

## Laborator 4

# Triangularizare ortogonală cu reflectori. Problema CMMP. Clasificare.

### 4.1 Triangularizare ortogonală cu reflectori

#### 4.1.1 Reflectori elementari

**Definiția 4.1** Un *reflector elementar* (sau Householder) de ordin  $m$  și indice  $k$  este o matrice  $U_k \in \mathbb{R}^{m \times m}$ , de forma

$$U_k = I_m - \frac{1}{\beta_k} u_k u_k^T, \quad (4.1)$$

unde  $u_k \in \mathbb{R}^m$  este vectorul

$$u_k = [\underbrace{0 \dots 0}_{k-1} \ u_{kk} \ u_{k+1,k} \ \dots \ u_{mk}]^T$$

și

$$\beta_k = \|u_k\|^2/2.$$

Principalele proprietăți ale transformărilor Householder sunt date în următoarea teoremă.

**Teorema 4.2** a) Un reflector elementar  $U_k \in \mathbb{R}^{m \times m}$  este o matrice simetrică și ortogonală.

b) Dându-se un reflector elementar  $U_k$  și un vector  $x \in \mathbb{R}^m$ , produsul lor  $y = U_k x$  se poate calcula prin

$$y_i = \begin{cases} x_i, & i = 1 : k - 1 \\ x_i - \tau u_{ik}, & i = k : m \end{cases} \quad (4.2)$$

unde

$$\tau = \frac{1}{\beta_k} \sum_{i=k}^m u_{ik} x_i.$$

c) Dându-se un indice  $k \in 1 : m - 1$  și un vector  $x \in \mathbb{R}^m$  astfel încât

$$\sigma^2 = \sum_{i=k}^m x_i^2 \neq 0, \quad (4.3)$$

reflectorul  $U_k$  cu ajutorul căruia se pot introduce zerouri în pozițiile  $k + 1 : m$  ale vectorului  $x$ , mai precis care satisface relația

$$(U_k x)_i = \begin{cases} x_i, & i = 1 : k - 1 \\ -\sigma, & i = k \\ 0, & i = k + 1 : m \end{cases}$$

este definit de vectorul  $u_k \in \mathbb{R}^m$  dat de

$$u_{ik} = \begin{cases} 0, & i = 1 : k - 1 \\ x_k + \sigma, & i = k \\ x_i, & i = k + 1 : m \end{cases} \quad (4.4)$$

și scalarul

$$\beta_k = \sigma u_{kk}.$$

#### 4.1.2 Triangularizarea ortogonală

Rezolvarea problemei CMMP într-o manieră stabilă din punct de vedere numeric necesită o procedură pentru reducerea matricelor la formă triunghiulară prin transformări ortogonale.

**Teorema 4.3** Fie  $A \in \mathbb{R}^{m \times n}$ , cu  $m \geq n$ ; atunci există reflectorii elementari  $U_1, U_2, \dots, U_p$ , unde  $p = \min(m - 1, n)$ , astfel încât

$$U_p \dots U_2 U_1 A = R, \quad (4.5)$$

unde  $R$  este superior triunghiulară.

Pentru triangularizarea ortogonală a matricei  $A \in \mathbb{R}^{m \times n}$  putem folosi următoarea schemă (care calculează  $R$  în  $A$ ):

1. **pentru**  $k = 1 : \min(m - 1, n)$ 
  1. Se determină  $U_k$  astfel încât  $(U_k a_k)_i = 0$ , pentru  $i = k + 1 : m$
  2.  $A \leftarrow U_k A$

Schema de calcul de mai sus conduce la următorul algoritm.

ALGORITM 4.4 (*TORT* – *triangularizare ortogonală cu reflectori*)  
 (Dată o matrice  $A \in \mathbb{R}^{m \times n}$ , algoritmul suprascrie matricea  $A$  cu matricea superior triunghiulară  $R$  și calculează vectorii  $u_k = U(:, k)$  și scalarii  $\beta_k$  care definesc reflectorii elementari  $U_k$ ,  $k \in 1 : p$ ,  $p = \min(m - 1, n)$ , astfel încât are loc (4.5).)

