

METODE NUMERICE: Tema #2

Vectori si valori proprii

Termen de predare: 23 APRILIE 2017

Ultima actualizare: 2 aprilie

Obiectivele Temei

Obiectivele generale ale acestei teme sunt:

- calcularea numarului de valori proprii ale unei matrice tridiagonale simetrice care sunt mai mici decat o valoare data;
- calcularea unei limite superioare, respectiv inferioare intre care sunt situate toate valorile proprii ale unei matrice tridiagonale simetrice;
- separarea in intervale a valorilor proprii ale unei matrice tridiagonale simetrice folosind metoda sirului lui Sturm;
- calcularea valorilor si a vectorilor proprii pentru o matrice tridiagonala simetrica.

Descriere Generala

Problema calculului valorilor proprii ale unei matrice simetrice oarecare poate fi redusa la calculul valorilor proprii ale unei matrice tridiagonale simetrice care are aceleasi valori proprii (o astfel de matrice se poate obtine utilizand transformarea Householder).

O matrice tridiagonala simetrica $A \in \mathbb{R}^{n \times n}$, $A = \{a_{ij}\}$ contine elemente nenule pe diagonala principala si pe diagonalele adiacente (numite supradiagonala, respectiv subdiagonala) cu diagonala principala. Celelalte elemente dintr-o matrice tridiagonala simetrica sunt nule, adica $a_{ij} = 0$ pentru $|i - j| > 1$. Mai mult, o matrice tridiagonala simetrica are proprietatea ca $a_{ij} = a_{ji}$ pentru $|i - j| = 1$. Cu alte cuvinte, supradiagonala este egala cu subdiagonala.

Se constata ca o matrice tridiagonala simetrica este o matrice rara. Prin urmare, o matrice tridiagonala simetrica se pastreaza eficient in memorie folosind doi vectori: un vector reprezentand diagonala principala si celalalt vector reprezentand supradiagonala. Luand in calcul aceasta observatie, vom considera, in cele ce urmeaza, o matrice tridiagonala simetrica $A \in \mathbb{R}^{n \times n}$ de forma:

$$A = \begin{bmatrix} d_1 & s_1 & & & & \\ s_1 & d_2 & s_2 & & & \\ & \ddots & \ddots & \ddots & & \\ & & \ddots & \ddots & \ddots & \\ & & & s_{n-2} & d_{n-1} & s_{n-1} \\ & & & & s_{n-1} & d_n \end{bmatrix} \quad (1)$$

Valorile proprii asociate cu matricea tridiagonală simetrică A pot fi obținute calculând rădăcinile polinomului caracteristic $P_n(\lambda)$ definit prin relația:

$$P_n(\lambda) = |A - \lambda I| = \begin{vmatrix} d_1 - \lambda & s_1 & & & \\ s_1 & d_2 - \lambda & s_2 & & \\ & \ddots & \ddots & \ddots & \\ & & s_{n-2} & d_{n-1} - \lambda & s_{n-1} \\ & & & s_{n-1} & d_n - \lambda \end{vmatrix} \quad (2)$$

Relația 2 nu constituie o metodă eficientă de a determina polinomul caracteristic $P_n(\lambda)$ deoarece calcularea determinantului implică efectuarea de multe operații. Polinomul caracteristic $P_n(\lambda)$ poate fi calculat, în mod eficient, folosind următoarea relație de recurență:

$$P_i(\lambda) = (d_i - \lambda)P_{i-1}(\lambda) - s_{i-1}^2 P_{i-2}(\lambda), \quad i = 2 : n \quad (3)$$

unde $P_0(\lambda) = 1$ și $P_1(\lambda) = d_1 - \lambda$.

Sirul de polinoame $P_0(\lambda), P_1(\lambda), P_2(\lambda), \dots, P_n(\lambda)$ formează un sir Sturm ([1]). Enunțăm, în continuare, o proprietate asociată unui sir Sturm care ne permite să determinăm numărul de valori proprii asociate cu matricea tridiagonală simetrică A , mai mici decât un număr real dat.

Proprietate sir Sturm: Fie sirul Sturm $P_0(\lambda), P_1(\lambda), P_2(\lambda), \dots, P_n(\lambda)$ și o valoare α . Considerăm valoarea polinoamelor din acest sir Sturm pentru α ca fiind $P_0(\alpha), P_1(\alpha), P_2(\alpha), \dots, P_n(\alpha)$. Numărul rădăcinilor ecuației $P_n(\lambda) = 0$, mai mici decât α , este egal cu numărul schimbărilor de semn ale sirului numeric $P_0(\alpha), P_1(\alpha), P_2(\alpha), \dots, P_n(\alpha)$.

Observație 1: În stabilirea numărului de schimbări de semn ale sirului numeric $P_0(\alpha), P_1(\alpha), P_2(\alpha), \dots, P_n(\alpha)$ se compară $P_i(\alpha)$ cu $P_{i+1}(\alpha)$ pentru $i = 0 : n-1$.

Observație 2: Dacă $P_i(\alpha) = 0$, semnul pentru $P_i(\alpha)$ se va considera ca fiind semnul opus lui $P_{i-1}(\alpha)$.

Următoarea teoremă, datorată lui Gershgorin ([2]), ne permite să obținem o limită inferioară λ_{min} și una superioară λ_{max} între care sunt situate toate valorile proprii ale unei matrice tridiagonale simetrice. Cu alte cuvinte, toate valorile proprii sunt cuprinse în intervalul $[\lambda_{min}, \lambda_{max}]$. Vom enunța această teoremă pentru cazul general când matricea patratică este de forma oarecare, dar teorema rămâne valabilă și pentru o matrice tridiagonală simetrică.

Teorema lui Gershgorin: Fie $B \in \mathbb{R}^{n \times n}$, $B = \{b_{ij}\}$ o matrice simetrică de forma oarecare. Limita inferioară λ_{min} și cea superioară λ_{max} a valorilor proprii ale matricei B sunt date de relațiile:

$$\lambda_{min} = \min_{i=1:n} (b_{ii} - t_i), \quad \lambda_{max} = \max_{i=1:n} (b_{ii} + t_i) \quad (4)$$

unde

$$t_i = \sum_{j=1, j \neq i}^n |b_{ij}| \quad (5)$$

Cerinta 1 (15p)

Scrieti o functie numita *ValoriPolinoame* care calculeaza valoarea fiecarui polinom din sirul de polinoame $P_0(\lambda)$, $P_1(\lambda)$, $P_2(\lambda)$, ..., $P_n(\lambda)$ pentru o valoare reala λ data. Implementarea acestei functii se va face in fisierul *ValoriPolinoame.m*. Functia *ValoriPolinoame* are ca intrare urmatoarele argumente in aceasta ordine: diagonala principala $d = [d_1, d_2, \dots, d_n]$, supradiagonala $s = [s_1, s_2, \dots, s_{n-1}]$ si un numar real *val_lambda* pentru care se calculeaza valoarea polinoamelor $P_0(\lambda)$, $P_1(\lambda)$, $P_2(\lambda)$, ..., $P_n(\lambda)$. Functia are ca rezultat vectorul $P = [P_0(val_lambda), P_1(val_lambda), P_2(val_lambda), \dots, P_n(val_lambda)]$ de lungime $n + 1$.

```

1 function P = ValoriPolinoame(d, s, val_lambda)
2     % Intrari:
3     % -> d:  diagonala principala a matricei tridiagonale simetrice;
4     % -> s:  supradiagonala matricei tridiagonale simetrice;
5     % -> val_lambda: valoare pentru lambda.
6     % Iesiri:
7     % -> P: vectorul P = [P0(val_lambda) P1(val_lambda), ..., Pn(val_lambda)].

```

Listing 1: ValoriPolinoame.m

Cerinta 2 (15p)

Folosind functia *ValoriPolinoame* si proprietatea unui sir Sturm enuntata in sectiunea Descriere Generala, scrieti o functie numita *NrValProprii* pentru a determina numarul de valori proprii ale unei matrice tridiagonale simetrice care sunt mai mici decat o valoare reala λ data. Implementarea functiei *NrValProprii* se va face in fisierul *NrValProprii.m*. Functia *NrValProprii* are ca intrare urmatoarele argumente in aceasta ordine: *d*, *s* si *val_lambda* (cu aceeasi interpretare ca la Cerinta 1). Functia are ca rezultat un numar natural reprezentand numarul de valori proprii ale matricei tridiagonale simetrice mai mici decat *val_lambda*.

```

1 function numvp = NrValProprii(d, s, val_lambda)
2     % Intrari:
3     % -> d:  diagonala principala a matricei tridiagonale simetrice;
4     % -> s:  supradiagonala matricei tridiagonale simetrice;
5     % -> val_lambda: valoare pentru lambda.
6     % Iesiri:
7     % -> numvp: numarul de valori proprii mai mici decit val_lambda.

