Lab 1120

Yiqiao Huang

Problem 1 (a)

$$1(a).\quad P = \begin{pmatrix} P_{11} & P_{12} & P_{13} \\ P_{21} & P_{22} & P_{23} \\ P_{31} & P_{32} & P_{33} \end{pmatrix} = \begin{pmatrix} 0.2 & 0.7 & 0.1 \\ 0.2 & 0.5 & 0.3 \\ 0.2 & 0.4 & 0.4 \end{pmatrix}$$

Problem 1 (b)

In [1]:
```
import numpy as np
import random
import matplotlib.pyplot as plt
```

In [2]:
```
Pmat = np.array([[.2, .7, .1], [.2, .5, .3], [.2, .4, .4]])
Pmat
```

Out[2]:
```
array([[0.2, 0.7, 0.1],
       [0.2, 0.5, 0.3],
       [0.2, 0.4, 0.4]])
```

In [5]:
```
# try one time if starting from node 2
random.choices([1,2,3], Pmat[1,:])[0]
```

Out[5]:
```
1
```

In [6]:
```
n_simu = 100 # time for simulation
res1b = [1] # realization of the chain

for i in range(n_simu):
    choices = [1, 2, 3] # 3 positions
    prob = Pmat[(res1b[i]-1),:] # probability in current position
    result = random.choices(choices, prob)[0] # choose next position
    res1b.append(result)
print('simulation for 100 times:')
print(res1b)
```

```
simulation for 100 times:
[1, 2, 3, 1, 1, 2, 1, 1, 2, 3, 2, 2, 3, 2, 3, 2, 2, 2, 2, 2, 2, 2, 1, 2, 3, 2, 2, 3,
2, 3, 3, 3, 3, 1, 2, 1, 2, 1, 2, 2, 3, 3, 3, 1, 1, 2, 3, 3, 1, 2, 1, 2, 3, 1, 3, 2,
3, 2, 2, 2, 2, 2, 2, 2, 3, 3, 1, 2, 3, 3, 3, 2, 3, 3, 3, 1, 1, 2, 1, 2, 2, 2, 2, 3,
2, 3, 3, 2, 3, 1, 2, 2, 1, 2, 1, 1, 2, 3, 3, 3, 3]
```

Problem 2 (a)

2(a). Solving $\Pi_\infty^T = \Pi_\infty^T P \iff \Pi_\infty^T (P-I) = 0 \iff (P^T - I)\Pi_\infty = 0$

$\Pi_\infty^T = (x_1, x_2, x_3)$. $\quad (x_1, x_2, x_3) = (x_1, x_2, x_3)\begin{pmatrix} 0.2 & 0.7 & 0.1 \\ 0.2 & 0.5 & 0.3 \\ 0.2 & 0.4 & 0.4 \end{pmatrix}$

$= (0.2(x_1+x_2+x_3),\ 0.7x_1+0.5x_2+0.4x_3,\ 0.1x_1+0.3x_2+0.4x_3)$

So $\begin{aligned} 10x_1 &= 2(x_1+x_2+x_3) \\ 10x_2 &= 7x_1+5x_2+4x_3 \\ 10x_3 &= x_1+3x_2+4x_3 \end{aligned} \Rightarrow \begin{aligned} 4x_1 &= x_2+x_3 \\ 5x_2 &= 7x_1+4x_3 \\ 6x_3 &= x_1+3x_2 \end{aligned}$

Let $x_1 = 1$. then $\left.\begin{aligned} x_2+x_3 &= 4 \\ 5x_2 &= 7+4x_3 \\ 6x_3 &= 1+3x_2 \end{aligned}\right\} \Rightarrow \begin{cases} x_2 = \frac{23}{9} \\ x_3 = \frac{13}{9} \end{cases}$

So $\Pi_\infty^T = \left(\frac{x_1}{\Sigma x_i}, \frac{x_2}{\Sigma x_i}, \frac{x_3}{\Sigma x_i}\right) = \left(\frac{1}{5}, \frac{23}{45}, \frac{13}{45}\right)$.

