Контрольная работа 1

Вариант 5

Богдан Владислав

1. Найти сингулярное разложение для матрицы A:

$$A = \begin{bmatrix} 27 & 23 & -4 \\ -15 & -6 & 29 \\ 12 & 29 & -4 \\ 15 & 20 & 6 \end{bmatrix}$$

Будем искать сингулярное разложение в виде $A = USV^*$.

Сначала найдём матрицу $B = AA^T$ и её собственные значения. Сингулярные значения матрицы A есть квадратные корни из соответствующих собственных значений матрицы B. S формируется как диагональная матрица с сингулярными значениями на диагонали. Матрица U составляется их соответствующих собственных векторов.

In [104]:

```
import numpy as np
from numpy import linalg
```

In [131]:

```
A = np.array([
    [27, 23, -4],
    [-15, -6, 29],
    [12, 29, -4],
    [15, 20, 6]
])

B = A @ A.transpose()
S_squared, U = np.linalg.eig(B)

S = np.sqrt(S_squared)
```

```
/usr/local/lib/python3.6/site-packages/ipykernel_launcher.py:11: Run
timeWarning: invalid value encountered in sqrt
  # This is added back by InteractiveShellApp.init_path()
```

Теперь справедливое выражение:

$$SV^* = U^{-1}A$$

Поскольку S - диагональная матрица, то матрица V^* получается путем деления строк матрицы $U^{-1}A$ на скаляры из соответствующих строк матрицы S.

In [132]:

```
V_pre = (np.linalg.inv(U) @ A).transpose()

V_pre[:,0] /= S[0]
V_pre[:,1] /= S[1]
V_pre[:,2] /= S[2]

V = np.delete(V.transpose(), 3, 0)
```

Получили следующие значения:

In [136]:

```
print("U:", U)

S_matrix = np.zeros((3,4))
S_matrix[:,:-1] = np.diag(S[:3])
S_matrix = S_matrix.transpose()
print("S:", S_matrix)

print("V*:", V)
```

```
U: [[ 0.63189875 -0.08883085  0.58888347  0.49601341]
 [-0.37563614 -0.88733767  0.04753892  0.26319079]
 [ 0.53874955 -0.20725599 -0.78841919  0.21257718]
 [ 0.41152202 -0.40222616  0.17132277 -0.7996951 ]]
S: [[ 55.42564122   0.            0.        ]
 [  0.          29.1330773    0.        ]
 [  0.           0.          10.82876273]
 [  0.           0.           0.        ]]
V*: [[ 0.63749616  0.73326469 -0.23647736]
 [ 0.08207742 -0.36982123 -0.92547045]
 [ 0.76606915 -0.57057441  0.29594409]]
```

Мы можем проверить полученный результат:

In [137]:

```
print("A:", A)
print("USV*:", U @ S_matrix @ V)
```

```
A: [[ 27  23  -4]
 [-15  -6  29]
 [ 12  29  -4]
 [ 15  20   6]]
USV*: [[ 27.  23.  -4.]
 [-15.  -6.  29.]
 [ 12.  29.  -4.]
 [ 15.  20.   6.]]
```

2. Найти какое-нибудь разложение полного ранга для A (из первого задания) и псевдообратную матрицу.

Разложение полного ранга - такое представление $A = BC$, где B имеет размерность $4 \times r$, а A - размерность $r \times 3$, где r - ранг матрицы A. Найдём его:

In [147]:

```
np.linalg.det(A[:3])
```

Out[147]:

```
-13982.999999999996
```

Минор третьего (одна из размерностей матрицы) порядка не равен 0, соответственно, ранг матрицы $r = 3$.

В качестве C можно взять первые три строки матрицы A - они являются линейно независимыми. Тогда $B = AC^+$, где C^+ - псевдообратная к C матрица.

