Національний технічний університет України «КПІ ім. Ігоря Сікорського»

Факультет інформатики та обчислювальної техніки

Кафедра інформаційних систем та технологій

Лабораторна робота № 4

з дисципліни «Спеціальні розділи математики-2. Чисельні методи»

Виконав:

студент гр. ІС-34

Колосов Ігор

Викладач:

доц. Рибачук Л.В.

Київ – 2024

Зміст

1. Постановка задачі

2 Завдання

Створити програму, для зведення матриці A до нормальної форми Фробеніуса P.

Отримане характеристичне рівняння розв'язати довільним способом у Mathcad і отримати всі власні числа λ_i, $i = 1,\ldots,m$ з точністю 5 знаків після коми.

Для кожного власного числа знайти по одному власному вектору через власні вектори матриці P.

Перевірити точність знайдених результатів, підставляючи у рівняння (1) знайдені власні числа та власні вектори.

Знайти власні числа матриці A виключно за допомогою Mathcad і порівняти з отриманими раніше результатами.

2. Проміжні матриці M^-1 та M_i, результуюча матриця P у нормальній формі фронебіуса

```
Starting matrix:
    7.03000,    1.14000,    0.93000,    1.13500
    1.14000,    3.39000,    1.30000,    0.16000
    0.93000,    1.30000,    6.21000,    2.10000
    1.13500,    0.16000,    2.10000,    5.33000
```

```
Matrix M3
     1.00000,     0.00000,    -0.54048,     0.00000
     0.00000,     1.00000,    -0.07619,     0.00000
     0.00000,     0.00000,     0.47619,     0.00000
     0.00000,     0.00000,    -2.53810,     1.00000

Matrix A1
     6.52736,     0.43738,     2.38318,     0.00000
     1.06914,     3.29095,     3.47643,     0.00000
     0.44286,     0.61905,    12.14169,     1.00000
    -1.22543,    -3.13952,   -30.58249,     0.00000

Matrix M2
     1.00000,    -0.68553,     0.00000,     0.00000
     0.00000,     0.28765,     0.00000,     0.00000
     0.00000,    -3.49257,     1.00000,     0.00000
     0.00000,     8.79710,     0.00000,     1.00000

Matrix A2
     5.79443,     7.48677,    -0.00000,     0.00000
     0.30754,    16.16557,     1.00000,     0.00000
    -3.29120,   -76.23167,     0.00000,     1.00000
     8.17992,   109.22541,     0.00000,     0.00000

Matrix M1
     0.13357,     0.00000,     0.00000,     0.00000
    -2.15922,     1.00000,     0.00000,     0.00000
    10.18218,     0.00000,     1.00000,     0.00000
   -14.58912,     0.00000,     0.00000,     1.00000

Matrix A3
    21.96000,     1.00000,    -0.00000,     0.00000
  -167.59947,     0.00000,     1.00000,    -0.00000
   526.30419,     0.00000,    -0.00000,     1.00000
  -571.65804,     0.00000,     0.00000,    -0.00000
```

$$M3 := \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ \frac{(-A)_{3,0}}{A_{3,2}} & \frac{-A_{3,1}}{A_{3,2}} & \frac{1}{A_{3,2}} & \frac{-A_{3,3}}{A_{3,2}} \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ -0.54 & -0.076 & 0.476 & -2.538 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$A1 := M3^{-1} \cdot A \cdot M3 = \begin{pmatrix} 6.527 & 1.069 & 0.443 & -1.225 \\ 0.437 & 3.291 & 0.619 & -3.14 \\ 2.383 & 3.476 & 12.142 & -30.582 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

$$M2 := \begin{bmatrix} 1 & 0 & 0 & 0 \\ \frac{(-A1)_{2,0}}{A1_{2,1}} & \frac{1}{A1_{2,1}} & \frac{(-A1)_{2,2}}{A1_{2,1}} & \frac{-A1_{2,3}}{A1_{2,1}} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ -0.686 & 0.288 & -3.493 & 8.797 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$A2 := M2^{-1} \cdot A1 \cdot M2 = \begin{pmatrix} 5.794 & 0.308 & -3.291 & 8.18 \\ 7.487 & 16.166 & -76.232 & 109.225 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

$$M1 := \begin{pmatrix} \frac{1}{A2_{1,0}} & \frac{-A2_{1,1}}{A2_{1,0}} & \frac{-A2_{1,2}}{A2_{1,0}} & \frac{-A2_{1,3}}{A2_{1,0}} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 0.134 & -2.159 & 10.182 & -14.589 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$A3 := M1^{-1} \cdot A2 \cdot M1 = \begin{pmatrix} 21.96 & -167.599 & 526.304 & -571.658 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