1.  $p = \min(m - 1, n)$
2. **pentru**  $k = 1 : p$ 
  1.  $\sigma \leftarrow \text{sign}(a_{kk}) \cdot \sqrt{\sum_{i=k}^m a_{ik}^2}$
  2. **dacă**  $\sigma = 0$  **atunci**  $\beta_k \leftarrow 0$   
**altfel**
    1.  $U(k, k) \leftarrow u_{kk} = a_{kk} + \sigma$
    2. **pentru**  $i = k + 1 : m$ 
      1.  $U(i, k) \leftarrow u_{ik} = a_{ik}$
    3.  $\beta_k \leftarrow \sigma u_{kk}$
    4.  $a_{kk} \leftarrow r_{kk} = -\sigma$
    5. (**pentru**  $i = k + 1 : m$ 
      1.  $a_{ik} \leftarrow 0$ )
    6. **pentru**  $j = k + 1 : n$ 
      1.  $\tau \leftarrow (\sum_{i=k}^m u_{ik} a_{ij}) / \beta_k$
      2. **pentru**  $i = k : m$ 
        1.  $a_{ij} \leftarrow a_{ij} - \tau u_{ik}$

Algoritmul poate fi apelat cu sintaxa  $[R, U, \beta] = TORT(A)$ .

## 4.2 Problema celor mai mici pătrate

Fie  $A \in \mathbb{R}^{m \times n}$ ,  $b \in \mathbb{R}^m$  date. Considerăm sistemul liniar *supradeterminat* ( $m > n$ )

$$Ax = b.$$

În general, un astfel de sistem, având mai multe ecuații decât necunoscute, nu are soluții. O reformulare naturală a rezolvării sistemului este problema găsirii unui vector  $x \in \mathbb{R}^n$  astfel încât vectorul  $y = Ax$  să fie cât mai aproape posibil de vectorul  $b$ . Când apropierea este definită de norma euclidiană, se obține *problema celor mai mici pătrate* (CMMP):

$$\min_{x \in \mathbb{R}^n} \|r\| = \min_{x \in \mathbb{R}^n} \|b - Ax\|. \quad (4.6)$$

Vectorul  $r = b - Ax$  este numit reziduu, iar norma lui este minimizată.

### 4.2.1 Rezolvarea problemei CMMP

Condițiile în care problema CMMP are o soluție și când această soluție este unică sunt date de următoarea teoremă.

**Teorema 4.5** *Fie date  $A \in \mathbb{R}^{m \times n}$  și  $b \in \mathbb{R}^m$ , cu  $m > n$ . Problema CMMP (4.6) admite întotdeauna o soluție. Soluția este unică dacă și numai dacă matricea  $A$  are coloanele liniar independente, adică are rang maxim. În acest caz soluția CMMP a sistemului supradeterminat  $Ax = b$  poate fi scrisă în forma*

$$x^* = (A^T A)^{-1} A^T b. \quad (4.7)$$

Relația (4.7) este utilizabilă direct doar pentru matrice de dimensiuni mici. Un mijloc de calcul numeric stabil a soluției CMMP constă în triangularizarea ortogonală a matricei sistemului (algoritmul *TORT* din secțiunea precedentă). Notăm  $U = U_p \dots U_2 U_1$  în (4.5) și observăm că putem scrie

$$R = \begin{bmatrix} R' \\ 0 \end{bmatrix},$$

cu  $R'$  o matrice pătrată  $n \times n$  superior triunghiulară. Matricea  $R$  are coloanele liniar independente întrucât  $A$  are această proprietate și, deci,  $R'$  este o matrice nesingulară. Transformările ortogonale conservând norma euclidiană, avem

$$\|b - Ax\| = \|U(b - Ax)\| = \|d - Rx\|, \quad (4.8)$$

unde  $d = Ub$ . Folosind partiția

$$d = \begin{bmatrix} d' \\ d'' \end{bmatrix}, \quad (4.9)$$

unde  $d' \in \mathbb{R}^n$  și  $d'' \in \mathbb{R}^{m-n}$ , relația (4.8) poate fi scrisă

$$\|b - Ax\| = \left\| \begin{bmatrix} d' - R'x \\ d'' \end{bmatrix} \right\| = \sqrt{\|d' - R'x\|^2 + \|d''\|^2},$$

care este minimă când  $d' - R'x = 0$ . Deci, soluția CMMP a sistemului supradeterminat  $Ax = b$  poate fi calculată rezolvând sistemul superior triunghiular nesingular

$$R'x = d'.$$

### 4.2.2 Algoritm

Operațiile de mai sus sunt implementate de următorul algoritm.