In [148]:

```
C = A[:3]
C_pseudo = C.transpose() @ np.linalg.inv(C @ C.transpose())
B = A @ C_pseudo
```

Проверим, что найденные матрицы B, C действительно удовлетворяют условию:

In [149]:

```
print("A: ", A)
print("A = BC: ", B @ C)
```

```
A:  [[ 27  23  -4]
 [-15  -6  29]
 [ 12  29  -4]
 [ 15  20   6]]
A = BC:  [[ 27.  23.  -4.]
 [-15.  -6.  29.]
 [ 12.  29.  -4.]
 [ 15.  20.   6.]]
```

Теперь псевдообратная матрица для A может быть найдена по формуле $A^+ = C^T(CC^T)^{-1}(B^TB)^{-1}B^T$:

In [150]:

```
A_pseudo = C.transpose() @ np.linalg.inv(C @ C.transpose()) @ np.linalg.inv(B.tr
anspose() @ B) @ B.transpose()
print("A+: ", A_pseudo)
```

```
A+:  [[ 0.04867765 -0.00345733 -0.05016317  0.0157201 ]
 [-0.02154118  0.00378964  0.05130075  0.00152315]
 [ 0.01621971  0.03108995 -0.01726177  0.01570389]]
```

3. Решение системы линейных уравнений

Дана система линейных уравнений $AX = B$. Найдём ранг матрицы A:

In [168]:

```
A = np.array([[-28, 3, -10, 39],
     [-25, -10, -22, 27],
     [19, 14, 14, -17],
     [-9, -26, 21, 4]])
B = np.array([-21, -18, 13, 13])

np.linalg.matrix_rank(A)
```

Out[168]:

```
4
```

Отсюда следует, что данная система линейных уравнений имеет ровно одно решение, которое может быть посчитано точными методами:

In [171]:

```
np.linalg.solve(A, B)
```

Out[171]:

```
array([ 4.61898962, -1.735914  , -0.10024359,  2.8855645 ])
```

Или, например, с использованием обратной матрицы:

In [172]:

```
np.linalg.inv(A) @ B
```

Out[172]:

```
array([ 4.61898962, -1.735914  , -0.10024359,  2.8855645 ])
```

Нашли точное решение системы линейных уравнений.

4. Наилучшее равномерное приближение

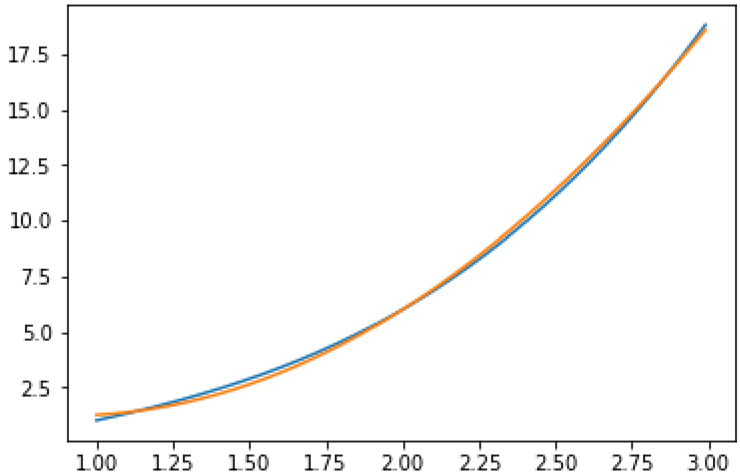
Задан многочлен $x^3 - 2x^2 + 4x - 2$ на отрезке $[1, 3]$ - требуется найти наилучшее равномерное приближение этого многочлена многочленом второй степени. Сделаем следующую замену переменной: $y = x - 2; x = y + 2$. Получим многочлен $f(y) = y^3 + 4y^2 + 8y + 6$, определённый уже на отрезке $[-1, 1]$. Известно, что наименьшее отклонение от нуля на отрезке $[-1, 1]$ среди всех многочленов третьей степени имеет многочлен $\frac{1}{2^2}T_3(y) = \frac{1}{4}(4y^3 - 3y) = y^3 - \frac{3}{4}y$. Таким образом, для многочлена второй степени $g(y)$, который наилучшим образом приближает многочлен $f(y)$ справедливо:

$$g(y) = f(y) - (y^3 - \frac{3}{4}y) = y^3 + 4y^2 + 8y + 6 - y^3 + \frac{3}{4}y = 4y^2 + \frac{35}{4}y + 6$$
$$g_1(x) = g(x-2) = 4(x-2)^2 + \frac{35}{4}(x-2) + 6 = 4x^2 - \frac{29}{4}x + \frac{9}{2}, x \in [1, 3]$$

