector :\n",egien_vector)
```

```
## egien_vector :
##  [[ 0.5         0.5         0.28571429]
##  [-0.80901699  0.30901699  0.42857143]
##  [ 0.30901699 -0.80901699  0.85714286]]
```

```
x_1=egien_vector[:,2]
print(x_1)
```

```
## [0.28571429 0.42857143 0.85714286]
```

```
v=x_1/np.sum(x_1)
print("stationary distribution:\n",v) # stationary distribution
```

```
## stationary distribution:
##  [0.18181818 0.27272727 0.54545455]
```

```
np.dot(v,P) # it is right
```

```
## array([0.18181818, 0.27272727, 0.54545455])
```

(c) What is the long-run fraction of time when your evaluation is either 2 or 3?

```
import numpy as np
from numpy.linalg import matrix_power

matrix_power(P,80)
```

```
## array([[0.18181818, 0.27272727, 0.54545455],
##        [0.18181818, 0.27272727, 0.54545455],
##        [0.18181818, 0.27272727, 0.54545455]])
```

```
matrix_power(P,100)
```

```
## array([[0.18181818, 0.27272727, 0.54545455],
##        [0.18181818, 0.27272727, 0.54545455],
##        [0.18181818, 0.27272727, 0.54545455]])
```

we can see that, it converge to some point

Your monthly salary is determined by the evaluation of each month in the following way. Salary when your evaluation is n = $5000 + n^2 \times 5000$; n = 1, 2, 3

(d) What is the long-run average monthly salary?

```
import numpy as np
from numpy.linalg import matrix_power

limiting_prob=matrix_power(P,100)

evaluation_1=5000+(1**2)*5000
evaluation_2=5000+(2**2)*5000
```

```
evaluation_3=5000+(3**2)*5000
evaluation=np.array([evaluation_1,evaluation_2,evaluation_3])

print("long-run average salary:",np.dot(evaluation,limiting_prob[1]))
```

```
## long-run average salary: 35909.09090909091
```

```
"Newsvendor"
```

```
## [1] "Newsvendor"
```