Tema nr. 3

Se dau $\boldsymbol{n}$ dimensiunea sistemului, $\boldsymbol{\varepsilon}$ - precizia calculelor, matricea pătratică $\boldsymbol{A} \in \mathbb{R}^{\boldsymbol{n}\times\boldsymbol{n}}$ şi un vector $\boldsymbol{s} \in \mathbb{R}^{\boldsymbol{n}}$.

1. Să se calculeze vectorul $\boldsymbol{b} \in \mathbb{R}^{\boldsymbol{n}}$ astfel:

$$\boldsymbol{b_i = \sum_{j=1}^{n} s_j\, a_{ij} \quad , \; i = 1,\dots,n}$$

2. Să se implementeze descompunerea $\boldsymbol{QR}$ a matricei $\boldsymbol{A}$ folosind algoritmul lui Householder.
3. Să se rezolve sistemul liniar:

$$\boldsymbol{Ax = b} \quad ,$$

   folosind descompunerea $\boldsymbol{QR}$ din una din bibliotecile menționate în pagina laboratorului (se obţine soluţia $\boldsymbol{x_{QR}}$) şi descompunerea $\boldsymbol{QR}$ calculată la punctul 2. (se obţine soluţia $\boldsymbol{x_{Householder}}$). Calculaţi şi afişaţi:

$$\left\| \boldsymbol{x_{QR} - x_{Householder}} \right\|_2 .$$

4. Să se calculeze şi să se afişeze următoarele erori (|| ·||$_2$ este norma euclidiană):

$$\| \boldsymbol{A^{init} x_{Householder} - b^{init}} \|_2 \quad ,$$

$$\| \boldsymbol{A^{init} x_{QR} - b^{init}} \|_2 \quad ,$$

$$\frac{\| \boldsymbol{x_{Householder} - s} \|_2}{\| \boldsymbol{s} \|_2} \quad ,$$

$$\frac{\| \boldsymbol{x_{QR} - s} \|_2}{\| \boldsymbol{s} \|_2} .$$

   (aceste valori ar trebui să fie mai mici decât $\boldsymbol{10^{-6}}$)

5. Să se calculeze inversa matricei $\boldsymbol{A}$ folosind descompunerea $\boldsymbol{QR}$ calculată la punctul 2. şi să se compare cu inversa calculată folosind funcţia din bibliotecă, afişând valoarea următoarei norme:

$$\left\| \boldsymbol{A^{-1}_{Householder} - A^{-1}_{bibl}} \right\| .$$

6. Scrieţi programul şi cu iniţializare random a datelor de intrare (astfel încât programul vostru să poată fi rulat cu orice dimensiune $\boldsymbol{n}$ a datelor).

Bonus (15pt): Considerați o matrice $A \in \mathbb{R}^{n \times n}$ simetrică. Calculați limita aproximativă a urmatorului șir de matrice:

1) $k=0,\ A^{(0)}=A$;
2) Pentru $A^{(k)}$ se calculează descompunerea QR cu algoritmul Householder, $A^{(k)} = Q*R$;
3) $A^{(k+1)} = R*Q$;
4) Limita aproximativă a acestui şir de matrice este matricea $A^{(k+1)}$ care satisface proprietatea:

$$\left\| A^{(k+1)} - A^{(k)} \right\| \leq \varepsilon \ .$$

Ce formă are matricea $A^{(k+1)}$ şi ce reprezintă elementele acestei matrice pentru matricea inițială A?

Restricții de calcul:

a) Folosiți un număr cât mai mic de matrice la construcția șirului $A^{(k)}$;
b) La înmulțirea matricială $R*Q$ folosiți doar elementele din partea superior triunghiulară a matricei R (r_{ij} cu $i \leq j$), şi toate elementele matricei Q, fără a utiliza instrucțiuni „*if-then-else*”.

Rezolvarea sistemelor liniare folosind descompunerea *QR*

Fie A o matrice reală pătratică de dimensiune n. Presupunem că pentru matricea A avem o descompunere de forma:

$$A=Q*R$$

unde Q este matrice ortogonală ($Q^TQ = QQ^T = I_n$) iar R este matrice superior triunghiulară. Având o asemenea descompunere pentru matricea A, rezolvarea sistemului $Ax=b$ se reduce la rezolvarea sistemului superior triunghiular $Rx=Q^Tb$.