$P := A3$

Матриця P має нормальну форму фронебіуса

3. Отримане характеристичне рівняння та власні числа

```
Eigenvalues
    9.38916,    6.00237,    4.08593,    2.48254
```

Знаходимо всі власні начення як корені характеристичного рівнняння за допомогою Mathcad

$$\lambda^4 - P_{0,0} \cdot \lambda^3 - P_{0,1} \cdot \lambda^2 - P_{0,2} \cdot \lambda - P_{0,3} \rightarrow -526.30418600000007 \cdot \lambda + 167.59947499999998 \cdot \lambda^2 + -21.96 \cdot \lambda^3 + \lambda^4 + 571.65804481249938$$

$$V := \begin{pmatrix} -P_{0,3} \\ -P_{0,2} \\ -P_{0,1} \\ -P_{0,0} \\ 1 \end{pmatrix} = \begin{pmatrix} 571.658 \\ -526.304 \\ 167.599 \\ -21.96 \\ 1 \end{pmatrix} \quad \lambda := \mathrm{polyroots}(V) = \begin{pmatrix} 2.483 \\ 4.086 \\ 6.002 \\ 9.389 \end{pmatrix}$$

Знаходимо власні значення як корені характеристичного рівняння з програми

$$\lambda 0 := \begin{pmatrix} 2.48254 \\ 4.08593 \\ 6.00237 \\ 9.38916 \end{pmatrix}$$

Eigenvalues
9.38916, 6.00237, 4.08593, 2.48254

$$e := |\lambda - \lambda 0| = 5.78 \times 10^{-6}$$

4. Власні числа - корені характеристичного рівняння

Знаходимо власні вектори матриці A

$$Y1 := \begin{bmatrix} (\lambda_0)^3 \\ (\lambda_0)^2 \\ (\lambda_0)^1 \\ (\lambda_0)^0 \end{bmatrix} = \begin{pmatrix} 15.3 \\ 6.163 \\ 2.483 \\ 1 \end{pmatrix} \quad Y2 := \begin{bmatrix} (\lambda_1)^3 \\ (\lambda_1)^2 \\ (\lambda_1)^1 \\ (\lambda_1)^0 \end{bmatrix} = \begin{pmatrix} 68.214 \\ 16.695 \\ 4.086 \\ 1 \end{pmatrix} \quad Y3 := \begin{bmatrix} (\lambda_2)^3 \\ (\lambda_2)^2 \\ (\lambda_2)^1 \\ (\lambda_2)^0 \end{bmatrix} = \begin{pmatrix} 216.256 \\ 36.028 \\ 6.002 \\ 1 \end{pmatrix} \quad Y4 := \begin{bmatrix} (\lambda_3)^3 \\ (\lambda_3)^2 \\ (\lambda_3)^1 \\ (\lambda_3)^0 \end{bmatrix} = \begin{pmatrix} 827.715 \\ 88.156 \\ 9.389 \\ 1 \end{pmatrix}$$

```
Y:
 827.71514,  216.25582,   68.21385,   15.29987
  88.15642,   36.02842,   16.69482,    6.16299
   9.38916,    6.00237,    4.08593,    2.48254
   1.00000,    1.00000,    1.00000,    1.00000
```

5. Матриця подібності S

$$S := M3 \cdot M2 \cdot M1 = \begin{pmatrix} 0.134 & -2.159 & 10.182 & -14.589 \\ -0.092 & 1.768 & -10.473 & 18.798 \\ -0.065 & 1.032 & -4.229 & 3.915 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

```
Matrix S:
   0.13357,   -0.09156,   -0.06521,    0.00000
  -2.15922,    1.76785,    1.03231,    0.00000
  10.18218,  -10.47272,   -4.22912,    0.00000
 -14.58912,   18.79831,    3.91473,    1.00000
```