```

Listing 2: NrValProprii.m

Cerinta 3 (15p)

Folosind teorema lui Gershgorin enuntata in sectiunea Descriere Generala, scrieti urmatoarea functie numita *LimiteValProprii* pentru a determina limitele incadratoare (limita inferioara, respectiv superioara) a valorilor proprii ale unei matrice tridiagonale simetrice. Implementarea functiei *LimiteValProprii* se va face in fisierul *LimiteValProprii.m*. Functia *LimiteValProprii* are ca intrare urmatoarele argumente in aceasta ordine: *d* si *s* (cu aceeasi interpretare ca la Cerinta 1). Functia are ca rezultat limita inferioara si cea superioara intre care sunt cuprinse toate valorile proprii ale matricei tridiagonale simetrice.

```

1 function [limita_inf, limita_sup] = LimiteValProprii(d, s)
2     % Intrari:
3     % -> d:  diagonala principala a matricei tridiagonale simetrice;
4     % -> s:  supradiagonala matricei tridiagonale simetrice.
5     % Iesiri:
6     % -> [limita_inf limita_sup]: limitele incadratoare pentru valorile proprii ale
    % matricei tridiagonale simetrice.

```

Listing 3: LimiteValProprii.m

Cerinta 4 (15p)

Folosind functiile *LimiteValProprii* si *NrValProprii*, scrieti o funcție numita *IntervaleValProprii* pentru a separa cele mai mici m valori proprii asociate cu o matrice tridiagonala simetrica. Separarea celor mai mici m valori proprii presupune determinarea unui sir numeric $r_1, r_2, \dots, r_m, r_{m+1}$ astfel incat:

- $r_1 < r_2 < \dots < r_m < r_{m+1}$;
- Fiecare interval (r_i, r_{i+1}) , $i = 1 : m$ contine o singura valoare proprie λ_i astfel incat cea mai mica valoare proprie a matricei tridiagonale simetrice se afla in intervalul (r_1, r_2) , urmatoarea valoare proprie cea mai mica se afla in intervalul (r_2, r_3) s.a.m.d.

Implementarea functiei *IntervaleValProprii* se va face in fisierul *IntervaleValProprii.m*. Functia *IntervaleValProprii* are ca intrare urmatoarele argumente in aceasta ordine: d, s (cu aceeasi interpretare ca la Cerinta 1) si m . Daca m depaseste lungimea vectorului d atunci trebuie setata valoarea pentru m la lungimea vectorului d .

```

1 function r = IntervaleValProprii(d, s, m)
2     % Intrari:
3     % -> d: diagonala principala a matricei tridiagonale simetrice;
4     % -> s: supradiagonala matricei tridiagonale simetrice;
5     % -> m: limita pentru numarul de valori proprii.
6     % Iesiri:
7     % -> r: un vector r = [r1, r2, ..., rm, rm+1] de dimensiune m+1.

```

Listing 4: IntervaleValProprii.m

Indicatie: Elementele sirului r se calculeaza in aceasta ordine: r_1 , apoi r_{m+1}, r_m, r_{m-1} s.a.m.d. pana se calculeaza si r_2 . Mai intai, folosind functia *LimiteValProprii*, se determina limita inferioara *limita_inf* si cea superioara *limita_sup* intre care se afla toate valorile proprii (inclusiv cele mai mici m valori proprii) ale matricei tridiagonale simetrice. Elementul r_1 se va initializa cu valoarea *limita_inf*. Mai mult, se introduce o variabila auxiliara r_{m+2} cu valoarea *limita_sup*. Variabila auxiliara r_{m+2} este folosita doar pentru a calcula elementul r_{m+1} si nu trebuie salvata in vectorul r .

Pentru determinarea elementelor r_{k+1} , $k = m : -1 : 1$ se parcurg urmatoarii pasi:

1. Se calculeaza mijlocul intervalului $[r_1, r_{k+2}]$ folosind relatia:

$$mij = (r_{k+2} + r_1)/2 \quad (6)$$

2. Se calculeaza lungimea intervalului $[mij, r_{k+2}]$ folosind relatia :

$$h = r_{k+2} - mij = r_{k+2} - (r_1 + r_{k+2})/2 = (r_{k+2} - r_1)/2 \quad (7)$$

3. Se calculeaza numarul de valori proprii mai mici decat valoarea mij

$$numvp = NrValProprii(d, s, mij) \quad (8)$$

4. Se actualizeaza valoarea mij astfel:

$$4.1. \quad h = h/2$$

$$4.2. \quad \text{daca } numvp < k \text{ atunci}$$

$$4.3. \quad mij = mij + h$$

$$4.4. \quad \text{altfel daca } numvp > k \text{ atunci}$$

$$4.5. \quad mij = mij - h$$

5. Se repeta algoritmul de la pasul 3 pana cand $numvp = k$.

6. Se obtine astfel valoarea petru r_{k+1} :

$$r_{k+1} = mij \quad (9)$$

Cerinta 5 (15p)

