

## Curs 4.

### Metode de rezolvare a sistemelor liniare bazate pe factorizare ortogonală.

Sistemul supradeterminat de ecuații liniare

$$\mathbf{Ax}=\mathbf{b}, \quad \mathbf{A} \in \mathbb{R}^{m \times n}, \quad \mathbf{b} \in \mathbb{R}^m, \quad m > n$$

nu admite în general soluție.

*Soluția în sensul celor mai mici pătrate (sau pseudosoluția)* se definește ca vectorul  $\mathbf{x}^*$  din  $\mathbb{R}^n$  care asigură minimizarea normei euclidiene a vectorului reziduu:

$$r(\mathbf{x}^*) = \|\mathbf{b} - \mathbf{A} \cdot \mathbf{x}^*\|_2 = \min_{\mathbf{x} \in \mathbb{R}^n} r(\mathbf{x}) = \min_{\mathbf{x} \in \mathbb{R}^n} \|\mathbf{b} - \mathbf{A} \cdot \mathbf{x}\|_2$$

*Pseudosoluția este soluție exactă pentru sistemul normal*

$$\mathbf{A}^T \mathbf{Ax} = \mathbf{A}^T \mathbf{b}$$

Sistemul normal este rău condiționat astfel încât metodele obișnuite de rezolvare (Gauss, Cholesky, etc) nu dau rezultate satisfăcătoare, fiind necesare metode mai stabile din punct de vedere numeric, bazate pe triunghiularizare ortogonală.

Pseudosoluția este unică dacă matricea  $\mathbf{A}$  are coloanele liniar independente, caz în care se exprimă ca:

$$\mathbf{x}^* = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{b} = \mathbf{A}^\# \mathbf{b}$$

în care se definește

$$\mathbf{A}^\# = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \in \mathbb{R}^{n \times m}$$

ca *inversă generalizată* sau *pseudoinversă Penrose - Moore* a matricei dreptunghiulare  $\mathbf{A} \in \mathbb{R}^{m \times n}$ .

Doi vectori  $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$  sunt ortogonali în raport cu produsul euclidian dacă  $\mathbf{x}^T \mathbf{y} = 0$ .

Complementul ortogonal  $\mathbf{S}^\perp$  a subspațiului  $\mathbf{S} \in \mathbb{R}^n$  se definește ca:

$$\mathbf{S}^\perp = \{\mathbf{y} \in \mathbb{R}^n \mid \mathbf{x}^T \mathbf{y} = 0, \mathbf{x} \in \mathbf{S}\}$$

Vectorii  $\mathbf{q}_1, \mathbf{q}_2, \dots, \mathbf{q}_n \in \mathbb{R}^m$  sunt ortonormați, dacă  $\mathbf{q}_i^T \mathbf{q}_j = \delta_{ij}$

Matricea  $\mathbf{Q} = [\mathbf{q}_1 \ \mathbf{q}_2 \ \dots \ \mathbf{q}_n] \in \mathbb{R}^{m \times n}$  este *ortogonală* și  $\mathbf{Q}^T \cdot \mathbf{Q} = \mathbf{I}_n$

Considerăm o matrice ortogonală pătrată  $\mathbf{Q} \in \mathbb{R}^{n \times n}$ . Matricele ortogonale au următoarele proprietăți:

$$1^0. \quad \mathbf{Q}^{-1} = \mathbf{Q}^T$$

$$2^0. \quad \det(\mathbf{Q}) = \pm 1$$

$$\det(\mathbf{Q}^T \cdot \mathbf{Q}) = \det(\mathbf{Q} \cdot \mathbf{Q}) = \det^2(\mathbf{Q}) = \det(\mathbf{I}_n) = 1$$

În  $\mathbb{C}^n$  vectorii sunt ortogonali dacă  $\mathbf{x}^H \mathbf{y} = 0$ , unde  $\mathbf{x}^H = \underline{\mathbf{x}}^T$  (transpus conjugat) Matricea  $\mathbf{Q}$  este *unitară* dacă  $\mathbf{Q}^H \cdot \mathbf{Q} = \mathbf{I}_n$ .

3<sup>0</sup>.  $\|\mathbf{Hx}\|_2 = \|\mathbf{x}\|_2$  -transformarea ortogonală conservă norma euclidiană a unui vector

$$\|\mathbf{Hx}\|_2^2 = (\mathbf{Hx}, \mathbf{Hx}) = (\mathbf{Hx})^T (\mathbf{Hx}) = \mathbf{x}^T \mathbf{H}^T \mathbf{Hx} = \mathbf{x}^T (\mathbf{H}^T \mathbf{H}) \mathbf{x} = \mathbf{x}^T \mathbf{x} = \|\mathbf{x}\|_2^2$$

4<sup>0</sup>.  $\|H\|_2=1$  -norma matricială euclidiană subordonată este 1:

$$\|H\|_2 = \sup_{x \neq 0} \frac{\|H \cdot x\|_2}{\|x\|_2} = 1$$

5<sup>0</sup>.  $\text{cond}_2(A) = 1$

6<sup>0</sup>.  $\|H \cdot A\|_2 = \|A\|_2$  -transformarea ortogonală conservă norma euclidiană a unei matrice:

$$R=H \cdot A \Rightarrow \|R\|_2 = \|H \cdot A\|_2 \leq \|H\|_2 \cdot \|A\|_2 = \|A\|_2$$

$$A=H^T \cdot R \Rightarrow \|A\|_2 = \|H^T \cdot R\|_2 \leq \|H\|_2 \cdot \|R\|_2 = \|R\|_2$$

$$\|H \cdot A\|_2 = \|A\|_2$$

O matrice  $P$  se numește *matrice idempotentă* sau *matrice proiector*, dacă  $P^2=P$ .