Problem 2 (b)

In [7]:
```
p_inf = np.array([.2, 23/45, 13/45])
```

Define a function to get the result of $||\pi_i - \pi_\infty||_2^2$

In [9]:
```
def get_norm2(pi, n=30):
    res = []
    for i in range(n):
        pi = pi @ Pmat # get pi_i.T
        norm2_res = np.linalg.norm(pi-p_inf, ord=2) # calculate norm2
        res.append(norm2_res)
    return res
```

Set 4 π_0 and compare the results

In [64]:
```
n = 20
pi1 = np.array([.2,.5,.3]) # pi0
res1 = get_norm2(pi1, n)
```

In [65]:
```
pi2 = np.array([.9,.1,0]) # pi0
res2 = get_norm2(pi2, n)
```

In [66]:
```
pi3 = np.array([1/3,1/3,1/3]) # pi0
res3 = get_norm2(pi3, n)
```

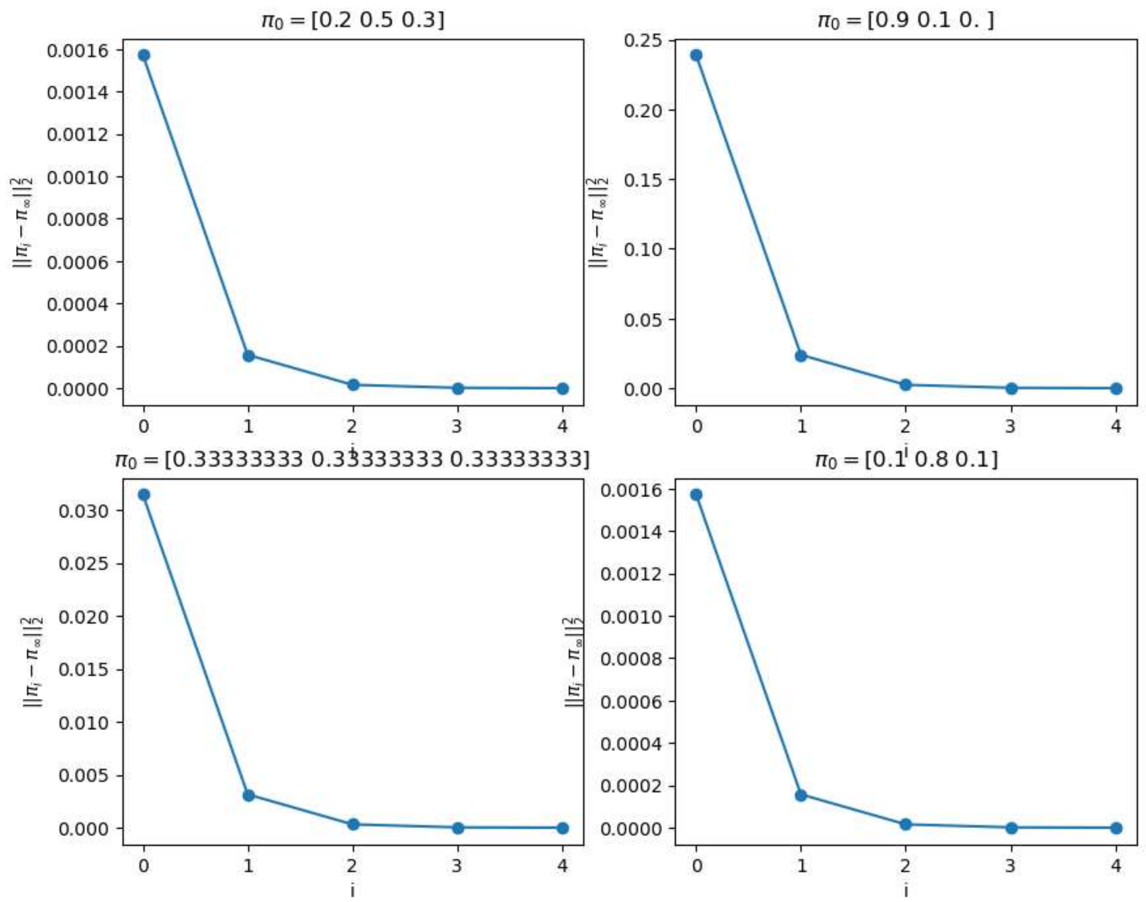
In [67]:

```
pi4 = np.array([.1,.8,.1]) # pi0
res4 = get_norm2(pi4, n)
```

