

D2 python ver

Lee SungHo

2021-01-15



Page 17 Analytic solution	2
Page 23 Implementation	3
Page 24 The full iteration process	4
Page 27 After 50 steps(coke only)	6
Page 27 After 100 steps(coke only)	7

Page 17 Analytic solution

```
import numpy as np

P = np.array([[0.7,0.3],[0.5,0.5]])
R = np.array([1.5,1])
gamma = 0.9

v = np.linalg.solve((np.eye(2)-gamma*P),R)
v

## array([13.35365854, 12.74390244])
```

Page 23 Implementation

```
import numpy as np

P = np.array([[0.7,0.3],[0.5,0.5]]).reshape(2,2)
R = np.array([1.5,1]).reshape(2,1)
gamma = 0.9
epsilon = 10**(-8)

v_old = np.zeros((2,1))
v_new = R + np.dot(gamma*P,v_old)

while True:
    v_old = v_new
    v_new = R + np.dot(gamma*P,v_old)

    if np.max(abs(v_new-v_old))>epsilon : continue

    break

print(v_old)
```

```
## [[13.35365844]
##  [12.74390234]]
```

Page 24 The full iteration process

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt

P = np.array([[0.7,0.3],[0.5,0.5]]).reshape(2,2)
R = np.array([1.5,1]).reshape(2,1)
gamma = 0.9
epsilon = 10**(-8)

v_old = np.zeros((2,1))
v_new = R + np.dot(gamma*P,v_old)
results = v_old.T
results = np.append(results,v_new.T,axis=0)

while True:
    v_old = v_new
    v_new = R + np.dot(gamma*P,v_old)
    results = np.append(results,v_new.T,axis=0)

    if np.max(abs(v_new-v_old))>epsilon : continue

    break

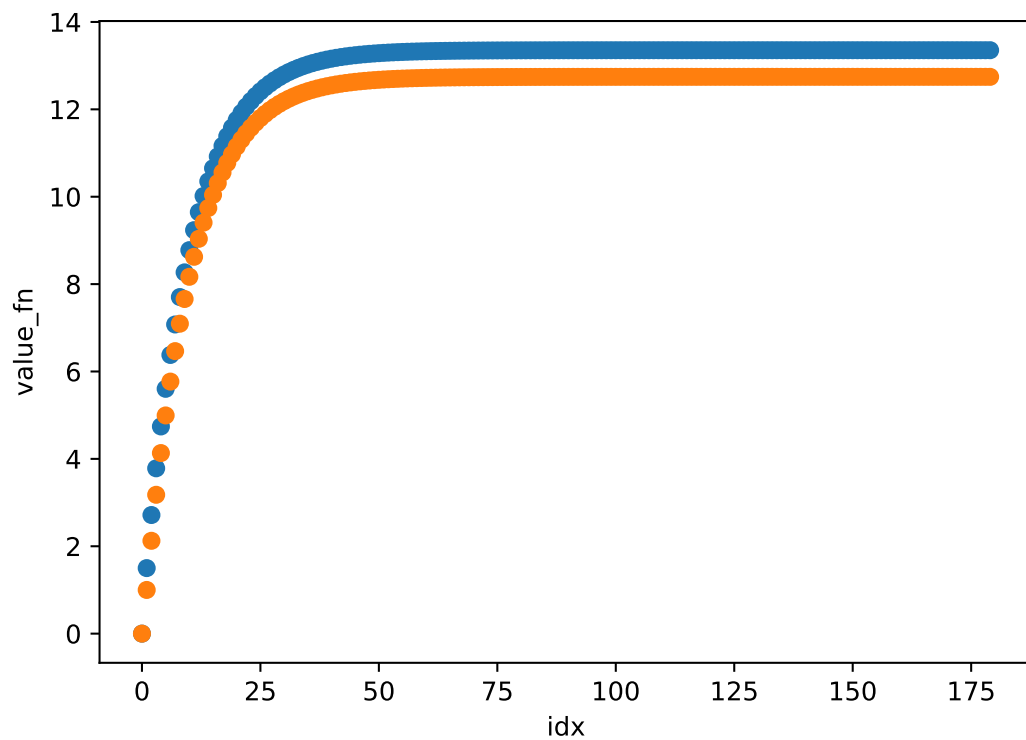
results = pd.DataFrame(results,columns=['coke','pepsi'])
print(results.head(n=7))
```