În raport cu un proiector  $P$ , un vector  $v \in \mathbb{R}^n$  poate fi descompus unic în:

$$v = P \cdot v + (I-P) \cdot v = v_1 + v_2$$

$$P \cdot v_2 = P \cdot (I-P) \cdot v = (P-P^2) \cdot v = 0 \cdot v = 0$$

$v_1 = P \cdot v \in S$  – proiecția lui  $v$  în *spațiul imagine* a lui  $P$

$v_2 = (I-P) \cdot v \in S^\perp$  – proiecția lui  $v$  în *spațiul nul* a lui  $P$

Dacă  $P$  este simetric ( $P=P^T$ ), atunci:

$$v_1^T v_2 = (P \cdot v)^T \cdot (I-P) \cdot v = v^T \cdot P \cdot (I-P) = v^T \cdot (P-P^2) v = 0 \Rightarrow v_2 \perp S \Rightarrow v_2 \in S^\perp$$

$P$  este proiectorul ortogonal în  $S$

$I-P$  este proiectorul ortogonal în  $S^\perp$

O *metodă ortogonală* transformă sistemul  $Ax=b$  într-un sistem echivalent cu matrice superior triunghiulară  $HAx=Hb$  în care matricea de transformare  $H \in \mathbb{R}^{m \times n}$  este ortogonală. Matricea sistemului echivalent (în sensul că are aceeași pseudosoluție cu sistemul inițial), are același număr de condiționare în normă 2. Într-adevăr, transformările ortogonale conservă norma euclidiană

### Metoda Householder

O *matrice ortogonală elementară* se obține modificând matricea unitate, cu o matrice de rang 1. Reflectorul Householder este o asemenea matrice ortogonală:

$$H = I - \beta u u^T, \quad \beta = 2 / (u^T u)$$

$H$  este simetrică:

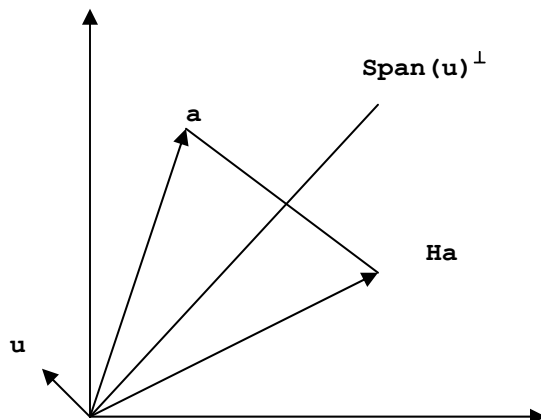
$$H^T = (I - \beta u u^T)^T = I - \beta (u u^T)^T = I - \beta (u^T)^T u^T = I - \beta u u^T = H$$

$H$  este ortogonală:

$$H^T H = H^2 = I - 2\beta u u^T + \beta^2 u (u^T u) u^T = I - 2\beta u u^T + \beta^2 u (2/\beta) u^T = I$$

$u$  se numește *vector Householder*.

$Hu = (I - \beta u u^T)u = u - \beta u (u^T u) = u - \beta u (2/\beta) = u - 2u = -u$  ceea ce justifică numele de reflector



$$H\mathbf{a} = (\mathbf{I} - \beta \mathbf{u} \mathbf{u}^T) \mathbf{a} = \mathbf{a} - \beta \mathbf{u} (\mathbf{u}^T \mathbf{a})$$

Dacă  $\mathbf{a} \perp \mathbf{u}$  atunci  $\mathbf{u}^T \mathbf{a} = 0$  și  $H\mathbf{a} = \mathbf{a}$

altfel  $\mathbf{a} - H\mathbf{a} = \beta \mathbf{u} (\mathbf{u}^T \mathbf{a})$  adică  $\mathbf{a} - H\mathbf{a} \parallel \mathbf{u}$

Transformarea  $H$  reflectă un vector  $\mathbf{a}$  în raport cu planul normal vectorului  $\mathbf{u}$ .

Vom determina transformarea  $H$ , astfel încât vectorul reflectat  $H\mathbf{a}$  să aibă toate componentele 0, cu excepția primeia dintre ele:

$$H\mathbf{a} = \pm \sigma \mathbf{e}_1, \quad \sigma = \|\mathbf{a}\|_2$$

$$HH\mathbf{a} = \pm \sigma H\mathbf{e}_1 \Rightarrow H\mathbf{e}_1 = \pm \mathbf{a} / \sigma$$

Vom determina deci matricea ortogonală  $H$ , având prima coloană proporțională cu  $\pm \mathbf{a} / \sigma$

Matricea Householder se folosește sub forma echivalentă:

$H = \mathbf{I} - 2\mathbf{u} \mathbf{u}^T$ , cu  $\mathbf{u}^T \mathbf{u} = 1$  (se observă că în acest caz particular  $\beta=2$ ).