**ALGORITM 4.6** (*CMMP – problema celor mai mici pătrate*) (Date o matrice  $A \in \mathbb{R}^{m \times n}$ ,  $m > n$ , cu coloanele liniar independente, și un vector  $b \in \mathbb{R}^m$ , acest algoritm calculează soluția CMMP a sistemului liniar supradeterminat  $Ax = b$ .)

- { Triangularizarea ortogonală a lui  $A$  }
  1.  $[R, U, \beta] = TORT(A)$
- { Calculul vectorului  $d$  care suprascrie  $b$  }
  2. **pentru**  $k = 1 : n$ 
    1.  $\tau \leftarrow (\sum_{i=k}^m u_{ik} b_i) / \beta_k$
    2. **pentru**  $i = k : m$ 
      1.  $b_i \leftarrow b_i - \tau u_{ik}$
- { Calculul soluției CMMP }
  3.  $x = UTRIS(R(1 : n, :), b(1 : n))$

### 4.3 Clasificare binară cu CMMP

#### 4.3.1 Problema, soluția CMMP și interpretare

Se presupune că avem exemple de semnale (vectori) din două clase și dorim ca, pe baza lor, să creăm o regulă prin care să clasificăm semnale viitoare. Notăm  $N_1$ ,  $N_2$  numărul de vectori (de antrenare) aparținând claselor  $\mathcal{C}_1$ , respectiv  $\mathcal{C}_2$ ; vectorii au dimensiune  $\ell$ . Un clasificator foarte simplu este cel bazat pe CMMP. (Atragem atenția că există metode de clasificare mai eficiente, dar și mai sofisticate.) Acesta construiește funcția liniară

$$f(v) = c^T v + d \quad (4.10)$$

unde  $v \in \mathbb{R}^\ell$  este un semnal de antrenare, iar  $c \in \mathbb{R}^\ell$  și  $d \in \mathbb{R}$  sunt parametrii încă necunoscuți ai funcției.

Pentru a afla acești parametri, fixăm ca obiectiv satisfacerea relației

$$f(v) = \begin{cases} 1, & v \in \mathcal{C}_1 \\ -1, & v \in \mathcal{C}_2 \end{cases} \quad (4.11)$$

Desigur, această relație nu poate fi satisfăcută exact, de aceea căutăm o soluție în sens CMMP. Notând  $\mathbf{1}_n$  un vector având toate elementele egale cu 1 și dimensiune  $n$ , relația (4.11) conduce la sistemul CMMP

$$\begin{bmatrix} V_1^T & \mathbf{1}_{N_1} \\ V_2^T & \mathbf{1}_{N_2} \end{bmatrix} \begin{bmatrix} c \\ d \end{bmatrix} = \begin{bmatrix} \mathbf{1}_{N_1} \\ -\mathbf{1}_{N_2} \end{bmatrix}, \quad (4.12)$$

în care  $V_1 \in \mathbb{R}^{\ell \times N_1}$  și  $V_2 \in \mathbb{R}^{\ell \times N_2}$  sunt matrice ale căror coloane sunt vectorii de antrenare din cele două clase. Rezolvarea sistemului (4.12), a cărui matrice are dimensiunea  $(N_1 + N_2) \times (\ell + 1)$ , produce valorile optime ale parametrilor  $c$  și  $d$  pentru această metodă.