Implementati o functie numita *CalculezValProprii* pentru a determina cu o anumita precizie cele mai mici m valori proprii ale unei matrice tridiagonale simetrice. Implementarea functiei *CalculezValProprii* se va face in fisierul *CalculezValProprii.m*. Functia *CalculezValProprii* are ca intrare urmatoarele argumente in aceasta ordine: d , s , m (cu aceeasi interpretare ca la Cerinta 4) si precizia tol folosita in metoda bisectiei. Rezultatul functiei este un vector $vp = [\lambda_1, \lambda_2, \dots, \lambda_{m-1}, \lambda_m]$ de dimensiune m , care contine cele mai mici m valori proprii. Hint: folositi metoda bisectiei si din functiile anterioare de la cerintele 1-4 (ramane ca voi sa va dati seama de ce functie/functii aveti nevoie).

```
1 function vp = CalculezValProprii(d, s, m, tol)
2     % Intrari:
3     % -> d: diagonala principala a matricei tridiagonale simetrice;
4     % -> s: supradiagonala matricei tridiagonale simetrice;
5     % -> m: limita pentru numarul de valori proprii;
6     % -> tol: precizia determinarii.
7     % Iesiri:
8     % -> vp: vector de valori proprii, de dimensiune m.
```

Listing 5: CalculezValProprii.m

Cerinta 6 (15p)

Folosind metoda puterii inverse, scrieti o functie numita *PutereInv*. Implementarea functiei *PutereInv* se va face in fisierul *PutereInv.m*. Functia *PutereInv* are ca intrare urmatoarele argumente in aceasta ordine: d , s (cu aceeasi interpretare ca la Cerinta 1), valoarea de deplasare h , aproximatia initiala y_0 a vectorului propriu, numarul maxim de iteratii $maxIter$ si precizia tol . Functia *PutereInv* are ca rezultat o valoarea proprie si un vector propriu. Rezultatele vor fi generate cind fie se va atinge precizia tol sau numarul maxim de iteratii $maxIter$.

Observatii: In metoda puterii inverse pe care o aveti de implementat, trebuie sa lucrati cu cei doi vectori d si s pe care functia *PutereInv* ii primeste ca intrare si nu trebuie sa compuneti matricea A din vectorii d si s . Pentru a calcula solutia unui sistem de ecuatii liniare, unde matricea sistemului este tridiagonala si simetrica, se va folosi algoritmul Thomas din Laboratorul 4. Pentru conditia de oprire in metoda puterii inverse folositi relatia: $norm(A * vecp - valp * vecp) < tol$, pe care trebuie sa o rescrieti in functie de datele de intrare d , s .

```
1 function [valp vecp] = PutereInv(d, s, h, y, maxIter, tol)
2     % Intrari:
3     % -> d: diagonala principala a matricei tridiagonale simetrice;
4     % -> s: supradiagonala matricei tridiagonale simetrice;
5     % -> y: aproximatia initiala a vectorului propriu;
6     % -> h: deplasare;
7     % -> maxIter: numarul maxim de iteratii;
8     % -> tol: precizia determinarii.
9     % Iesiri:
10    % -> [valp vecp]: valoarea proprie valp, respectiv vectorul propriu vecp.
```

Listing 6: PutereInv.m

Detalii de implementare si redactare

Pentru implementarea temei, functiile mentionate in enuntul temei trebuie implementate conform antetelor prezentate mai sus, fiecare functie in fisierul sau (vor exista deci minim sase fisiere *.m*). Nu este permisa

folosirea in implementarea cerintelor 1-6 a unor functii predefinite din Octave pentru calculul valorilor proprii/vectorilor proprii (cum este functia *eig()*). De asemenea, trebuie sa tineti cont de urmatoarele aspecte:

- Nu aveti voie sa compuneti matricea A in rezolvarea celor 6 cerinte.
- Codul sursa va contine comentarii cu privire la implementarea cerintelor.
- Existenta unui fisier README care va prezenta detaliile legate de implementarea temei.
- Fisierele sursa **.m* folosite in rezolvarea temei impreuna cu fisierul README vor fi incluse intr-o arhiva *.zip*. Denumirea arhivei respecta specificatiile din regulamentul cursului.
- Tema se va implementa in Octave.
- Se acorda 90 de puncte pentru o tema care functioneaza conform cerintelor descrise mai sus. 10 puncte se acorda pentru fisierul README scris corespunzator.

Resurse Web

1. http://en.wikipedia.org/wiki/Sturm's_theorem
2. http://www.encyclopediaofmath.org/index.php/Gershgorin_theorem