Dacă  $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ ,  $\mathbf{x} \neq \mathbf{y}$ ,  $\|\mathbf{x}\|_2 = \|\mathbf{y}\|_2$ ,  $\mathbf{u} = \frac{\mathbf{x} - \mathbf{y}}{\|\mathbf{x} - \mathbf{y}\|_2}$  atunci  $(\mathbf{I} - 2\mathbf{u} \mathbf{u}^T) \mathbf{x} = \mathbf{y}$ .

$$\|\mathbf{x} - \mathbf{y}\|_2^2 = (\mathbf{x} - \mathbf{y})^T (\mathbf{x} - \mathbf{y}) = \mathbf{x}^T \mathbf{x} - \mathbf{x}^T \mathbf{y} - \mathbf{y}^T \mathbf{x} + \mathbf{y}^T \mathbf{y} = 2(\mathbf{x}^T \mathbf{x} - \mathbf{y}^T \mathbf{x})$$

$$(\mathbf{I} - 2\mathbf{u} \mathbf{u}^T) \mathbf{x} = \mathbf{x} - 2 \frac{(\mathbf{x} - \mathbf{y})(\mathbf{x} - \mathbf{y})^T}{\|\mathbf{x} - \mathbf{y}\|_2^2} \mathbf{x} = \mathbf{x} - \frac{2(\mathbf{x} - \mathbf{y})(\mathbf{x}^T \mathbf{x} - \mathbf{y}^T \mathbf{x})}{2(\mathbf{x}^T \mathbf{x} - \mathbf{y}^T \mathbf{y})}$$

$$(\mathbf{I} - 2\mathbf{u} \mathbf{u}^T) \mathbf{x} = \mathbf{x} - (\mathbf{x} - \mathbf{y}) = \mathbf{y}$$

Cu alte cuvinte, dacă doi vectori  $\mathbf{x}$  și  $\mathbf{y}$  au aceeași origine și aceeași lungime, atunci putem determina o transformare (rotație), care să aducă pe unul din ei peste celălalt.

Suntem interesați ca unul din vectori să fie paralel cu o axă de coordonate, adică dacă notăm lungimea  $\sigma = \pm \|\mathbf{x}\|_2$ , și alegem  $\mathbf{v} = \mathbf{x} + \sigma \mathbf{e}_1$ ,  $\mathbf{u} = \mathbf{v} / \|\mathbf{v}\|_2$  atunci  $(\mathbf{I} - 2\mathbf{u} \mathbf{u}^T) \mathbf{x} = -\sigma \mathbf{e}_1$   
 $\mathbf{u} \mathbf{u}^T = \mathbf{v} \mathbf{v}^T / \|\mathbf{v}\|_2^2$

$$\|\mathbf{v}\|_2^2 = (\mathbf{x} + \sigma \mathbf{e}_1)^T (\mathbf{x} + \sigma \mathbf{e}_1) = \mathbf{x}^T \mathbf{x} + \sigma \mathbf{x}^T \mathbf{e}_1 + \sigma \mathbf{e}_1^T \mathbf{x} + \sigma^2 \mathbf{e}_1^T \mathbf{e}_1 = \|\mathbf{x}\|_2^2 + \sigma^2 + 2\sigma \mathbf{x}_1 = 2\sigma^2 + 2\sigma \mathbf{x}_1 =$$

$$2\sigma(\sigma + \mathbf{x}_1)$$

Considerăm  $\mathbf{x} \in \mathbb{R}^m$  cu  $\sigma^2 = \sum_{i=p}^m \mathbf{x}_i^2 = \|\mathbf{x}(p+1:m)\|_2^2 \neq 0$

Vom aplica vectorului  $\mathbf{x}$  un reflector Householder  $\mathbf{H}_p$  pentru a-i anula componentele  $\mathbf{x}(p+1:m)$

Fixăm componentele vectorului  $\mathbf{u}_p$  care intervine în expresia reflectorului  $\mathbf{H}_p$  la valorile:

$$\mathbf{u}_{ip} = \begin{cases} 0 & i = 1 : p-1 \\ \mathbf{x}_p + \sigma & i = p \\ \mathbf{x}_i & i = p+1 : m \end{cases}$$

Acest vector poartă numele de *vector Householder*.