Clasificarea efectivă a unui semnal nou se face ținând cont de care dintre valorile 1 sau  $-1$  este mai aproape valoarea funcției (4.10). Aceasta decizie se poate lua pe

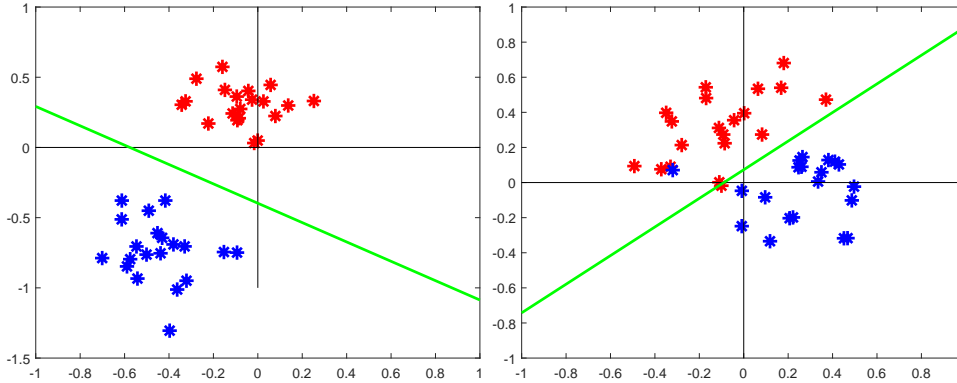


Figure 4.1: Exemple de clasificator CMMP.

baza semnului funcției (4.10). Mai precis, dacă  $v$  este acum un semnal oarecare, decizia de clasificare este

$$v \in \begin{cases} \mathcal{C}_1, & \text{sign}(c^T v + d) = 1, \\ \mathcal{C}_2, & \text{altfel} \end{cases} \quad (4.13)$$

Semnificația geometrică este simplă. Hiperplanul  $c^T v + d = 0$  separă spațiul  $\mathbb{R}^\ell$  în două, asociind fiecare jumătate unei clase. Exemple pentru  $\ell = 2$ , când hiperplanul separator devine o dreaptă, sunt prezentate în figura 4.1. Vectorii de antrenare din cele două clase sunt reprezentați cu roșu și albastru, iar dreapta separatoare cu verde (cu negru sunt axele de coordonate). În exemplul din stânga, cele două clase sunt bine separate de dreaptă; aceasta este situația ideală, mai rar întâlnită în practică. În exemplul din dreapta nu există o dreaptă separatoare; cea furnizată de metoda CMMP este însă un compromis acceptabil pentru a generaliza din exemplele clasele “roșie” și “albastră”.

Rezultatele clasificării semnalelor de antrenare este cuantificată prin matricea de confuzie, al cărui element  $(i, j)$  reprezintă numărul de semnale din clasa  $i$  care sunt clasate în clasa  $j$ ; deci pe diagonală se află numărul de semnale clasificate corect, iar pe celelalte poziții numărul de semnale clasificate incorect. Pentru figura 4.1 din dreapta (clasa roșie fiind  $\mathcal{C}_1$  și fiecare clasă având 20 de semnale), matricea de confuzie este

$$\begin{bmatrix} 20 & 0 \\ 1 & 19 \end{bmatrix}.$$

Toate semnalele roșii sunt clasificate corect, dar unul albastru este clasificat incorect.

### 4.3.2 Date despre flori de iris

Un exemplu simplu și celebru de date pentru clasificare conține măsurători ale petalelor și separelor pentru trei tipuri de flori de iris: Iris Setosa, Iris Versicolor, Iris Virginica. (Dacă nu mai știți ce sunt separele, figura 4.2 vă reamintește.)

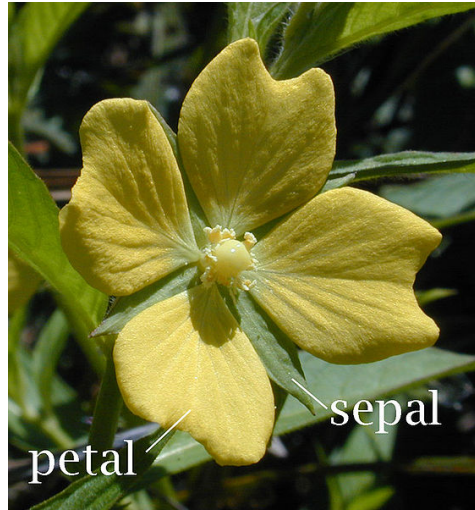


Figure 4.2: Petale și sepale. (In engleză, în figură.)