$$Ax=b \ \leftrightarrow \ QRx = b \ \leftrightarrow \ Q^TQRx = Q^T b \ \leftrightarrow \ Rx=Q^T b$$

$$Ax=b \quad \Leftrightarrow \quad Rx=Q^T b$$

Algoritmul lui Householder

Pentru a aduce sistemul ***Ax=b*** la forma $\boldsymbol{Rx = Q^T b}$ se folosesc matricele de reflexie. O matrice de reflexie $\boldsymbol{P = (p_{ij})_{i,j=\overline{1,n}}}$ are următoarea formă:

$$\boldsymbol{P = I_n - 2vv^T} \quad , \quad \boldsymbol{v} \in \mathbb{R}^n, \ \|\boldsymbol{v}\|_2 = |\boldsymbol{v}| = \sqrt{\sum_{i=1}^{n} \boldsymbol{v}_i^2} = \mathbf{1}.$$

(unde cu $\boldsymbol{I_n}$ am notat matricea unitate de dimensiune ***n***)

Se poate arăta că matricele de reflexie sunt simetrice şi ortogonale :

$$\boldsymbol{P = P^T} \quad , \quad \boldsymbol{P^2 = I_n}.$$

Algoritmul Householder de calcul al descompunerii ***QR*** se desfăşoară în ***(n-1)*** paşi. La pasul ***r*** se transformă coloana ***r*** a matricei ***A*** în formă superior triunghiulară fără a modifica primele $\boldsymbol{(r-1)}$ coloane. La acest pas se obţine coloana ***r*** a matricei ***R***. Calculul matricei ***R*** se poate face direct în matricea ***A*** (de fapt se transformă matricea ***A*** într-o matrice superior triunghiulară). În acelaşi timp, se fac transformările necesare asupra vectorului termenilor liberi şi se pot efectua transformări pentru a calcula matricea $\boldsymbol{Q^T}$. Pentru a calcula matricea $\boldsymbol{Q^T}$, se iniţializează o matrice $\boldsymbol{\bar{Q} = I_n}$ şi se fac aceleaşi trasformări asupra matricei $\boldsymbol{\bar{Q}}$ ca şi cele făcute asupra matricei ***A***.

Pasul *r (r=1,2,…,n-1)*

La acest pas, matricea ***A*** are primele ***(r-1)*** coloane în formă superior triunghiulară şi se foloseşte o matrice de reflexie pentru a transforma coloana ***r*** a matricei ***A*** în formă superior triunghiulară (fără a le modifica pe primele ***(r-1)***).
Pasul ***r*** constă în următoarele calcule :

$$\boldsymbol{A} = \boldsymbol{P}_r \boldsymbol{A}$$
$$\boldsymbol{b} = \boldsymbol{P}_r \boldsymbol{b}$$
$$\overline{\boldsymbol{Q}} = \boldsymbol{P}_r \overline{\boldsymbol{Q}}$$

unde matricea $\boldsymbol{P}_r$ se construieşte astfel :

$$\boldsymbol{P}_r = \boldsymbol{I}_n - \frac{1}{\beta} \boldsymbol{u}\boldsymbol{u}^T$$

$$\beta = \sigma - k a_{rr}, \quad \sigma = \sum_{j=r}^{n} a_{jr}^2 \ , \quad k = -\mathrm{semn}(a_{rr})\sqrt{\sigma} \ , \quad \mathbf{semn}(x) = \begin{cases} +1 & \textbf{dacă } x \geq 0, \\ -1 & \textbf{dacă } x < 0. \end{cases}$$

$$\boldsymbol{u} = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ a_{rr} - k \\ a_{r+1r} \\ \vdots \\ a_{nr} \end{pmatrix}, \qquad \begin{aligned} u_i &= 0,\ i = 1,\dots,r-1, \\ u_r &= a_{rr} - k, \\ u_i &= a_{ir},\ i = r+1,\dots,n \end{aligned}$$

Matricea $\boldsymbol{V} = \boldsymbol{u}\boldsymbol{u}^T \in \mathbb{R}^{n\times n}$, $\boldsymbol{V} = \left(v_{ij}\right)_{i,j=1,\dots,n}$ are următoarele elemente:

$$v_{ij} = \begin{cases} 0 & \text{pentru } i = 1,\dots,r-1 \ , \ j = 1,\dots,n \\ 0 & \text{pentru } i = r,\dots,n \ , \ j = 1,\dots,r-1 \\ u_i u_j & \text{pentru } i = r,\dots,n \ , \ j = r,\dots,n \end{cases}$$