$$\mathbf{H}_p \cdot \mathbf{x} = (\mathbf{I}_m - \beta \cdot \mathbf{u}_p \cdot \mathbf{u}_p^T) \cdot \mathbf{x} = \mathbf{x} - \beta(\mathbf{u}_p^T \cdot \mathbf{x}) \cdot \mathbf{u}_p$$

$$\mathbf{u}_p^T \cdot \mathbf{x} = \sum_{i=p}^m \mathbf{u}_{ip} \mathbf{x}_i = \mathbf{u}_{pp} \mathbf{x}_p + \sum_{i=p+1}^m \mathbf{u}_{ip} \mathbf{x}_i = (\mathbf{x}_p + \sigma) \cdot \mathbf{x}_p + \sum_{i=p+1}^m \mathbf{x}_i^2$$

$$\mathbf{u}_p^T \cdot \mathbf{x} = (\mathbf{x}_p + \sigma) \cdot \mathbf{x}_p + \sum_{i=p}^m \mathbf{x}_i^2 - \mathbf{x}_p^2 = (\mathbf{x}_p + \sigma) \cdot \mathbf{x}_p + \sigma^2 - \mathbf{x}_p^2 = \sigma \cdot (\mathbf{x}_p + \sigma)$$

$$\mathbf{u}_p^T \cdot \mathbf{u}_p = \|\mathbf{u}_p\|_2^2 = \sum_{i=p}^m \mathbf{u}_{ip}^2 = \left( \mathbf{u}_{pp}^2 + \sum_{i=p+1}^m \mathbf{x}_i^2 \right) = (\mathbf{x}_p + \sigma)^2 + \sigma^2 - \mathbf{x}_p^2 = 2\sigma(\mathbf{x}_p + \sigma)$$

$$\beta = \frac{2}{\mathbf{u}_p^T \cdot \mathbf{u}_p} = \frac{1}{\sigma \cdot (\mathbf{x}_p + \sigma)}$$

Pentru a evita anularea numitorului lui  $\beta$  vom alege ca  $\sigma$ , să aibă același semn cu  $\mathbf{x}_p$

$$\sigma = \text{sign}(\mathbf{x}_p) \cdot \sqrt{\sum_{i=p}^m \mathbf{x}_i^2} = \text{sign}(\mathbf{x}_p) \cdot \|\mathbf{x}(p:m)\|_2$$

$$(\mathbf{H}_p \cdot \mathbf{x})_i = \mathbf{x}_i - \beta \cdot (\mathbf{u}_p^T \cdot \mathbf{x}) \cdot \mathbf{u}_{ip} = \mathbf{x}_i - \frac{\sigma \cdot (\mathbf{x}_p + \sigma)}{\sigma \cdot (\mathbf{x}_p + \sigma)} \mathbf{u}_{ip} = \mathbf{x}_i - \mathbf{u}_{ip}$$

$$(\mathbf{H}_p \cdot \mathbf{x})_i = \mathbf{x}_i - \mathbf{u}_{ip} = \begin{cases} \mathbf{x}_i & i = 1 : p-1 \\ \mathbf{x}_p - \mathbf{u}_{pp} = -\sigma & i = p \\ 0 & i = p+1 : m \end{cases}$$

În urma aplicării transformării  $\mathbf{H}_p$ , vectorul  $\mathbf{x}$  se modifică astfel:

- primele  $p-1$  componente rămân nemodificate
- componenta  $p$  devine *mare* în valoare absolută
- restul componentelor  $(p+1:m)$  se anulează.

```
function [u, b, x]=HSx(x, p)
% determină reflectorul Householder care
% lasa primele p-1 componente neschimbate
% modifică componenta p la o valoare mare
% anulează restul componentelor
```

```

x=x(:); %ne asiguram ca x e vector coloana
m=length(x);
sig=sign(x(p))*norm(x(p:m));
b=1/sig/(x(p)+sig);
u=[zeros(p-1,1); x(p)+sig; x(p+1:m)];
x(p)=-sig;
x(p+1:m)=0;
% H este determinat de v si b
% H=eye(m)-v*v'./b;

```

Particularizăm pentru  $p=1$

$$(H_1 \cdot x)_i = \begin{cases} -\sigma, & i = 1 \\ 0, & i = 2 : m \end{cases}$$

$$H_1 \cdot x = -\sigma \cdot e_1 = -\|x\|_2 \cdot \text{sign}(x_1) \cdot e_1$$

$$u_{i1} = \begin{cases} x_1 + \sigma \\ x_i, & i = 2 : m \end{cases}$$

$$u_1 = x + \sigma \cdot e_1$$

Transformarea  $H_1$  este determinată numai de vectorul  $u$  și de  $\beta$ . Așadar

$$\begin{bmatrix} x_1 \\ x_2 \\ \dots \\ x_m \end{bmatrix} \xrightarrow{H_1} \begin{bmatrix} -\sigma \\ 0 \\ \dots \\ 0 \end{bmatrix} \quad \text{în care} \quad H_1 = \left\{ \begin{bmatrix} u_1 \\ u_2 \\ \dots \\ u_m \end{bmatrix}, \beta \right\}$$

Cu excepția primei componente, vectorii  $u$  și  $x$  coincid, ceea ce ne sugerează să păstrăm pe  $u$  în  $x$ , în pozițiile  $2:m$ , unde apar zerouri.