plot the results

In [108]:

```
plt.figure(figsize=(10, 8))
for i in range(4):
    plt.subplot(2, 2, (i+1))
    plt.plot(range(5), locals()[f'res{i+1}'][0:5], marker='o')
    plt.xlabel('i')
    plt.ylabel(r'$||\pi_i-\pi_{\infty}||_2^2$')
    pi_name = f'pi{i+1}'
    plt.title(r'$\pi_0 =$'+f'{locals()[pi_name]}')
```



In [106]:

```
print('norm2 difference goes to 0 when i =', np.min(np.where(np.array(res1)==0))+1, 'wi
print('norm2 difference goes to 0 when i =', np.min(np.where(np.array(res2)==0))+1, 'wi
print('norm2 difference goes to 0 when i =', np.min(np.where(np.array(res3)==0))+1, 'wi
print('norm2 difference goes to 0 when i =', np.min(np.where(np.array(res4)==0))+1, 'wi
```

```
norm2 difference goes to 0 when i = 16 with pi0 = [0.2 0.5 0.3]
norm2 difference goes to 0 when i = 19 with pi0 = [0.9 0.1 0. ]
norm2 difference goes to 0 when i = 16 with pi0 = [0.33333333 0.33333333 0.33333333]
norm2 difference goes to 0 when i = 16 with pi0 = [0.1 0.8 0.1]
```

From the above plot and output results, we can observe that for π_1, it is close to π_∞. Therefore, the computed $||\pi_i - \pi_\infty||_2^2$ is relatively small, and the convergence is very rapid. However, for π_2, it is far from π_∞, resulting in slower convergence and a larger $||\pi_i - \pi_\infty||_2^2$.

problem 3(a)

Define a funciton to get arrival time.

In [133]:
```
def arrival_time(start_node, Pmat):
    res = [start_node] # realization of the chain
    node = start_node
    i = 0
    while node != 3:
        choices = [1, 2, 3] # 3 positions
        prob = Pmat[(node-1),:] # probability in current position
        node = random.choices(choices, prob)[0] # choose next position
        i = i+1
        #print(i,': ',node)
    return i
```