6. Оцінка точності обчислень

$$x1 := S \cdot Y1 = \begin{pmatrix} -0.575 \\ 2.294 \\ -1.22 \\ 1 \end{pmatrix} \quad A \cdot x1 = \begin{pmatrix} -1.428 \\ 5.694 \\ -3.028 \\ 2.483 \end{pmatrix} \quad \lambda_0 \cdot x1 = \begin{pmatrix} -1.428 \\ 5.694 \\ -3.028 \\ 2.483 \end{pmatrix}$$

$$x2 := S \cdot Y2 = \begin{pmatrix} 0.078 \\ -0.725 \\ -0.579 \\ 1 \end{pmatrix} \quad A \cdot x2 = \begin{pmatrix} 0.319 \\ -2.96 \\ -2.367 \\ 4.086 \end{pmatrix} \quad \lambda_1 \cdot x2 = \begin{pmatrix} 0.319 \\ -2.96 \\ -2.367 \\ 4.086 \end{pmatrix}$$

$$x3 := S \cdot Y3 = \begin{pmatrix} -2.38 \\ -0.171 \\ 1.62 \\ 1 \end{pmatrix} \quad A \cdot x3 = \begin{pmatrix} -14.286 \\ -1.029 \\ 9.721 \\ 6.002 \end{pmatrix} \quad \lambda_2 \cdot x3 = \begin{pmatrix} -14.286 \\ -1.029 \\ 9.721 \\ 6.002 \end{pmatrix}$$

$$x4 := S \cdot Y4 = \begin{pmatrix} 1.221 \\ 0.526 \\ 1.233 \\ 1 \end{pmatrix} \quad A \cdot x4 = \begin{pmatrix} 11.466 \\ 4.938 \\ 11.575 \\ 9.389 \end{pmatrix} \quad \lambda_3 \cdot x4 = \begin{pmatrix} 11.466 \\ 4.938 \\ 11.575 \\ 9.389 \end{pmatrix}$$

7. Порівняння власного рішення з рішенням Mathcad

```
Eigenvectors:
    1.22121,    0.52588,    1.23283,    1.00000
   -2.38005,   -0.17141,    1.61960,    1.00000
    0.07806,   -0.72454,   -0.57940,    1.00000
   -0.57513,    2.29370,   -1.21985,    1.00000
```

Власні вектори отримані програмним шляхом:
1.22121, 0.52588, 1.23283, 1.00000
-2.38005, -0.17141, 1.61960, 1.00000
0.07806, -0.72454, -0.57940, 1.00000
-0.57513, 2.29370, -1.21985, 1.00000

$$x\_prog1 := \begin{pmatrix} 1.22121 \\ 0.52588 \\ 1.23283 \\ 1.00000 \end{pmatrix} \quad x\_prog2 := \begin{pmatrix} -2.38005 \\ -0.17141 \\ 1.61960 \\ 1.00000 \end{pmatrix} \quad x\_prog3 := \begin{pmatrix} 0.07806 \\ -0.72454 \\ -0.57940 \\ 1.00000 \end{pmatrix} \quad x\_prog4 := \begin{pmatrix} -0.57513 \\ 2.29370 \\ -1.21985 \\ 1.00000 \end{pmatrix}$$

$$x\_prog1 - x4 = \begin{pmatrix} -2.402 \times 10^{-6} \\ -3.839 \times 10^{-6} \\ -2.097 \times 10^{-6} \\ 0 \end{pmatrix} \qquad x\_prog2 - x3 = \begin{pmatrix} 1.945 \times 10^{-6} \\ -7.442 \times 10^{-7} \\ 3.695 \times 10^{-6} \\ 0 \end{pmatrix}$$

$$x\_prog4 - x1 = \begin{pmatrix} -4.544 \times 10^{-6} \\ -4.133 \times 10^{-6} \\ 1.104 \times 10^{-6} \\ 0 \end{pmatrix} \qquad x\_prog3 - x2 = \begin{pmatrix} -3.432 \times 10^{-6} \\ 3.644 \times 10^{-6} \\ 2.694 \times 10^{-6} \\ 0 \end{pmatrix}$$