Cum vectorul  $u$  este determinat esențial numai ca direcție, îl vom norma cu  $\sigma$ :  $\bar{u} = \frac{u}{\sigma}$

$$u_{i1} = \begin{cases} \frac{x_1}{\sigma} + 1 = \bar{x}_1 + 1 \\ \frac{x_i}{\sigma} = \bar{x}_i, & i = 2 : m \end{cases}$$

Prima componentă  $u_1$  va fi păstrată în  $\beta$ . Într-adevăr, înainte de scalarea cu  $\sigma$ ,  $\beta = \sigma \cdot u_1$

$$\begin{bmatrix} x_1 \\ x_2 \\ \dots \\ x_m \end{bmatrix} \xrightarrow{H_1} \begin{bmatrix} -\sigma \\ u_2 \\ \dots \\ u_m \end{bmatrix}, \quad \beta = u_1$$

```

function [beta, x]=HS1x(x)
% aplica transformarea H1 unui vector x
% intoarce beta=u(1) si u(2:m) in x
x=x(:);
m=length(x);

```

```

beta=0;
sig=norm(x);
if sig~=0
    if x(1)~=0
        sig = sig*sign(x(1));
    end;
    x(1:m)=x(1:m)/sig;
    beta= 1+x(1);
    x(1)=-sig;
else
    error('elemente deja nule');
end

```

Întrucât transformarea  $H_p$  modifică numai componentele din pozițiile  $p:m$ , ea se poate realiza aplicând  $H_1$  unui vector oarecare  $y$

Aplicăm reflectorul  $H_1$  unui vector oarecare  $y$

$$H_1 \cdot y = (I_m - \beta \cdot u_1 \cdot u_1^T) \cdot y = y - \beta \cdot (u_1^T \cdot y) \cdot u_1 = y - \rho \cdot u_1$$

$$\rho = \beta \cdot u_1^T \cdot y = \beta \cdot \sum_{i=1}^m u_{i1} \cdot y_i$$

$$(H_1 \cdot y)_i = y_i - \rho \cdot u_{i1}, \quad i = 1 : m$$

```

function y=HSl1y(x, beta, y)
% aplica reflectorul H1 unui vector oarecare y
    if beta ~= 0
        t = x(1);
        x(1) = beta;
        m=length(x);
        ro=v(1:m)'*y(1:m)*beta;
        y(1:m)=y(1:m)-ro.*x(1:m);
    end

```

În locul rezolvării sistemului  $A \cdot x = b$ , cu  $A \in \mathbb{R}^{m \times n}$ ,  $b \in \mathbb{R}^m$  se rezolvă sistemul echivalent  $H \cdot A \cdot x = H \cdot b$ , cu  $H \in \mathbb{R}^{m \times m}$ ,  $H$ -ortogonală, sistem care are aceeași condiționare cu sistemul inițial ( $\text{cond}(H \cdot A) = \text{cond}(A)$ )

***Pentru orice matrice  $A \in \mathbb{R}^{m \times n}$  există o matrice ortogonală  $H \in \mathbb{R}^{m \times m}$  astfel încât  $H \cdot A = R$  în care  $R \in \mathbb{R}^{m \times n}$  este o matrice superior triunghiulară***

Matricea ortogonală se formează dintr-un produs

$$H = H_n H_{n-1} \dots H_2 H_1$$

$$H \cdot A = H_n H_{n-1} \dots H_2 H_1 A$$

$$A_{p+1} = H_p A_p \quad \text{cu } p=1:n, \text{ pornind cu } A_1 = A.$$

De exemplu reflectorul  $H_1$  care anulează prima coloană a matricei  $A \in \mathbb{R}^{m \times n}$  este:

```

[beta, A(:,1)]=HSH1x(A(:,1))
for j=2:n
    A(:,j)=HSl1y(A(:,1), beta, A(:,j));
end

```

Matricea are structura 
$$H_p = \begin{bmatrix} I_{p-1} & 0 \\ 0 & v_{n-p+1} \end{bmatrix}$$

Spre deosebire de metodele gaussiene, metodele ortogonale asigură elemente diagonale mari în valoare absolută, ceea ce conferă o stabilitate deosebită a metodei.

Vom determina matricea  $H_p$ , care transformă matricea  $A_p$ , utilizând algoritmul **HSx()**, punând ca vector  $x$  coloana  $p$  din  $H_p A_p$

$$H_p A_p = H_p [a_1 \ a_2 \ \dots \ a_n] = [H_p a_1 \ H_p a_2 \ \dots \ H_p a_n]$$

Coloanele situate în dreapta coloanei  $p$ , ( $j > p$ ) se modifică folosind algoritmul **HSy()** Transformarea  $H_p$  nu modifică coloanele  $a_j$  pentru care  $\sigma = 0$  (coloanele din stânga coloanei  $p$ , cu  $j < p$ ).

Acestea au  $a_{ij} = 0$  pentru  $i < p < j$ .

```
function A=trorto(A)
% triunghiularizare ortogonala Householder
[m,n]=size(A);
H=zeros(m,m);
for p=1:min(m-1,n)
    % se determina Hp
    sig=sign(A(p,p))*norm(A(p:m,p));
    H(p,p)=A(p,p)+sig;
    H(p+1:m,p)=A(p+1:m,p);
    beta(p)=sig*H(p,p);
    % se aplica Hp coloanelor din A
    for j=p+1:n
        ro = H(p,p:m)*A(p:m,j)/beta(p);
        A(p:m,j)=A(p:m,j)-ro*H(p:m,p);
    end;
    A(p,p)=-sig;
    A(p+1:m,p)=0;
end
```

Matricea  $H$  poate fi păstrată în triunghiul inferior din  $A$  (care se anulează), iar elementele diagonale în vectorul **beta**. În triunghiul superior din  $A$  vom avea matricea  $R$

În rezumat, transformarea aplicată matricei :

- îi lasă neschimbate primele coloane
- modifică elementele din coloana astfel:
  - primele elemente rămân neschimbat
  - elementul diagonal devine mare
  - elementele subdiagonale se anulează
  - modifică elementele din coloanele  $j > p$  astfel:
    - primele  $p-1$  elemente rămân neschimbate
    - elementele din pozițiile  $p:m$  se calculează cu relația Householder.

