

## hw5

April 3, 2023

```
[19]: import numpy as np
transition_matrix = np.array([[22/45,1/6,1/9,1/10,0,0,2/15,0,0],
                             [2/9,8/21,1/9,0,1/7,0,0,1/7,0],
                             [2/9,1/6,13/36,0,0,1/8,0,0,1/8],
                             [4/15,0,0,4/15,2/9,1/9,2/15,0,0],
                             [0,3/14,0,1/6,23/63,1/9,0,1/7,0],
                             [0,0,1/4,1/6,2/9,17/72,0,0,1/8],
                             [4/15,0,0,1/10,0,0,1/3,1/5,1/10],
                             [0,3/14,0,0,1/7,0,1/5,12/35,1/10],
                             [0,0,1/4,0,0,1/8,1/5,1/5,9/40]])

transition_matrix_transp = transition_matrix.T
eigenvals, eigenvects = np.linalg.eig(transition_matrix_transp)

# Find the indexes of the eigenvalues that are close to one. Use them to select
# the target eigen vectors. Flatten the result.
close_to_1_idx = np.isclose(eigenvals,1)
target_eigenvect = eigenvects[:,close_to_1_idx]
target_eigenvect = target_eigenvect[:,0]
# Turn the eigenvector elements into probabilities
py_stationary_distrib = target_eigenvect / sum(target_eigenvect)
target_stationary_distrib = [8/37,6/37,4/37,3/37,4/37,2/37,4/37,4/37,2/37]

print("Target stationary distribution")
print(target_stationary_distrib)

print("Python code output for stationary distribution")
print(list(py_stationary_distrib))

print("Difference between target and computed stationary distribution")
print(target_stationary_distrib-py_stationary_distrib)
```

Target stationary distribution

[0.21621621621621623, 0.16216216216216217, 0.10810810810810811,  
0.08108108108108109, 0.10810810810810811, 0.05405405405405406,  
0.10810810810810811, 0.10810810810810811, 0.05405405405405406]

Python code output for stationary distribution

[(0.21621621621621648-0j), (0.16216216216216206-0j), (0.10810810810810813-0j),

```
(0.08108108108108104-0j), (0.10810810810810804-0j), (0.05405405405405406-0j),  
(0.10810810810810813-0j), (0.10810810810810803-0j), (0.054054054054054015-0j)]  
Difference between target and computed stationary distribution  
[-2.49800181e-16+0.j  1.11022302e-16+0.j -1.38777878e-17+0.j  
  4.16333634e-17+0.j  6.93889390e-17+0.j  0.00000000e+00+0.j  
 -1.38777878e-17+0.j  8.32667268e-17+0.j  4.16333634e-17+0.j]
```

[ ]: