Domaći zadatak broj 9

Ime i prezime studenta:Nastasija Stankovic Broj indeksa:17955

Uputstvo: 1. Pre početka izrade promenite ime datoteke u Domaci9_ime_prezime. (ubacite svoje ime i prezime) 2. Popunite ćeliju ispod naslova odgovarajućim podacima. 3. Za rešavanje zadataka, ukoliko je potrebno, otvorite ispod teksta zadataka dodatne ćelije za upisivanje tekstualnog odgovora (Markdown) ili programskog koda (Code). 4. Nakon završetka izrade rešenja Notebook dokument sačuvati u pdf formatu i proslediti ga nastavniku. To možete da uradite ili kroz Teams ili na mail adresu jovana.dzunic@elfak.ni.ac.rs

In [4]:
```
import numpy as np
import numpy.random as rndm
```

Zadatak 1. a) Slične simetrične matrice su iste definitnosti. Dokazati.

(5 poena)

Definitnost neke matrice zavisi od sopstvenih vrednosti te matrice.Sopstvene vrednosti slicnih matrica su jednake,iz toga sledi da ce i simetricne matrice biti iste definitosti.

b) Ako je simetrična matrica A pozitivno definitna onda je takva i matrica A^{-1}. Dokazati.

(5 poena)

Posto je matrica A pozitivno definisana iz toga sledi da su sopstvene vrednosti te matrice pozitivne. Mnozenjem sopstvenog vektora V sa inverznom matricom dobija se: $A^{-1}v = \dfrac{1}{\lambda}v$. Sopstvena vrednost matrice A^{-1} je $\frac{1}{\lambda}$ pozitivna za svako λ,pa mozemo zakljuciti da je i matrica A^{-1} pozitivno definitna.

Zadatak 2. Neka je x^TAx kvadratna forma, v jedinični sopstveni vektor i λ odgovarajuća sopstvena vrednost matrice A. Dokazati da je $v^TAv = \lambda$.

(5 poena)

Mnozenjem matrice A i jedinicnog sopstvenog vektora v dobijamo: $Av = \lambda v$

$v^TAv = v^T\lambda v$

$v^TAv = \lambda v^Tv$

$v^TAv = \lambda||v||$ **, norma je jednaka 1.**

$v^TAv = \lambda$ **, sto se i trazilo da dokazemo.**

Zadatak 3. Neka su $v_1, v_2, \dots, v_n$ konjugovani vektori pozitivno definitne (simetrične) matrice A. Opisati primenu razvoja

$$v = \alpha_1 v_1 + \alpha_2 v_2 + \cdots + \alpha_n v_n$$

na rešenje sistema linearnih jednačina $Ax = b$.

(10poena)

Iz izraza: $||x-v||_A^2 = (x-v)^TA(x-v)$ Sredjivanjem dobijamo:

$||x-v||_A^2 = (x^TA - v^T)A(x-v)$

$||x-v||_A^2 = (x^TA - v^TA)(x-v)$

$||x-v||_A^2 = x^TAx - x^TAv - v^TAx + v^TAv$

Matrica A je simetricna iz toga sledi da je $x^TAv = v^TAx$.

$||x-v||_A^2 = x^TAx - 2x^TAv + v^TAv$

$f(x) = x^TAx - 2x^TAv = x^TAx - 2x^Tb = x^TAx - 2b^Tx$

Dobijamo da dobijeni izraz ne zavisi od x pa dalje trazimo minimum funkcije.Da bismo nasli minimum potrebno je da vektor x napisemo u konjugovanoj bazi $v_1, v_2, \dots, v_n$:

$x = x_1 \cdot v_1 + x_2 \cdot v_2 + \cdots + x_n \cdot v_n$

$$f(x) = (x_1 \cdot v_1 + x_2 \cdot v_2 + \cdots + x_n \cdot v_n)^T A (x_1 \cdot v_1 + x_2 \cdot v_2 + \cdots + x_n \cdot v_n) - 2b^T(x_1 \cdot v_1 + x_2 \cdot v_2 + \cdots + x_n \cdot v_n)$$

$$f(x) = \sum_{i=1}^{n} x_i(x_i - 2b^Tv_i)$$

$$f'(x) = \sum_{i=1}^{n} 2x_i - 2b^Tv_i$$

$f(x) = \sum_{i=1}^{n} 2x_i - 2b^Tv_i$

Vektor v je direktno zavisi od vektora v_i,pa nam poznavanje vektora v_i pojednostavljuje resavanje linearnog sistema jednacina.