Prin urmare, matricea de reflexie $\boldsymbol{P_r = (p_{ij})_{i,j=1,n}}$ are următoarele elemente:

$$p_{ij} = \begin{cases} \mathbf{0} & \text{pentru } \boldsymbol{i = 1,\ldots,r-1\,,\ j = 1,\ldots,n,\ i \neq j} \\ \mathbf{1} & \text{pentru } \boldsymbol{i = 1,\ldots,r-1,\ j = i} \\ \mathbf{0} & \text{pentru } \boldsymbol{i = r,\ldots,n\,,\ j = 1,\ldots,r-1} \\ \boldsymbol{-f\,u_i u_j} & \text{pentru } \boldsymbol{i = r,\ldots,n\,,\ j = r,\ldots,n,\ i \neq j} \\ \boldsymbol{1 - f\,u_i^2} & \text{pentru } \boldsymbol{i = r,\ldots,n\,,\ j = i} \end{cases} \quad , \quad \boldsymbol{f = \frac{1.0}{\beta}}$$

La sfârşitul algoritmului de mai jos, matricea superior triunghiulară $\boldsymbol{R}$ se găseşte în matricea $\boldsymbol{A}$, iar transpusa matricei $\boldsymbol{Q}$ în matricea $\boldsymbol{\bar{Q}}$. În vectorul $\boldsymbol{b}$ vom avea $\boldsymbol{Q^T b^{init}}$. Verificarea faptului că matricea iniţială nu este singulară, se face testând dacă toate elementele de pe diagonala matricei $\boldsymbol{R}$ (sau $\boldsymbol{A}$) sunt nenule ($|\boldsymbol{r_{ii}}| \leq \boldsymbol{\varepsilon}, \forall \boldsymbol{i}$) .

Algoritmul Householder pentru factorizarea ***A=QR***

$\widetilde{Q} = I_n;$

for $r = 1,\dots,n-1$

// **construcţia matricei** P_r **– constanta** β **şi vectorul** u

$\bullet\ \sigma = \sum_{i=r}^{n} a_{ir}^2;$

$\bullet$ **if** $(\sigma \le \varepsilon)$ **break**; $// \ r = r+1 \leftrightarrow P_r = I_n$ (A **singulară**)

$\bullet\ k = \sqrt{\sigma};$

$\bullet$ **if** $(a_{rr} > 0)\ k = -k;$

$\bullet\ \beta = \sigma - k\, a_{rr};$

$\bullet\ u_r = a_{rr} - k;\ \ u_i = a_{ir},\ i = r+1,\dots,n;$

$// \ A = P_r * A$

// **transformarea coloanelor** $j = r+1,\dots,n$

$\bullet$ **for** $j = r+1,\dots,n$

$*\ \gamma = (\gamma_j / \beta) = (Ae_j, u)/\beta = (\sum_{i=r}^{n} u_i a_{ij})/\beta;$

$*$ **for** $i = r,\dots,n$

$a_{ij} = a_{ij} - \gamma * u_i;$

// **transformarea coloanei** r **a matricei** A

$\bullet\ a_{rr} = k;\ a_{ir} = 0,\ \ i = r+1,\dots,n;$

$// \ b = P_r * b$

$\bullet\ \gamma = (\gamma / \beta) = (b, u)/\beta = (\sum_{i=r}^{n} u_i b_i)/\beta;$

$\bullet$ **for** $i = r,\dots,n \quad b_i = b_i - \gamma * u_i;$

$// \ \widetilde{Q} = P_r * \widetilde{Q}$

$\bullet$ **for** $j = 1,\dots,n$

$*\ \gamma = (\widetilde{Q}e_j, u)/\beta = (\sum_{i=r}^{n} u_i \tilde{q}_{ij})/\beta;$

$*$ **for** $i = r,\dots,n$

$\tilde{q}_{ij} = \tilde{q}_{ij} - \gamma * u_i;$

Calculul unei aproximări a inversei unei matrice

Dacă se cunoaște o metodă numerică de rezolvare a sistemelor liniare (în cazul de față se va folosi descompunerea $\boldsymbol{QR}$ calculată folosind algoritmul lui Householder), coloanele matricei inverse se pot aproxima rezolvând $\boldsymbol{n}$ sisteme liniare.