8. Лістинг програми

```
import numpy as np
from lib_print import *
np.set_printoptions(suppress=True)


__default_print_format = ANSI.FG.BRIGHT_BLACK + ANSI.Styles.ITALIC
__default_numbers_format = "10.5f"
__default_text_format = ANSI.Styles.BOLD

def default_matrix():
    t = 7.0
    k = 3 * (3 - 4) + 1
    a = 0.11 * t
    b = 0.02 * k
    g = 0.02 * k
    d = 0.015 * t
    return np.array([
        [6.26 + a, 1.10 - b, 0.97 + g, 1.24 - d],
```

```python
        [1.10 - b, 4.16 - a, 1.30, 0.16],
        [0.97 + g, 1.30, 5.44 + a, 2.1],
        [1.24 - d, 0.16, 2.10, 6.10 - a]
    ]), 4

a, m = default_matrix()
true_eigenvalues, true_eigenvectors = np.linalg.eig(a)
# підставити значення з маткаду


print(printer(
    a,
    f"{__default_text_format} Starting matrix: ",default_style=
__default_print_format, formatting=__default_numbers_format
) + "\n")

"""
1. Порахувати нормальну форму фронебіуса (скрипт)
 * Показати M_i^-1, M_i, результуючу матрицю P.
   Порахувати характеристичне рівняння методом данилевського (скрипт)
3. Знайти власні числа характеристичного рівняння (маткад) -> (numpy)
4. Знайти власнтй вектор для кожного власного числа (скрипт)
5. Перевірити точність підстановкою у рівняння знайдені числа та вектори.
    A * x = lambda * x
"""

# Нормальна форма фронебіуса
mat_a_i = a.copy()
s = None


for i in range(1, m):
    idx = m - i
    mat_m = np.eye(m)

    row = [
        -mat_a_i[idx,j]/mat_a_i[idx,idx-1] # що тут вбіса відбувається
        if j != m - i - 1 else 1 / mat_a_i[idx, idx-1]
    for j in range(m)]
    mat_m[idx-1] = row

    mat_a_i = np.linalg.inv(mat_m) @ mat_a_i @ mat_m


    if s is None:
        s = mat_m.copy()
```

```
    else: s = s @ mat_m

    print(printer(
        mat_m,
        f"{__default_text_format} Matrix M{idx}", default_style=
__default_print_format, formatting=__default_numbers_format
    ) + "\n")
    print(printer(
        mat_a_i,
        f"{__default_text_format} Matrix A{i}",default_style=
__default_print_format, formatting=__default_numbers_format
    ) + "\n")

p = mat_a_i.copy()

print(printer(
    p, f"{__default_text_format} Matrix P, phronebius form: ",
default_style= __default_print_format, formatting=__default_numbers_format
) + "\n")
print(printer(
    s, f"{__default_text_format} Matrix S: ", default_style=
__default_print_format, formatting=__default_numbers_format
) + "\n")

# що таке форма фронебіуса
# що таке eigenvalues (власні значення)
v = np.array([-p[0,3], -p[0,2], -p[0,1], -p[0,0], 1])
lambda_values = np.roots(v[::-1]) # eigenvals
# що робить метод roots,
# як знаходяться власні значення з нормальної форми фронебіуса

print(printer(
    lambda_values.reshape(-1, 1)[:4], f"{__default_text_format}
Eigenvalues ", default_style= __default_print_format,
formatting=__default_numbers_format,
) + "\n")



# y
mat_y = np.array([
    [l**3, l**2, l, 1] for l in lambda_values
])
# чому значення для y рахуються як eigenvalues у якійсь степені?
print(printer(
    mat_y, f"{__default_text_format} Y: ", default_style=
__default_print_format, formatting=__default_numbers_format,
```

```
) + "\n")

# що таке eigenvectors (власні вектори)
# eigenvec
eigenvec = s @ mat_y.T
print(printer(
    eigenvec, f"{__default_text_format} Eigenvectors: ", default_style=
__default_print_format, formatting=__default_numbers_format,
) + "\n")


np_eigenvalues, np_eigenvectors = np.linalg.eig(a)
print(printer(
    np_eigenvalues.reshape(-1, 1)[:4], f"{__default_text_format} (numpy)
Eigenvalues ", default_style= __default_print_format,
formatting=__default_numbers_format,
) + "\n")


# чому eigenvalues через numpy у іншому порядку?
errors = []
for i, (lambda_i, v) in enumerate(zip(lambda_values, eigenvec.T), 1):
    errors.append(np.linalg.norm(a @ v - lambda_i * v))

print(printer(
     np.array(errors).reshape(-1, 1)[:4], f"{__default_text_format}
(numpy) eigenvectorc - (own) eigenvectors ", default_style=
__default_print_format, formatting=__default_numbers_format,
) + "\n")
```