Fișierul `iris.mat` conține câte 50 de măsurători pentru fiecare tip de iris, grupate într-o matrice cu 150 de linii și 5 coloane. Primele patru coloane sunt lungimea și lățimea sepalei, respectiv petalei. A cincea coloană conține o etichetă numerică (1 pentru Setosa, 2 pentru Versicolor, 3 pentru Virginica).

Deoarece ne ocupăm doar de clasificare binară, putem totuși discerne între cele trei tipuri de iris proiectând trei clasificatoare, antrenate cu prima clasă formată dintr-un tip de iris și a doua clasă formată din celelalte două tipuri.

## 4.4 Sarcini de lucru

### 4.4.1 A. In laborator

1. Elaborați și editați programul MATLAB pentru implementarea algoritmului *TORT* de triangularizare ortogonală. Testați funcționarea programului pentru matrice cu elemente aleatoare de dimensiuni cel puțin  $7 \times 5$ .
2. Elaborați și editați programul MATLAB pentru implementarea algoritmului CMMP de rezolvare a sistemelor supradeterminate și verificați corectitudinea pentru sisteme cu 20 de ecuații și 6 necunoscute.
3. Studiați și executați programul `script_clasificare`, care implementează metoda CMMP de clasificare binară pe date 2D. Modificați diverși parametri pentru a vedea rezultate diferite. (Figura 4.1 este generată cu acest program.)
4. Adaptați programul `script_clasificare` pentru datele din fișierul `iris.mat`. (Eliminați reprezentarea grafică, posibilă doar pentru proiecții pe două dimensiuni.) Calculați matricele de confuzie pentru toate cele trei clasificări binare

*unul vs restul* posibile. Veți observa că Iris Setosa poate fi perfect separată de celelalte două; în schimb niciuna dintre celelalte, nu poate fi separată de restul. Pe baza acestor informații propuneți o metodă de a decide cărui tip îi aparține un vector de măsurători nou.

Observație. Dacă obiectivele de mai sus nu se ating în laborator atunci se continuă acasă.

#### 4.4.2 B. Acasă

- 1 Scrieți și testați programe MATLAB pentru <sup>1</sup>:
  - a) calculul elementelor definatorii ale unui reflector  $U_k$  care anulează componentele  $k + 1 : n$  ale unui vector  $x \in \mathbb{R}^n$  dat;
  - b) înmulțirea unei matrice  $A$  pe stânga cu un reflector  $U_k$ :  $A \leftarrow U_k A$ ;
  - c) înmulțirea unei matrice  $A$  pe dreapta cu un reflector  $U_k$ :  $A \leftarrow A U_k$ .
- 2 Rescrieți algoritmul de triangularizare ortogonală a unei matrice cu folosirea procedurilor de mai sus.
- 3 Fie  $H \in \mathbb{R}^{m \times n}$  ( $m > n$ ) o matrice superior Hessenberg ( $h_{ij} = 0$  for  $i > j+1$ ). Scrieți algoritmi eficienți și programe pentru triangularizarea ortogonală a matricei  $H$ .
- 4 Considerăm dat vectorul  $x \in \mathbb{R}^n$ , de normă unitară. Scrieți un program de calcul al unei matrice cu coloanele ortogonale  $Y \in \mathbb{R}^{n \times (n-1)}$  astfel încât  $x^T Y = 0$  (i.e. matricea  $[x \ Y]$  să fie ortogonală).
- 5 Fie  $A \in \mathbb{R}^{m \times n}$ ,  $m > n$ , o matrice cu rang maxim. Elaborați un algoritim și un program de calcul al pseudo-inversei  $A^+ = (A^T A)^{-1} A^T$  a lui  $A$ .
- 6 Găsiți alte date pentru clasificare și încercați metoda CMMP.

---

<sup>1</sup>Aceste proceduri vor fi utile la laboratorul dedicat calculului valorilor proprii.