MAT1120 Oblig 2

Oskar Idland

Oppgave 1

$$A = \begin{bmatrix} 0 & 0 & \frac{1}{2} & 1 \\ \frac{1}{3} & 0 & 0 & 0 \\ \frac{1}{3} & \frac{1}{2} & 0 & 0 \\ \frac{1}{3} & \frac{1}{2} & \frac{1}{2} & 0 \end{bmatrix}$$

Vi finner nullrommet ved å løse likningen

$$(A - I_4)\mathbf{x} = \mathbf{0}$$

Da får vi at nullrommet har en basis

$$\mathbf{x} = \begin{pmatrix} \frac{4}{3} \\ \frac{4}{9} \\ \frac{2}{3} \\ 1 \end{pmatrix}$$

For å finne score vektoren må vi sørge for at summen s av elementene er lik 1. Vi summerer da alle verdiene og deler $\mathbf{x}$ på denne summen

$$s = \frac{4}{3} + \frac{4}{9} + \frac{2}{3} + 1 = \frac{31}{9}$$

Score vektoren blir da

$$\frac{9}{31}\mathbf{x} = \begin{pmatrix} \frac{12}{31} \\ \frac{4}{31} \\ \frac{6}{31} \\ \frac{9}{31} \end{pmatrix} = \frac{1}{31}\begin{pmatrix} 12 \\ 4 \\ 6 \\ 9 \end{pmatrix}$$

Nå har hvert dokument en score og de kan rangeres

Dokument	Rank
1	1
4	2
3	3
2	4

Oppgave 2

$$A = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & \frac{1}{2} & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & \frac{1}{2} & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Vi finner nullrommet ved å løse likningen

$$(A - I_5)\mathbf{x} = \mathbf{0}$$

$$\begin{bmatrix} 1 & 0 & 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Dette likningsettet har ingen løsning og derfor har ikke matrisen en unik score vektor.

Oppgave 3

Begge matriser er stokastiske ettersom alle søylene inneholder ikke-negative elementer. For å se om den første matrisen er regulær må vi finne en eksponent som gjør at alle elementene er positive tall

$$A = \begin{bmatrix} 0 & 0 & \frac{1}{2} & 1 \\ \frac{1}{3} & 0 & 0 & 0 \\ \frac{1}{3} & \frac{1}{2} & 0 & 0 \\ \frac{1}{3} & \frac{1}{2} & \frac{1}{2} & 0 \end{bmatrix}$$

$$A^4 = \begin{bmatrix} \frac{1}{3} & \frac{3}{8} & \frac{11}{24} & \frac{5}{12} \\ \frac{5}{36} & \frac{1}{12} & \frac{1}{12} & \frac{1}{6} \\ \frac{2}{9} & \frac{5}{24} & \frac{1}{6} & \frac{1}{6} \\ \frac{11}{36} & \frac{1}{3} & \frac{7}{24} & \frac{1}{4} \end{bmatrix}$$

Vi ser at hvis den første matrisen får 4 som eksponent blir alle elementene positive og den er dermed regulær. Vi prøver så dette på den andre matrisen

$$B = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & \frac{1}{2} & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & \frac{1}{2} & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$B^2 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & \frac{1}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{2} & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} = B^4 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & \frac{1}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{2} & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

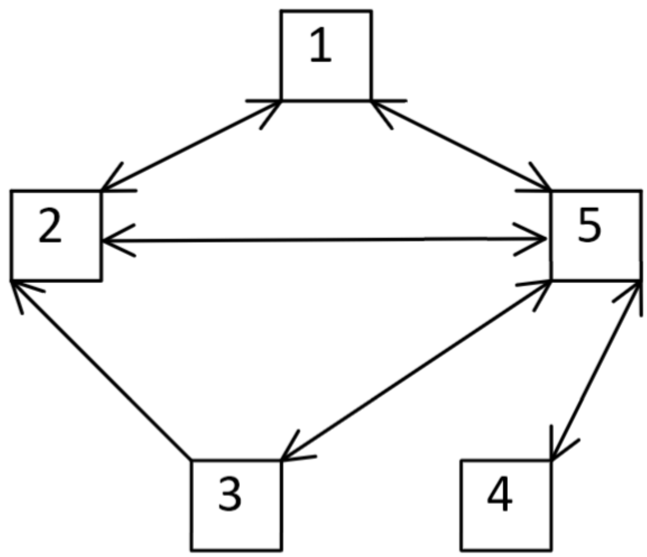
Her ser vi et klart mønster hvor matrisen opphøyd i n er en speilet versjon av matrisen opphøyd i $n + 1$ om midterste søyle og vil dermed aldri opppnå kravet om at alle elementer er positive. Denne matrisen er dermed ikke regulær

