D2_MRP2 Python

Jaemin Park

2021-01-15

차 례

Exercises

p.11 analytic solution

Python code

```
P = np.matrix([[0.7,0.3],[0.5,0.5]])
R = np.matrix([[1.5,1]]).reshape(2,1)
gamma = 0.9
v = np.linalg.inv(np.identity(2)-gamma*P)*R
print(v)
```

```
## [[13.35365854]
##  [12.74390244]]
```

p.21 Iterative algorithm

Python code

```
epsilon = 10**(-8)
v_old = np.matrix([[0,0]]).reshape(2,1)
v_new = R+gamma*P*v_old
while(np.linalg.norm(v_new-v_old)>epsilon):
    v_old=v_new
    v_new = R+gamma*P*v_old
print(v_old)
```

```
## [[13.35365847]
##  [12.74390237]]
```

p.24 full iteration process

Python code

```
epsilon = 10**(-8)
v_old = np.matrix([[0,0]]).reshape(2,1)
v_new = R+gamma*P*v_old
coke_res = []
pep_res = []
while(np.linalg.norm(v_new-v_old)>epsilon):
    coke_res.append(v_old.item(0))
    pep_res.append(v_old.item(1))
    v_old=v_new
    v_new = R+gamma*P*v_old

results = pd.DataFrame({'coke':coke_res, 'pepsi':pep_res})
print(results[:6])
```

```
##        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
```

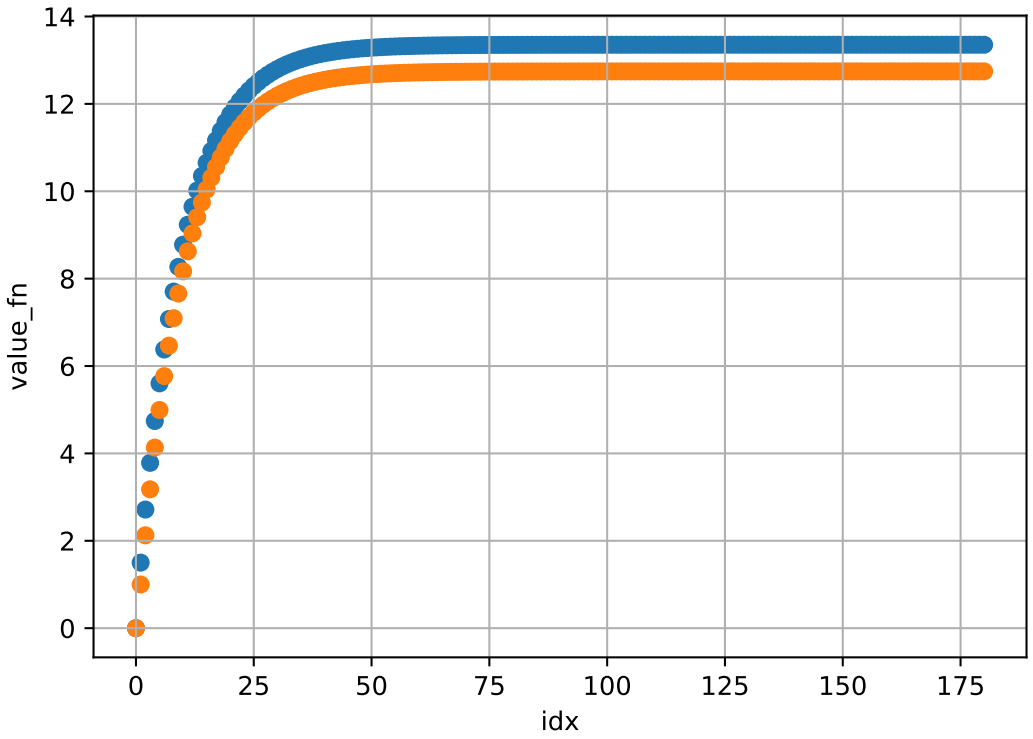
```
print(results[-6:])
```

```
##           coke      pepsi
## 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
## 180  13.353658  12.743902
```