In [132]:
```
arrival_time(1, Pmat) # try one time
```

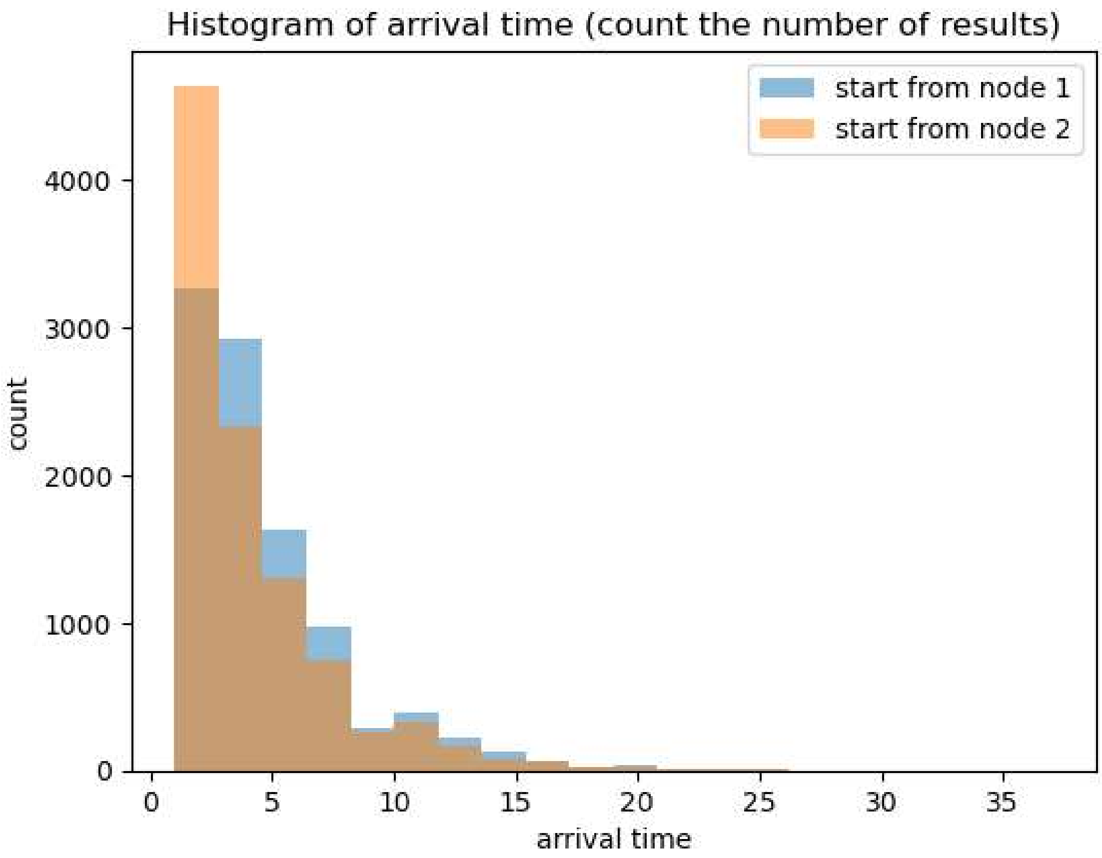
```
1 :  1
2 :  1
3 :  2
4 :  2
5 :  2
6 :  2
7 :  2
8 :  2
9 :  3
```

Out[132]: 9

In [150]:
```
n = 10000
res1 = [arrival_time(1, Pmat) for n0 in range(n)]
res2 = [arrival_time(2, Pmat) for n0 in range(n)]
```