Pornind de la sistemul inițial, prin aplicarea transformărilor:

$$\begin{aligned} A_{p+1} &= H_p A_p, & A_1 &= A, & p &= 1 : n \\ b_{p+1} &= H_p b_p, & b_1 &= b \end{aligned}$$

se ajunge la sistemul echivalent superior triunghiular

$$\mathbf{A}_{n+1}\mathbf{x} = \mathbf{b}_{n+1}$$

### Metoda Givens (matrice de rotație elementare).

Metoda Givens utilizează matrice elementare de rotație, de forma:

$$\mathbf{G}_{kl} = \begin{bmatrix} \mathbf{I}_{k-1} & & & \\ & \mathbf{c} & & \mathbf{s} \\ & & \mathbf{I}_{l-k-1} & \\ & -\mathbf{s} & & \mathbf{c} \\ & & & & \mathbf{I}_{m-l} \end{bmatrix}, \quad \mathbf{c}^2 + \mathbf{s}^2 = 1,$$

O asemenea matrice definește o rotație de ordinul  $m$  în planul  $(k, l)$  cu unghiul  $\theta$ , cu  $\mathbf{c} = \cos \theta$  și  $\mathbf{s} = \sin \theta$

Matricea  $\mathbf{G}_{kl}$  este ortogonală:  $\mathbf{G}_{kl} \cdot \mathbf{G}_{kl}^T = \mathbf{I}_n$

Transformarea  $\mathbf{G}_{kl}$  aplicată unui vector  $\mathbf{x} \in \mathbb{R}^m$  îi modifică numai componentele  $\mathbf{x}_k$  și  $\mathbf{x}_l$

$$\mathbf{G}_{kl}\mathbf{x} = [\mathbf{x}_1 \dots \mathbf{c}\mathbf{x}_k + \mathbf{s}\mathbf{x}_l \dots -\mathbf{s}\mathbf{x}_k + \mathbf{c}\mathbf{x}_l \dots \mathbf{x}_m]'$$

```
function x=rotvec(k, l, c, s, x)
%aplica transformarea Gkl vectorului x
t = zeros(2,1);
t = [c s;-s c]*[x(k); x(l)];
x(k) = t(1);
x(l) = t(2);
```

Determinăm transformarea  $(\mathbf{c}, \mathbf{s})$ , care anulează componenta  $\mathbf{x}_l$ :

$$-\mathbf{s} \cdot \mathbf{x}_k + \mathbf{c} \cdot \mathbf{x}_l = 0$$

$$\mathbf{c} = \frac{\mathbf{x}_k}{\sqrt{\mathbf{x}_k^2 + \mathbf{x}_l^2}} = \frac{\mathbf{x}_k}{r} \quad \mathbf{s} = \frac{\mathbf{x}_l}{\sqrt{\mathbf{x}_k^2 + \mathbf{x}_l^2}} = \frac{\mathbf{x}_l}{r}$$

$$\mathbf{x}_k = \mathbf{c} \cdot \mathbf{x}_k + \mathbf{s} \cdot \mathbf{x}_l = \frac{\mathbf{x}_k^2 + \mathbf{x}_l^2}{\sqrt{\mathbf{x}_k^2 + \mathbf{x}_l^2}} = \sqrt{\mathbf{x}_k^2 + \mathbf{x}_l^2} = r$$

Rotația  $\mathbf{G}_{kl}$ , care modifică componentele la valorile:  $\mathbf{x}_k=r$  și  $\mathbf{x}_l=0$  este:

```
function [c, s, x]=rot(k, l, x)
% determina rotatia (c,s) care
% anuleaza x(l)
r=sqrt(x(k)^2+x(l)^2);
c=x(k)/r;
s=x(l)/r;
```

În metoda Givens matricea ortogonală  $\mathbf{G}$  se formează ca un produs de *matrice elementare de "rotație"* de forma

$$\mathbf{G} = \underbrace{\mathbf{G}_{n-1,n}}_{\mathbf{G}_{n-1}} \cdot \underbrace{\mathbf{G}_{n-2,n} \cdot \mathbf{G}_{n-2,n-1}}_{\mathbf{G}_{n-2}} \cdot \underbrace{\mathbf{G}_{1n} \cdot \mathbf{G}_{1,n-1} \dots \mathbf{G}_{12}}_{\mathbf{G}_1}$$