Посмотрим на графики функций $f(y)$ и $g_1(y)$

In [159]:

```
import matplotlib.pyplot as plt
plt.plot(np.arange(1., 3., 0.01), list(map(lambda x: x**3-2*x**2+4*x-2, np.arang
e(1., 3., 0.01))))
plt.plot(np.arange(1., 3., 0.01), list(map(lambda x: 4*x**2-7.25*x+4.5, np.arang
e(1., 3., 0.01))))
plt.show()
```



Многочлен второй степени действительно приближает многочлен третьей степени из условий задачи!

5. Единичная окружность относительно нормы

In [1]:

```
from IPython.display import Image
Image("img/5.jpg")
```

Out[1]:

5 Единичная окружность относительно нормы.

(Богдан

$3(-3z+4y)^2+4(-2z-2xq)^2q+2(-4y+4xq)^2=1$

$3(9z^2+16y^2-24yz)+4q(4z^2+4x^2q^2+2xzq)+2(16y^2+16x^2q^2-32yxq)$

$27z^2+48y^2-72yz+16q\cdot z^2+16q^3\cdot x^2+8q^2xz+32y^2+32q^2\cdot x^2-64q\cdot xy$

$x^2(16q^3+32q^2)+y^2(48+32)+z^2(27+16q)+xy(64q)+xz(8q^2)+yz(-72)$

$$A=\begin{pmatrix} 16q^3+32q^2 & 32q & 4q^2 \\ 32q & 80 & -36 \\ 4q^2 & -36 & 27+16q \end{pmatrix}$$

$\Delta_1 = 16q^3+32q^2>0$

$q^2(32+16q)>0$

$16q^2(2+q)>0$

$q\neq 0. \quad 2+q>0$

$q>-2.$

$\Delta_2 = 80\cdot 16q^2(q+2)-32^2q^2=$

$=16^2q^2(5q+10-4)>0$

$q\neq 0. \quad 5q+6>0.$

$5q>-6.$

$q>-\frac{6}{5}.$

$\Delta_3=\det A=(27+16q)(16^2q^2(5q+6))+36\begin{vmatrix}16q^3+32q^2 & 32q\\ 4q^2 & -36\end{vmatrix}+$

$+4q^2\begin{vmatrix}32q & 80\\ 4q^2 & -36\end{vmatrix}=16^2q^2(27+16q)(5q+6)+36\cdot(-36(16q^3+32q^2)-32\cdot 4q^3)+$

$+4q^2(-36\cdot 32q-80\cdot 4q^2).$

$q\neq 0$ и $/=q^2$, $/=32$

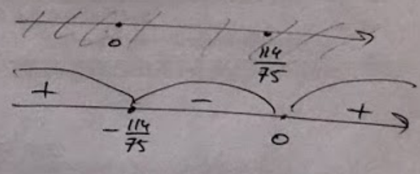
$8(27+16q)(5q+6)+18(-36(q+2)-8q)+(-36\cdot 4q-40q^2)>0$

$/=8$: $(27+16q)(5q+6)+9(-9(q+2)-2q)+(-18q-5q^2)>0.$

$135q+162+80q^2+96q+9(-11q-18)-18q-5q^2>0.$

$75q^2+114q>0.$

$q(75q+114)>0.$



$q<-\frac{114}{75}\cup q>0$

Получаем $q\neq 0 \cap q>-2 \cap q>-\frac{6}{5}$

$\cap(q>0\cup q<-\frac{114}{75})$

$\Rightarrow q>0$ – при таких q матрица положительно определена и мы имеем единичную окружность.