p.25 plot

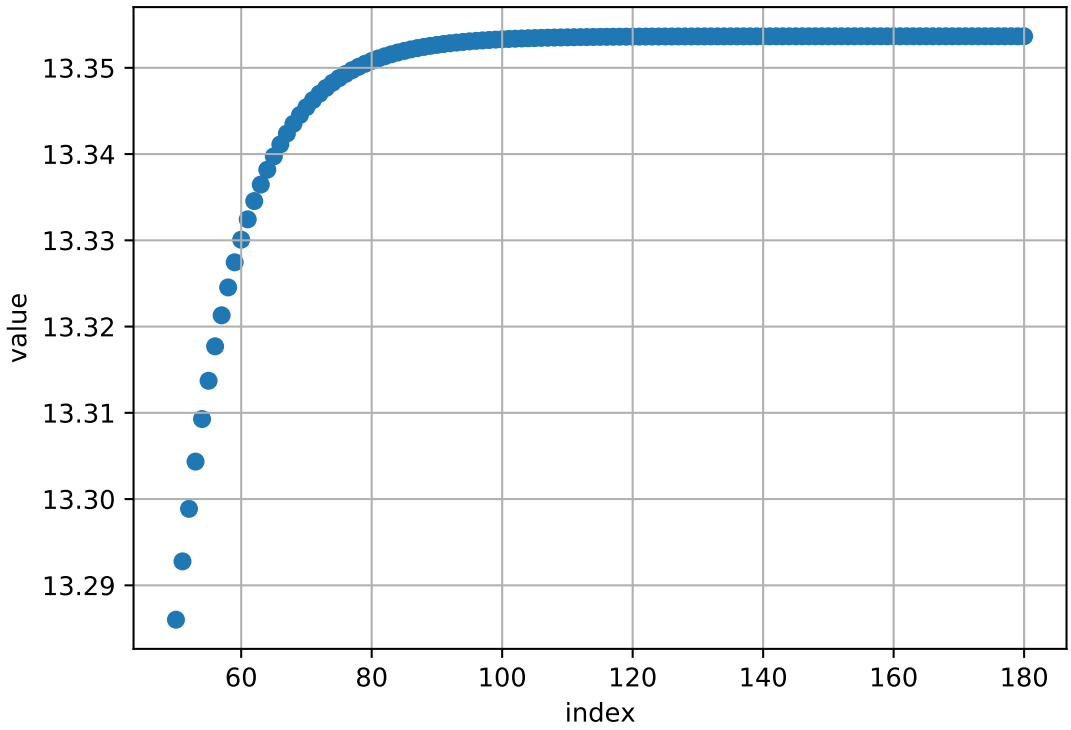
Python code

```
plt.scatter(results.index,results.coke, label='coke')
plt.scatter(results.index,results.pepsi,label='pepsi')
plt.grid(True,axis='both')
plt.xlabel('idx')
plt.ylabel('value_fn')
plt.show()
```



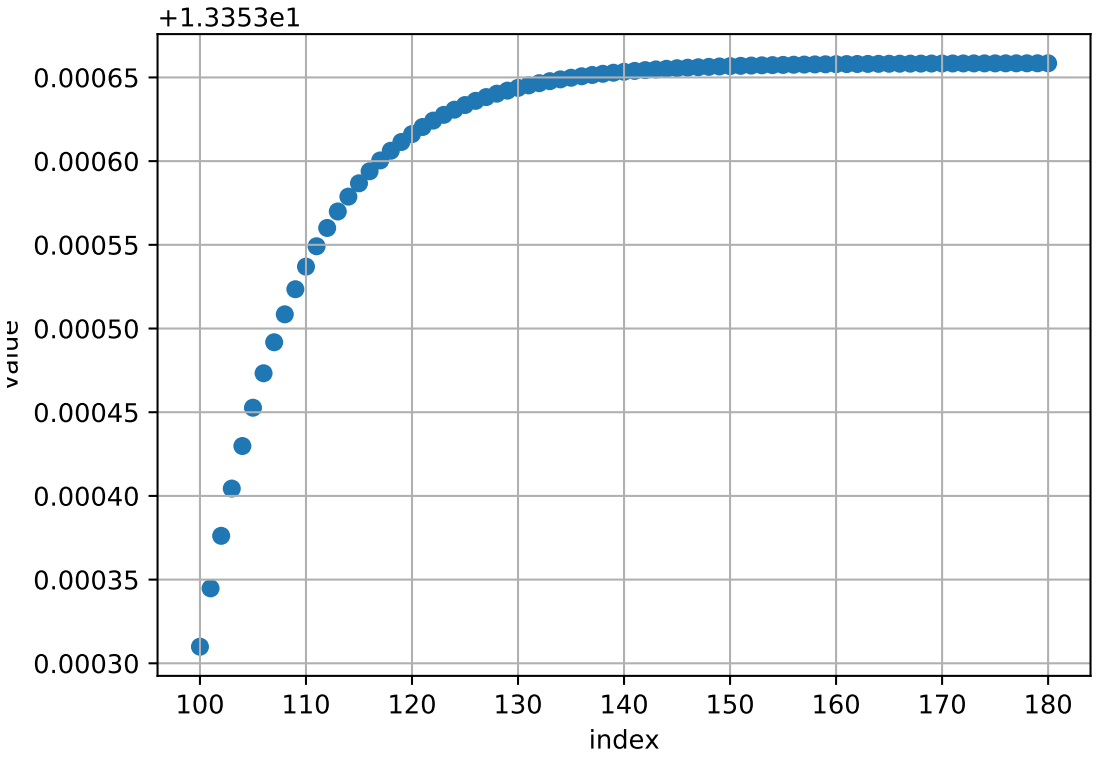
After 50 steps

```
filtered_result = results[50:]
#filtered_result = results[100:]
plt.scatter(filtered_result.index, filtered_result.coke)
plt.grid(True,axis='both')
plt.xlabel("index")
plt.ylabel("value")
plt.show()
```



After 100 steps

```
#filtered_result = results[50:]
filtered_result = results[100:]
plt.scatter(filtered_result.index, filtered_result.coke)
plt.grid(True,axis='both')
plt.xlabel("index")
plt.ylabel("value")
plt.show()
```


+1.3353e1
0.00065
0.00060
0.00055
0.00050
0.00045
0.00040
0.00035
0.00030
value
100
110
120
130
140
150
160
170
180
index