Coloana $\boldsymbol{j}$ a matricei $\boldsymbol{A^{-1}}$ se aproximează rezolvând sistemul liniar:

$$\boldsymbol{Ax = e_j} \ , \ \boldsymbol{e_j = (0,0,\ldots,\underbrace{1}_{\text{pozitia } j},\ldots,0)^T} \ , \ \boldsymbol{j = 1,\ldots,n} \ .$$

Procedura de calcul a matricei $\boldsymbol{A^{-1}_{Householder}}$ este următoarea:

- Se calculează descompunerea $\boldsymbol{QR}$ a matricei $\boldsymbol{A}$ folosind algoritmul Householder
- Dacă matricea $\boldsymbol{A}$ este singulară (det $\boldsymbol{A} = 0 \leftrightarrow \exists \ |r_{ii}| < \varepsilon$), STOP, inversa nu se poate calcula.
- Altfel, se calculează coloanele matricei $\boldsymbol{A^{-1}_{Householder}}$ astfel:

  for $\boldsymbol{j=1,..,n}$

  1. Se iniţializează vectorul $\boldsymbol{b = Q^T e_j}$ = coloana $\boldsymbol{j}$ a matricei $\boldsymbol{Q^T}$ sau linia $\boldsymbol{j}$ a matricei $\boldsymbol{Q}$;
     (se foloseşte iniţializarea cu coloana $\boldsymbol{j}$ a matricei $\boldsymbol{Q^T}$ sau linia $\boldsymbol{j}$ a matricei $\boldsymbol{Q}$ în funcţie de matricea returnată de algoritmul Householder)
  2. Se rezolvă sistemul superior triunghiular $\boldsymbol{Rx=b}$, folosind metoda substituţiei inverse. Se obţine soluţia $\boldsymbol{x^*}$ ($\boldsymbol{x^*}$ este şi soluţia sistemului liniar $\boldsymbol{Ax=e_j}$);
  3. Se memorează $\boldsymbol{x^*}$ în coloana $\boldsymbol{j}$ a matricei $\boldsymbol{A^{-1}_{Householder}}$.

Procedura descrisă mai sus detaliază rezolvarea numerică, folosind o descompunere $\boldsymbol{QR}$, a ecuaţiei matriceale:

$$\boldsymbol{AX=I_n} \ .$$

Exemplu

$$A = \begin{pmatrix} 0 & 0 & 4 \\ 1 & 2 & 3 \\ 0 & 1 & 2 \end{pmatrix}, \quad s = \begin{pmatrix} 3 \\ 2 \\ 1 \end{pmatrix}$$

$$\begin{array}{ccccccc} & & & & 4x_3 & = & 4 \\ x_1 & + & 2x_2 & + & 3x_3 & = & 10 \\ & & x_2 & + & 2x_3 & = & 4 \end{array}$$

$$A = \begin{pmatrix} 0 & 0 & 4 \\ 1 & 2 & 3 \\ 0 & 1 & 2 \end{pmatrix} = QR = \begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} * \begin{pmatrix} 1 & 2 & 3 \\ 0 & 1 & 2 \\ 0 & 0 & 4 \end{pmatrix}, \quad b = A * s = \begin{pmatrix} 4 \\ 10 \\ 4 \end{pmatrix}$$

$$Ax = b \Leftrightarrow Rx = Q^T b \Leftrightarrow \begin{array}{ccccccc} x_1 & + & 2x_2 & + & 3x_3 & = & 10 \\ & & x_2 & + & 2x_3 & = & 4 \\ & & & & 4x_3 & = & 4 \end{array} \Rightarrow x^* = \begin{pmatrix} 3 \\ 2 \\ 1 \end{pmatrix}$$

Prima coloană a matricei $A^{-1}_{Householder}$

$$\begin{array}{ccccccc} x_1 & + & 2x_2 & + & 3x_3 & = & 0 \\ & & x_2 & + & 2x_3 & = & 0 \\ & & & & 4x_3 & = & 1 \end{array} \Rightarrow x^* = \begin{pmatrix} 0.25 \\ -0.5 \\ 0.25 \end{pmatrix}$$

Coloana a 2-a a matricei $A^{-1}_{Householder}$

$$\begin{array}{ccccccc} x_1 & + & 2x_2 & + & 3x_3 & = & 1 \\ & & x_2 & + & 2x_3 & = & 0 \\ & & & & 4x_3 & = & 0 \end{array} \Rightarrow x^* = \begin{pmatrix} 1.0 \\ 0.0 \\ 0.0 \end{pmatrix}$$

Coloana a 3-a a matricei $A^{-1}_{Householder}$

$$\begin{array}{ccccccc} x_1 & + & 2x_2 & + & 3x_3 & = & 0 \\ & & x_2 & + & 2x_3 & = & 1 \\ & & & & 4x_3 & = & 0 \end{array} \Rightarrow x^* = \begin{pmatrix} -2.0 \\ 1.0 \\ 0.0 \end{pmatrix}$$

$$A_{Householder}^{-1} = \begin{pmatrix} 0.25 & 1.0 & -2.0 \\ -0.5 & 0.0 & 1.0 \\ 0.25 & 0.0 & 0.0 \end{pmatrix}$$