Zadatak 4. a) Odrediti glavne ose i klasifikovati krivu drugog reda u 2D. Dati njen grafički prikaz.

$$2x^2 - 4xy + y^2 - 3x + y - 9 = 0.$$

(10 poena)

$2x^2 - 4xy + y^2 - 3x + y = 9$

$\begin{bmatrix} x & y \end{bmatrix} \begin{bmatrix} 2x-2y & -2x+y \end{bmatrix} + \begin{bmatrix} -3 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = 9$

$\begin{bmatrix} x & y \end{bmatrix} \begin{bmatrix} 2 & -2 \\ -2 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} -3 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = 9$

Determinanta matrice $\begin{bmatrix} 2 & -2 \\ -2 & 1 \end{bmatrix}$ je $det(A) = 2 - ((-2)*(-2)) = -2$,zakljucujemo da matrica nije definitna.

b) Odrediti glavne ose i klasifikovati površ drugog reda u 3D.

$$2x^2 - y^2 + z^2 - 3xy + yz - 9x + 3 = 0.$$

(5 poena)

$2x^2 - y^2 + z^2 - 3xy + yz - 9x + 3 = \begin{bmatrix} x & y & z \end{bmatrix} \begin{bmatrix} 2 & -3/2 & 0 \\ -3/2 & -1 & -1/2 \\ 0 & 1/2 & 1 \end{bmatrix} + \begin{bmatrix} -9 & 0 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} + 3 = 0$

Sredjivanjem dobijamo:

$-(\dfrac{x''}{a})^2 + (\dfrac{y''}{b})^2 + (\dfrac{z''}{c})^2 = 1$

Ova jednacina predstavlja jednokrilni hiperboloid.

In [5]:
```
A=np.array([-9,0,0])
B=np.array([[2,-3/2,0],[-3/2,-1,-1/2],[0,1/2,1]])
k1,k2=np.linalg.eigh(B)
print(k1)
print()
print(k2)
print()
print(A@k2)
```

```
[-1.70058459  1.05629958  2.64428502]

[[ 0.37024342  0.17510326  0.91228212]
 [ 0.9134114   0.11016335 -0.39184647]
 [-0.16911364  0.97836746 -0.11915406]]

[-3.33219079 -1.57592933 -8.21053904]
```

Zadatak 5. Kreirati Python kod za rešenje sistema linearnih jednačina $Ax = b$ upotrebom metoda konjugovanih gradijenata polazeći od vektora $x_0 = \theta$ i toleranscije greške $\varepsilon = 1e-10$.

$$\begin{bmatrix} 10 & 1 & 2 & 3 & 4 \\ 1 & 9 & -1 & 2 & -3 \\ 2 & -1 & 7 & 3 & -5 \\ 3 & 2 & 3 & 12 & -1 \\ 4 & -3 & -5 & -1 & 15 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ t \\ s \end{bmatrix} = \begin{bmatrix} 12 \\ -27 \\ 14 \\ -17 \\ 12 \end{bmatrix}.$$

Pseudo kod za metod konjugovanih gradijenata dat je u nastavku:

- **ulaz:** x_0, A, B, ε
- m, n **su dimenzije matrice** A
- $r_0 = b - Ax_0$
- $v_0 = r_0$
- $k = 0$
- **dok je** $(k \leq n-1)$ **i** $v_k \neq \theta$ **i** $\|r_k\| > \varepsilon$ **radi:**
  - $t_k = \dfrac{r_k^T r_k}{v_k^T A v_k}$
  - $x_{k+1} = x_k + t_k v_k$
  - $r_{k+1} = r_k - t_k A v_k$
  - **Ako je** $\|r_{k+1}\| > \varepsilon$ **radi:**
    - $s_k = \dfrac{r_{k+1}^T r_{k+1}}{r_k^T r_k}$
    - $v_{k+1} = r_{k+1} + s_k v_k$
  - $k++$
- **izlaz** $x_k,\ r_k$

  (20 poena)

In [6]:
```
A=np.array([[10,1,2,3,4],[1,9,-1,2,-3],[2,-1,7,3,-5],[3,2,3,12,-1],[4,-3,-5,-1,15]])
B=np.array([[12],[-27],[14],[-17],[12]])
x0=np.array([[0],[0],[0],[0],[0]])
r0=B-A@x0
v0=r0
k=0
e=1e-10
temp=np.array([[0],[0],[0],[0],[0]])
while(k<=10 and (v0!=temp).all() and np.linalg.norm(r0)>e):
    t0=(np.transpose(r0)@r0)/(np.transpose(v0)@(A@v0))
    x0=x0+t0[0][0]*v0
    r=r0
    r0=r0-t0[0][0]*(A@v0)
    if np.linalg.norm(r0)>e:
        s=(np.transpose(r0)@r0)/(np.transpose(r)@r)
        v0=r0+s[0][0]*v0
    k+=1
print(x0)
print()
print (r0)
```

```
[[ 1.]
 [-2.]
 [ 3.]
 [-2.]
 [ 1.]]

[[ 4.44089210e-16]
 [-5.86336535e-16]
 [ 5.55111512e-16]
 [-2.22044605e-16]
 [-2.22044605e-16]]
```