Oppgave 7

Vi kjører koden og får følgende matrise:

$$A = \begin{bmatrix} 0 & \frac{1}{2} & 0 & 0 & \frac{1}{4} \\ \frac{1}{2} & 0 & \frac{1}{2} & 0 & \frac{1}{4} \\ 0 & 0 & 0 & 0 & \frac{1}{4} \\ 0 & 0 & 0 & 0 & \frac{1}{4} \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & 1 & 0 \end{bmatrix}$$

Dette tilsvarer følgende web



Figur 1: Web som tilsvarer matrisen gitt over

Oppgave 8-10

Ved bruk av en python funskjon ser vi at score vektoren til A blir følgende

$$\approx \begin{pmatrix} 0.21 \\ 0.24 \\ 0.10 \\ 0.10 \\ 0.36 \end{pmatrix}$$

Ved bruk av python funksjonen rankingApprox rett etter den nøyaktige løsningen ser vi at approksimasjonen er helt presis for vårt anntall desimaler.

```
[Running] python -u
"c:\Users\oskar\OneDrive\Dokumenter\GitHub\P
rivat\MAT1120\Oblig 2\MAT1120 Oblig 2.py"
Score vektor ved bruk av ranking
[[0.20701799]
 [0.23802371]
 [0.09990732]
 [0.09990732]
 [0.35514365]]

Score vektor ved bruk av rankingApprox
[[0.20701799]
 [0.23802371]
 [0.09990732]
 [0.09990732]
 [0.35514365]]

[Done] exited with code=0 in 0.703 seconds
```

Figur 2: Output fra kode ved å kjøre funksjonene ranking og rankingApprox

Kode

```
import numpy as np
from numpy import random
from scipy.linalg import null_space

def randlinkmatrix(n):
    A = np.round(np.random.rand(n,n))
    for k in range(n-1):
        A[k,k] = 0
        if np.all(A[:,k]) == 0:
            A[n-1,k] = 1
        s = np.sum(A[:,k])
        A[:,k] = 1/s * A[:,k]

    A[n-1,n-1] = 0
    if np.all(A[:,n-1]) == 0:
        A[1,n-1] = 1
    s = np.sum(A[:,n-1])
    A[:,n-1] = 1/s * A[:,n-1]

    return A

def ranking(A):
    n = A.shape[0]
    if A.shape[0] != A.shape[1]:
        raise ValueError("This matrix is not a n x n matrix")

    if np.count_nonzero(np.diag(A)) != 0:
        raise ValueError("This matrix has non zero values along its diagonal")

    for k in range(n):
        if np.sum(A[:,k]) != 1:
            raise ValueError("The sum of this matrix’s colums does not equal 1")

    if any(A) < 0:
        raise ValueError("There are negative values in this matrix")

    m = .1
    S = np.full((n,n), 1/n)
    M =  (1-m)*A + m*S
    I = np.eye(n)
    nullspace = null_space(M-I)
    score = nullspace/np.sum(nullspace)
    return score

def rankingApprox(A, delta):

    n = A.shape[0]
    if A.shape[0] != A.shape[1]:
        raise ValueError("This matrix is not a n x n matrix")

    if np.count_nonzero(np.diag(A)) != 0:
        raise ValueError("This matrix has non zero values along its diagonal")

    for k in range(n):
        if np.sum(A[:,k]) != 1:
```

```
            raise ValueError("The sum of this matrix’s columns does not equal 1")

    if np.any(A) < 0:
        raise ValueError('There are negative values in the matrix')

    n = A.shape[0]
    m = .1
    S = np.full((n,n), 1/n)
    M =  (1-m)*A + m*S
    x0 = np.full((n,1), 1/n)
    x = [x0]
    x.append(np.matmul(M,x0))
    k = 1
    while np.any(x[k] - x[k-1]) >= delta:
        x.append(np.matmul(M,x[k]))
        k += 1
    return x[-1]

A = np.array([[  0, 1/2,   0, 0, 1/4],
              [1/2,   0, 1/2, 0, 1/4],
              [  0,   0,   0, 0, 1/4],
              [  0,   0,   0, 0, 1/4],
              [1/2, 1/2, 1/2, 1,   0]])

print(f'Score vektor ved bruk av ranking \n{ranking(A)}')
print()
print(f'Score vektor ved bruk av rankingApprox \n{rankingApprox(A, .01)}')
```