Matricea  $G_{k1}$  afectează prin înmulțire numai liniile  $k$  și  $1$ :

$$A^{(p+1)} = G_{k1} A^{(p)}, \quad k=1:n-1, \quad 1+k+1:n, \quad A^{(0)} = A$$

Plecând cu matricea  $A$  și folosind transformările ortogonale  $G_{k1}$  se obține o matrice superior triunghiulară:

```
function A=trrot(A)
% triunghiularizare ortogonala cu rotatii
[m,n]=size(A);
t=zeros(2,n);
for k=1:min(m-1,n)
    for l=k+1:m
        [c,s]=rot(k,l,A(:,k));
        t=[c s;-s c]*[A(k,1:n);A(l,1:n)];
        A(k,1:n)=t(1,1:n);
        A(l,1:n)=t(2,1:n);
    end
end
end
```

### Factorizarea QR

Orice matrice  $A \in \mathbb{R}^{m \times n}$  cu coloane independente, poate fi descompusă sub forma  $A=QR$ , cunoscută sub numele de factorizare QR în care  $Q$  este o matrice cu coloane ortogonale și  $R$  este o matrice pătrată superior triunghiulară.

Pornim de la teorema Householder:  $\forall A \in \mathbb{R}^{m \times n}$ ,  $A$ -matrice cu coloane liniar independente,  $\exists H \in \mathbb{R}^{m \times m}$ ,  $H$  – ortogonală ( $H^T H = I_m$ ) astfel încât  $H \cdot A = R$ , cu  $R \in \mathbb{R}^{m \times n}$ ,  $R$  – superior triunghiulară.

$$H^T \cdot H \cdot A = A = H^T \cdot R = Q \cdot R$$

Factorizarea QR admite și o “reprezentare economică”, obținută prin partiționarea matricei  $Q$ :

$$Q = \begin{bmatrix} Q_1 & Q_2 \end{bmatrix} \cdot R = \begin{bmatrix} R_1 \\ 0 \end{bmatrix} \begin{matrix} n \\ m-n \end{matrix}$$

$\begin{matrix} n & m-n & n & n \end{matrix}$

$$Q_1 \in \mathbb{R}^{m \times n}, Q_2 \in \mathbb{R}^{(m-n) \times n}, R_1 \in \mathbb{R}^{n \times n}.$$

$$Q^T = H = H_n H_{n-1} \dots H_2 H_1 I_m$$

$$Q_1 = \begin{bmatrix} Q_1 & Q_2 \end{bmatrix} \cdot \begin{bmatrix} I_n \\ 0 \end{bmatrix} = Q \cdot \begin{bmatrix} I_n \\ 0 \end{bmatrix} = H_1 H_2 \dots H_n \cdot \begin{bmatrix} I_n \\ 0 \end{bmatrix}$$

```
[x, Q, R] = Householder(A, b)
%Intrări: A = matricea sistemului
%          b = vectorul termenilor liberi
%Ieșiri: x = vectorul necunoscutelelor
% Q = matricea ortogonală
% R = matricea superior triunghiulară
Q = eye(m);
```

```

for k = 1:n
    s = norm(A(k:m,k)).^2;
    if A(k,k) < 0
        s = -s
    end;
    v(1:k-1) = 0;
    v(k) = A(k,k) + s;
    A(k,k) = -s;
    v(k+1:m) = A(k+1:m,k)
    A(k+1:m,k) = 0
    p = s * v(k);
% Ak+1 = Qk * Ak
    for j = k+1:n
        t = v(k:m)' * A(k:m,j) / p;
        A(k:m,j) = A(k:m,j) - t * v(k:m);
    end;
    % bk+1 = Qk * bk
    t = v(k:m)' * b(k:m) / p;
    b(k:m) = b(k:m) - t * v(k:m);
    % Q=Qk * Q
    for j = k+1:n
        t = v(k:m)' * Q(k:m,j) / p;
        Q(k:m,j) = Q(k:m,j) - t * v(k:m);
    end;
end;
% rezolvare sistem triunghiular
for i = n:-1:1
    s = A(i,i+1:n) * x(i+1:n);
    x(i) = (b(i) - s) / A(i,i);
end;
R = triu(A);
Q = Q';

[x, Q, R] = Givens(A, b)
% Intrări: A = matricea sistemului
%          b = vectorul termenilor liberi
% Ieșiri:  x = vectorul necunoscutelelor
%          Q = matricea factor ortogonală
%          R = factorul superior triunghiular
Q=eye(m);
t=zeros(2,n);
u=zeros(2,1);
for k = 1:n
    for l = k+1:m
        r = sqrt(A(k,k)^2 + A(l,k)^2);
        c = A(k,k) / r;
        s = A(l,k) / r;
        t = [c s; -s c] * [A(k,1:n); A(l,1:n)];
        A(k,1:n) = t(1);
        A(l,1:n) = t(2);
        u = [c s; -s c] * [b(k); b(l)];
        b(k) = t(1);

```

```

    b(1) = t(2);
    t = [c s;-s c] * [Q(k,1:m); Q(1,1:m)];
    Q(k,1:m) = t(1);
    Q(1,1:m) = t(2);
end
end;
for i = n:-1:1
    s = A(i,i+1:n) * x(i+1:n);
    x(i) = (b(i) - s) / A(i,i);
end
R = A; Q = Q';

```

În MATLAB apelul  $[Q,R] = \text{qr}(A)$  produce descompunerea matricei  $A$  într-o matrice superior triunghiulară  $R \in \mathbb{R}^{m \times n}$  (de aceeași dimensiune cu  $A$ ) și o matrice  $Q \in \mathbb{R}^{m \times m}$  unitară astfel încât  $A = Q \cdot R$ .