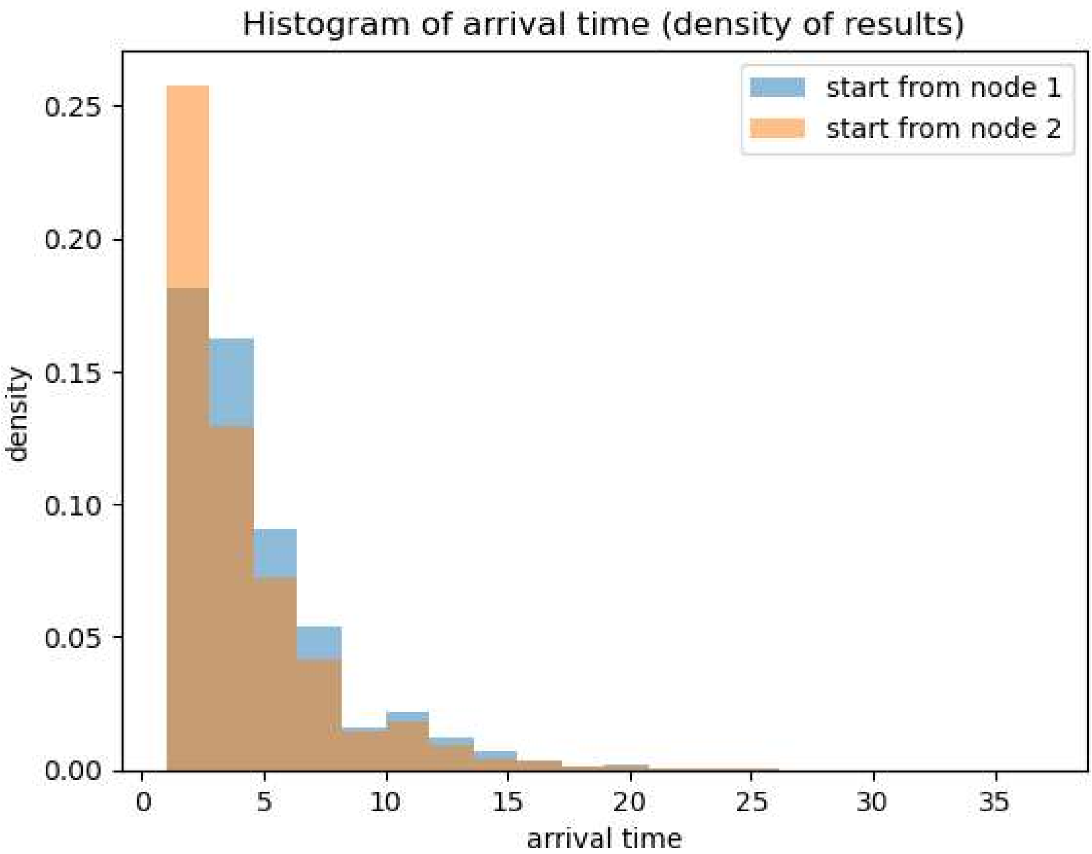
In [151]:
```
plt.hist(res1, density=0, bins=20, alpha=.5, label='start from node 1')
plt.hist(res2, density=0, bins=20, alpha=.5, label='start from node 2')
plt.xlabel('arrival time')
plt.ylabel('count')
plt.legend()
plt.title('Histogram of arrival time (count the number of results)')
```

Out[151]:
```
Text(0.5, 1.0, 'Histogram of arrival time (count the number of results)')
```



In [152]:
```
plt.hist(res1, density=1, bins=20, alpha=.5, label='start from node 1')
plt.hist(res2, density=1, bins=20, alpha=.5, label='start from node 2')
plt.xlabel('arrival time')
plt.ylabel('density')
plt.legend()
plt.title('Histogram of arrival time (density of results)')
```

Out[152]:
```
Text(0.5, 1.0, 'Histogram of arrival time (density of results)')
```



In [153]:
```
print('mean time for arriving node 3 if starting from node 1:',np.mean(res1))
print('mean time for arriving node 3 if starting from node 2:',np.mean(res2))
```

```
mean time for arriving node 3 if starting from node 1: 4.6241
mean time for arriving node 3 if starting from node 2: 3.8811
```

In [154]:
```
print('theoritical mu1:', 60/13)
print('theoritical mu2:', 50/13)
```

```
theoritical mu1: 4.615384615384615
theoritical mu2: 3.8461538461538463
```

The simulation results and theoritical results are close.

problem 3(b)

3(b). $\mu_i = 1 + \sum_{j=1}^{3} P_{ij}\mu_j$

then $\mu_1 = 1 + \sum_{j=1}^{3} P_{1j}\mu_j = 1 + 0.2\mu_1 + 0.7\mu_2 + 0.1\mu_3$

$\mu_2 = 1 + 0.2\mu_1 + 0.5\mu_2 + 0.3\mu_3$

$\mu_3 = 0$

then $\begin{cases} 8\mu_1 = 10 + 7\mu_2 + \mu_3 \\ 5\mu_2 = 10 + 2\mu_1 + 3\mu_3 \\ \mu_3 = 0 \end{cases}$

Solve the above equations, we can get

$\mu_1 = \frac{60}{13}$ $\mu_2 = \frac{50}{13}$.

In [ ]: