In [2]:

```
import numpy as np
```

1. Найти с помощью NumPy SVD для матрицы

$$\begin{pmatrix} 1 & 2 & 0 \\ 0 & 0 & 5 \\ 3 & -4 & 2 \\ 1 & 6 & 5 \\ 0 & 1 & 0 \end{pmatrix}.$$

$$\begin{pmatrix} 1 & 2 & 0 \\ 0 & 0 & 5 \\ 3 & -4 & 2 \\ 1 & 6 & 5 \\ 0 & 1 & 0 \end{pmatrix}.$$

In [3]:

```
A = np.array([[1,2,0],[0,0,5],[3,-4,2],[1,6,5],[0,1,0]])
```

In [4]:

```
U, s, W = np.linalg.svd(A)
U
```

Out[4]:

```
array([[ 0.17056501,  0.15680918, -0.53077508, -0.79
905375, -0.16158397],
       [ 0.39287016, -0.52933945,  0.6134793 , -0.43
375771,  0.03082495],
       [-0.14366152, -0.82449256, -0.52379105,  0.14
049848,  0.07400343],
       [ 0.88843702,  0.06074346, -0.24655277,  0.37
755832, -0.06042632],
       [ 0.08125046,  0.10831843, -0.08231425, -0.10
524851,  0.98173958]])
```

In [5]:

```
# Транспонируем матрицу W
V = W.T

# s - список диагональных элементов, его нужно привести к виду ди
D = np.zeros_like(A, dtype=float)
D[np.diag_indices(min(A.shape))] = s
```

In [6]:

```
# Убедимся, что матрица U действительно ортогональна
print(np.dot(U.T, U))
```

```
[[ 1.00000000e+00  6.31672308e-17 -1.75301316e-16 -1
.17939618e-16
  -8.57750402e-18]
 [ 6.31672308e-17  1.00000000e+00  8.26028228e-17 -5
.17318224e-17
   3.82745166e-18]
 [-1.75301316e-16  8.26028228e-17  1.00000000e+00 -2
.71839651e-16
  -9.89547134e-17]
 [-1.17939618e-16 -5.17318224e-17 -2.71839651e-16  1
.00000000e+00
  -2.61324878e-17]
 [-8.57750402e-18  3.82745166e-18 -9.89547134e-17 -2
.61324878e-17
   1.00000000e+00]]
```

In [7]:

```
# Проведем проверку
print(np.dot(np.dot(U, D), V.T))
```

```
[[ 1.00000000e+00  2.00000000e+00  5.69432293e-16]
 [ 1.03164637e-15 -2.16649373e-15  5.00000000e+00]
 [ 3.00000000e+00 -4.00000000e+00  2.00000000e+00]
 [ 1.00000000e+00  6.00000000e+00  5.00000000e+00]
 [-1.05854814e-16  1.00000000e+00  5.52894695e-17]]
```

2. Для матрицы из предыдущего задания найти:

а) евклидову норму;

б) норму Фробениуса.

Евклидова норма матрицы равна евклидовой норме диагональной матрицы из ее сингулярных чисел DD. Максимальное значение полученного отношения будет равно максимальному сингулярному числу $\mu_{max}\mu_{max}$, и, принимая во внимание факт сортировки по убыванию сингулярных чисел, получим

$$\|A\|_E = \mu_1.$$

$$\|A\|_E = \mu_1.$$

In [14]:

```
D[0][0]
```

Out[14]:

```
8.824868854820442
```

Таким образом Евклидова норма равна 8.824868854820442

проверяем

In [9]:

```
np.linalg.norm(A, ord = 2)
```

Out[9]:

```
8.824868854820444
```

б)

Норма Фробениуса, определяемая как

$$\|A\|_F = \sqrt{\sum_{i=1}^{m} \sum_{j=1}^{n} a_{ij}^2}.$$

$$\|A\|_F = \sqrt{\sum_{i=1}^{m} \sum_{j=1}^{n} a_{ij}^2}.$$

In [11]:

```
m = 5
n = 3

a = []
for i in range(m):
    row_abs = [D[i][j]**2 for j in range(n)]
    a.append(sum(row_abs))
print(f'Норма Фробениуса матрицы A = {(sum(a))**0.5}')
```

```
Норма Фробениуса матрицы A = 11.045361017187261
```

проверка 1

In [12]:

```
A = np.array([[1,2,0],[0,0,5],[3,-4,2],[1,6,5],[0,1,0]])
m = 5
n = 3

a = []
for i in range(m):
    row_abs = [A[i][j]**2 for j in range(n)]
    a.append(sum(row_abs))
print(f'Норма Фробениуса матрицы A = {(sum(a))**0.5}')
```

```
Норма Фробениуса матрицы A = 11.045361017187261
```

проверка 2

In [13]:

```
np.linalg.norm(A)
```

Out[13]:

```
11.045361017187261
```

In [2]:

1. Найти с помощью NumPy SVD для матрицы

$$\begin{pmatrix} 1 & 2 & 0 \\ 0 & 0 & 5 \\ 3 & -4 & 2 \\ 1 & 6 & 5 \\ 0 & 1 & 0 \end{pmatrix}.$$

In [3]:

In [4]:

Out[4]:

```
array([[ 0.17056501,  0.15680918, -0.53077508, -0.79
905375, -0.16158397],
       [ 0.39287016, -0.52933945,  0.6134793 , -0.43
375771,  0.03082495],
       [-0.14366152, -0.82449256, -0.52379105,  0.14
049848,  0.07400343],
       [ 0.88843702,  0.06074346, -0.24655277,  0.37
755832, -0.06042632],
       [ 0.08125046,  0.10831843, -0.08231425, -0.10
524851,  0.98173958]])
```

In [5]:

In [6]:

```
[[ 1.00000000e+00  6.31672308e-17 -1.75301316e-16 -1
.17939618e-16
  -8.57750402e-18]
 [ 6.31672308e-17  1.00000000e+00  8.26028228e-17 -5
.17318224e-17
   3.82745166e-18]
 [-1.75301316e-16  8.26028228e-17  1.00000000e+00 -2
.71839651e-16
  -9.89547134e-17]
 [-1.17939618e-16 -5.17318224e-17 -2.71839651e-16  1
.00000000e+00
  -2.61324878e-17]
 [-8.57750402e-18  3.82745166e-18 -9.89547134e-17 -2
.61324878e-17
   1.00000000e+00]]
```

In [7]:

```
[[ 1.00000000e+00  2.00000000e+00  5.69432293e-16]
 [ 1.03164637e-15 -2.16649373e-15  5.00000000e+00]
 [ 3.00000000e+00 -4.00000000e+00  2.00000000e+00]
 [ 1.00000000e+00  6.00000000e+00  5.00000000e+00]
 [-1.05854814e-16  1.00000000e+00  5.52894695e-17]]
```

2. Для матрицы из предыдущего задания найти:

   а) евклидову норму;

   б) норму Фробениуса.

Евклидова норма матрицы равна евклидовой норме диагональной матрицы из ее сингулярных чисел D. Максимальное значение полученного отношения будет равно максимальному сингулярному числу μ_{max}, и, принимая во внимание факт сортировки по убыванию сингулярных чисел, получим

$$\|A\|_E = \mu_1.$$

In [8]:

```

```

Out[8]:

```
array([[8.82486885, 0.        , 0.        ],
       [0.        , 6.14060608, 0.        ],
       [0.        , 0.        , 2.53271528],
       [0.        , 0.        , 0.        ],
       [0.        , 0.        , 0.        ]])
```

Таким образом Евклидова норма равна 8.82486885

проверяем

In [9]:

Out[9]:

8.824868854820444

б)

Норма Фробениуса, определяемая как

$$\|A\|_F = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n} a_{ij}^2}.$$

In [11]:

Норма Фробениуса матрицы A = 11.045361017187261

проверка 1

In [12]:

Норма Фробениуса матрицы A = 11.045361017187261

проверка 2

In [13]:

Out[13]:

11.045361017187261