$[Q,R] = \text{qr}(A,0)$  produce o descompunere "economică", în care se calculează numai primele  $n$  coloane ale matricei  $Q$ .

### Ortogonalizarea Gram-Schmidt

$$[a_1 \ \dots \ a_j \ \dots \ a_n] = [q_1 \ \dots \ q_j \ \dots \ q_n] \cdot \begin{bmatrix} r_{11} & \dots & r_{1j} & \dots & r_{1n} \\ 0 & \dots & \dots & \dots & \dots \\ 0 & 0 & r_{jj} & \dots & r_{jn} \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & r_{nn} \end{bmatrix}$$

Coloanele matricei  $Q$  sunt ortogonale (formează o bază ortonormată):

$$q_i^T \cdot q_j = \begin{cases} 1, & i = j \\ 0, & i \neq j \end{cases}$$

$$a_1 = q_1 \cdot r_{11}$$

$$\|q_1\| = 1 \text{ de unde } r_{11} = \|a_1\| \text{ și } q_1 = a_1 / r_{11}$$

$$a_2 = q_1 r_{12} + q_2 r_{22}, \quad q_1^T a_2 = r_{12},$$

$$\|q_2\| = 1, \quad r_{22} = \|a_2 - q_1 r_{12}\|, \quad q_2 = (a_2 - q_1 r_{12}) / r_{22}$$

$$a_3 = q_1 r_{13} + q_2 r_{23} + q_3 r_{33}, \quad q_2^T a_3 = r_{23},$$

$$\|q_3\| = 1, \quad r_{33} = \|a_3 - q_1 r_{13} - q_2 r_{23}\|, \quad q_3 = (a_3 - q_1 r_{13} - q_2 r_{23}) / r_{33}$$

$$a_j = q_1 r_{1j} + q_2 r_{2j} + \dots + q_j r_{jj} \quad q_k^T a_j = r_{kj}, \quad k=1:j-1$$

$$\|q_j\| = 1, \quad r_{jj} = \left\| a_j - \sum_{k=1}^{j-1} q_k \cdot r_{kj} \right\| \quad q_j = \left( a_j - \sum_{k=1}^{j-1} q_k \cdot r_{kj} \right) / r_{jj}$$

pentru  $j=1:n$

$$q_j = a_j$$

pentru  $k=1:j-1$

$$r_{kj} = q_k^T \cdot q_j$$

$$q_j = a_j - \sum_{k=1}^{j-1} q_k r_{kj}$$

$$r_{jj} = \|q_j\|$$

$$q_j = q_j / r_{jj}$$

Algoritmul Gram-Schmidt clasic este numeric instabil. Rezultate mult mai bune ne furnizează o variantă modificată

$$\begin{aligned} q_1' a_k &= r_{1k} & q_1 q_1' a_k &= q_1 r_{1k} \\ (I - q_1 q_1') a_k &= q_1 r_{2k} + \dots + q_k r_{kk} & q_2' (I - q_1 q_1') a_k &= r_{2k} \\ q_2 q_2' (I - q_1 q_1') a_k &= q_2 r_{2k} & (I - q_1 q_1') (I - q_2 q_2') a_k &= q_3 r_{3k} + \dots + q_k r_{kk} \end{aligned}$$

```
Q = A
pentru k=1:n
    r_kk = ||q_k||
    q_k = q_k / r_kk
    pentru j=k+1:n
        r_kj = q_k' q_j
        q_j = q_j - q_k r_kj
```

### Algoritmul Gram-Schmidt clasic

```
[Q, R] = Gram_Schmidt(m, n, A)
% Intrări:
A = matricea de factorizat (baza inițială)
% Ieșiri: Q = factorul ortogonal (baza
% ortonormată)
% R = factorul superior triunghiular
[m,n]=size(A);
for i = 1 : n
    R(1:i-1,i) = Q(1:m,1:i-1)'*A(1:m,i);
    y = A(1:m,i)-Q(1:m,1:i-1)*R(1:i-1,i);
    R(i,i) = norm(y);
    Q(1:m,i) = y ./ R(i,i);
end
```

### Algoritmul Gram-Schmidt modificat

```
[Q, R] = Gram_Schmidt_Modificat(m, n, A)
% Intrări: A = matricea de factorizat
% (baza inițială)
% Ieșiri: Q = baza ortonormată
% R = factorul superior triunghiular
[m,n]=size(A);
for i = 1 : n
    R(i,i) = norm(A(1:m,i));
    Q(1:m,i) = A(1:m,i) / R(i,i);
    for j = i+1 : n
        R(i,j) = Q(1:m,i)' * A(1:m,j);
        A(1:m,j) = A(1:m,j) - Q(1:m,i)*R(i,j);
    end
end
```