C1_Exercises

Kwon do yun

2021-01-04

Exercise 1

1. State Space
2. Transition Probability Matrix
3. Transition Diagram
4. Initial distribution

- State Space

  a set of all possible states that S. can take Coke or Pepsi, $S = \{c, p\}$

- Transition Probability Matrix

$$P_{2,2} = \begin{bmatrix} 0.7 & 0.3 \\ 0.5 & 0.5 \end{bmatrix} \tag{1}$$

- Transition Diagram

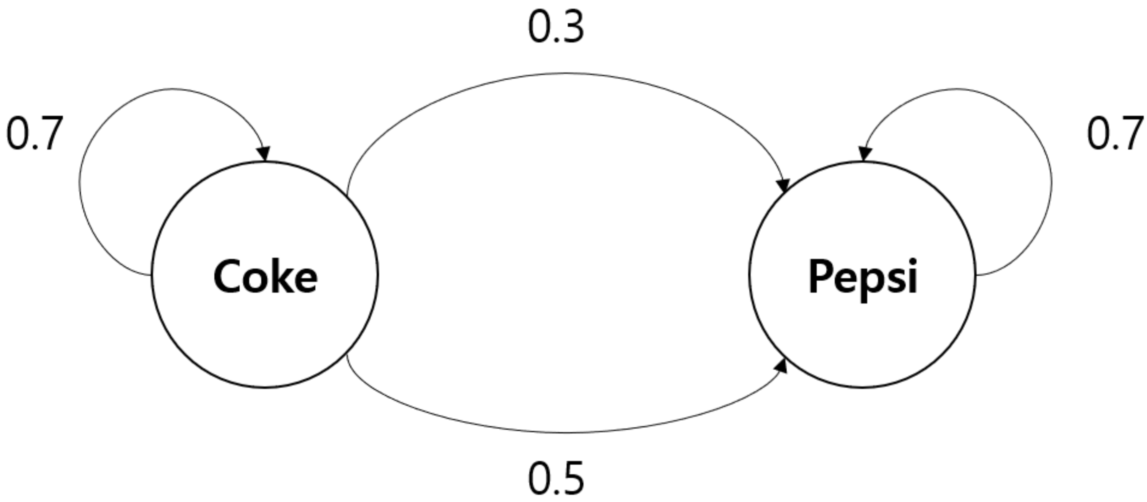

그림 1: Transition Diagram

- Initial distribution

  $\mathbb{P}(S_0 = c) = 0.6, \mathbb{P}(S_0 = p) = 0.4 \Leftrightarrow a_0 = (0.6, 0.4)$

Exercise 2

Suppose, $\mathbb{P}(S_0 = c) = 0.6, \mathbb{P}(S_0 = p) = 0.4$ **then, what is** $\mathbb{P}(S_1 = c) = ?$

$$a_0 = (0.6, 0.4) \tag{2}$$

$$P = \begin{pmatrix} 0.7 & 0.3 \\ 0.5 & 0.5 \end{pmatrix} \tag{3}$$

$$a_0 P = (.6 \quad .4) \begin{pmatrix} .7 & .3 \\ .5 & .5 \end{pmatrix} = 0.62 \tag{4}$$

Excercise 3

Suppose, $\mathbb{P}(S_0 = c) = 0.6, \mathbb{P}(S_0 = p) = 0.4$ **then, what is** $\mathbb{P}(S_2 = c) = ?$

$$a_0 = (0.6, 0.4) \tag{5}$$

$$a_0 P = a_1 \tag{6}$$

$$a_1 P = a_2 \tag{7}$$

$$a_2 = a_0 P P \tag{8}$$

$$a_0 P^2 = (.6 \quad .4) \begin{pmatrix} .64 & .36 \\ .6 & .4 \end{pmatrix} = (.624 \quad .376) \tag{9}$$

Excercise 4

Suppose, $\mathbb{P}(S_0 = c) = 0.6, \mathbb{P}(S_0 = p) = 0.4$ **then, what is** $\mathbb{P}(S_2 = p) = ?$

$$\mathbb{P}(S_0 = p) = a_0 = (0 \quad 1) \tag{10}$$

$$\mathbb{P}(S_2 = p) = a_0 p^2 = (0 \quad 1) \begin{pmatrix} .64 & .36 \\ .6 & .4 \end{pmatrix} = (.6 \quad .4) \tag{11}$$

Thus, $\mathbb{P}(S_2 = p) = 0.4$

DTMC Simulator (p.25)

```
import numpy as np
def soda_simul(this_state):
  u=np.random.rand(1)
  if (this_state == "c"):
    if(u<=0.7):
      next_state = "c"
    else:
      next_state = "p"
  else:
    if(u<=0.5):
      next_state = "c"
    else:
      next_state = "p"
  return next_state
for i in range(5):
  path ="c"
  for i in range (9):
    this_state=path[-1]
    next_state=soda_simul(this_state)
    path=path+next_state
  print(path)
```

```
## cpccppcppp
## cppccpcccc
## cccpcppcpc
## ccppppcppp
## ccpcccccccc
```

p.26

```
def cost_eval(path):
    cost_one_path=path.count("c")*1.5+path.count("p")*1
    return cost_one_path
MC_N=100000
spending_records=np.arange(0,MC_N)
for i in range(MC_N):
    path="c"
    for t in range (9):
        this_state = path[-1]
```

```
        next_state = soda_simul(this_state)
        path=path+next_state
    spending_records[i]=cost_eval(path)
print(np.mean(spending_records))
```

```
## 13.11322
```

```
"Man can learn nothing unless he proceeds from the known to the unknown. - Claude Bernard"
```