```
##      coke      pepsi
## 0  0.000000  0.000000
## 1  1.500000  1.000000
## 2  2.715000  2.125000
## 3  3.784200  3.178000
## 4  4.742106  4.132990
## 5  5.603434  4.993793
## 6  6.378488  5.768752
```

```
print(results.tail(n=7))
```

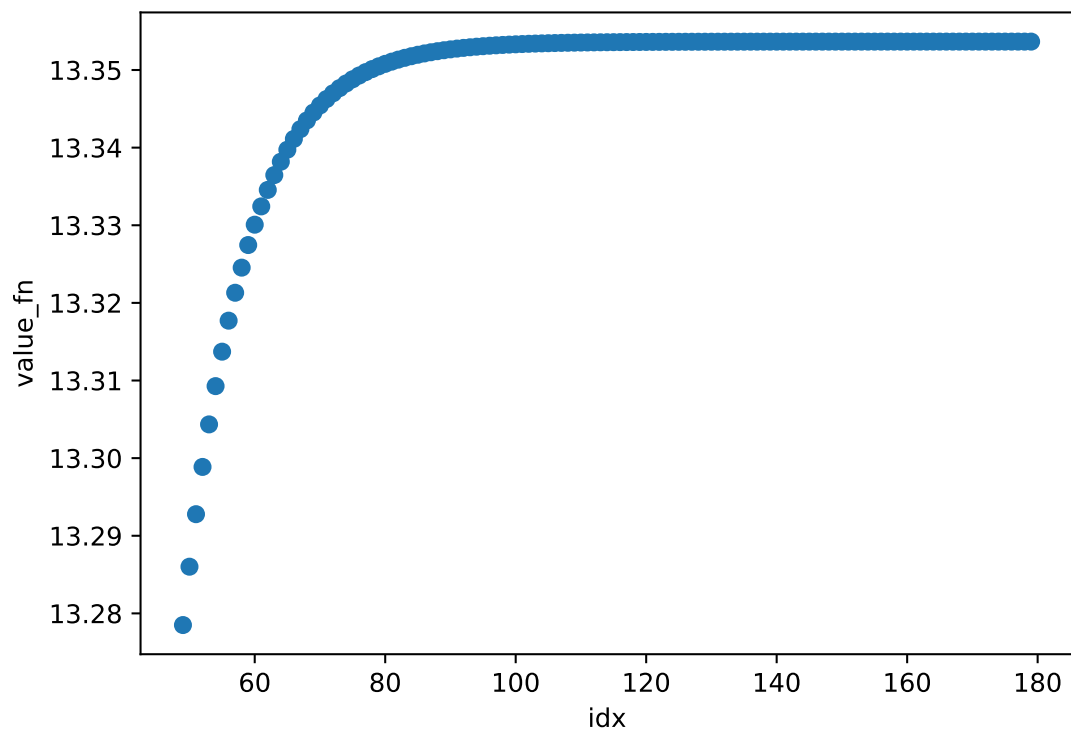
```
##          coke      pepsi
## 173  13.353658  12.743902
## 174  13.353658  12.743902
## 175  13.353658  12.743902
## 176  13.353658  12.743902
## 177  13.353658  12.743902
## 178  13.353658  12.743902
## 179  13.353658  12.743902
```

```
plt.scatter(results.index, results['coke'], label='coke')
plt.scatter(results.index, results['pepsi'], label='pepsi')
plt.xlabel('idx')
plt.ylabel('value_fn')
plt.show()
```



Page 27 After 50 steps(coke only)

```
plt.scatter(results.index[49:],results['coke'][49:], label='coke')  
plt.xlabel('idx')  
plt.ylabel('value_fn')  
plt.show()
```



Page 27 After 100 steps(coke only)

```
plt.scatter(results.index[99:],results['coke'][99:], label='coke')
plt.xlabel('idx')
plt.ylabel('value_fn')
plt.show()
```

