Calcul Numeric

Cursul 5

2020

Anca Ignat

Descompunerea Cholesky

O matrice $\boldsymbol{A} \in \mathbb{R}^{n \times n}$ se numeşte ***pozitiv definit***ă dacă:

$$\left(\boldsymbol{Ax}, \boldsymbol{x}\right)_{\mathbb{R}^n} > 0 \quad \forall\, \boldsymbol{x} \in \mathbb{R}^n,\ \boldsymbol{x} \neq \boldsymbol{0}$$

Notaţie: $\boldsymbol{A} > \boldsymbol{0}$

Fie $\boldsymbol{A} \in \mathbb{R}^{n \times n}$ o matrice simetrică ($\boldsymbol{A}=\boldsymbol{A}^T$) şi pozitiv definită.

Descompunerea Cholesky pentru matricea $\boldsymbol{A}$ este de forma:

$\boldsymbol{A} = \boldsymbol{LL}^T$, $\boldsymbol{L}$ matrice inferior triunghiulară

$$
A=\begin{pmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & & \\ a_{n1} & \cdots & a_{nn} \end{pmatrix} = LL^T = \begin{pmatrix} l_{11} & 0 & \cdots & 0 \\ l_{21} & l_{22} & \cdots & 0 \\ \vdots & & & \\ l_{n1} & l_{n2} & \cdots & l_{nn} \end{pmatrix} \begin{pmatrix} l_{11} & l_{21} & \cdots & l_{n1} \\ 0 & l_{22} & \cdots & l_{n2} \\ \vdots & & & \\ 0 & 0 & \cdots & l_{nn} \end{pmatrix}
$$

Matricea L se calculează în n paşi, coloană după coloană.

Pas *r (r=1,…,n)*

Se calculează elementele coloanei ***r*** a matricii ***L***:
întâi elementul diagonal l_{rr} apoi
celelalte elemente l_{ir} ***(i=r+1,…n)***

Coloana ***r*** a matricii ***L:***

$$\begin{pmatrix}0 & \cdots & 0 & l_{rr} & l_{r+1r} & \cdots & l_{ir} & \cdots & l_{nr}\end{pmatrix}^T$$

- se cunosc elementele primelor ***(r-1)*** coloane ale matricii ***L***

Calcul $\boldsymbol{l_{rr}}$:

$$a_{rr} = \left(LL^T\right)_{rr} = \begin{pmatrix} l_{r1} & \cdots & l_{rr-1} & l_{rr-1} & 0 & \cdots & 0 \end{pmatrix} \begin{pmatrix} l_{r1} \\ \vdots \\ l_{rr-1} \\ l_{rr} \\ 0 \\ \vdots \\ 0 \end{pmatrix} =$$

$$= \ l_{r1}^2 + l_{r2}^2 + \cdots + l_{rr-1}^2 + l_{rr}^2 \quad \Rightarrow \quad l_{rr} = \pm \sqrt{a_{rr} - \sum_{k=1}^{r-1} l_{rk}^2}$$

Calcul $\boldsymbol{l_{ir}}$ $\boldsymbol{(i=r+1,\ldots,n)}$:

$$a_{ir} = \left(LL^T\right)_{ir} = \begin{pmatrix} l_{i1} & \cdots & l_{ir-1} & l_{ir} & \cdots & l_{ii} & \cdots & 0 \end{pmatrix} \begin{pmatrix} l_{r1} \\ \vdots \\ l_{rr-1} \\ l_{rr} \\ 0 \\ \vdots \\ 0 \end{pmatrix} =$$

$$= l_{i1}l_{r1} + l_{i2}l_{r2} + \cdots + l_{ir-1}l_{rr-1} + l_{ir}l_{rr} \quad \Rightarrow \quad l_{ir} = \frac{\left(a_{ir} - \sum_{k=1}^{r-1} l_{ik}l_{rk}\right)}{l_{rr}}$$

Algoritmul de eliminare Gauss fără schimbare de ecuații $\Leftrightarrow$ descompunere LU

Presupunem că la fiecare pas al algoritmului de eliminare Gauss pivotul este nenul ($a_{rr}^{(r-1)} \neq 0$), deci nu e nevoie de schimbare de ecuații.

Algoritmul se poate scrie astfel:

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

for $i = r+1, \ldots, n$

- $f = -\dfrac{a_{ir}}{a_{rr}};$

$// \; E_i = E_i + f * E_r$

- for $j = r+1, \ldots, n$

  $a_{ij} = a_{ij} + f * a_{rj};$

- $a_{ir} = 0;$
- $b_i = b_i + f * b_r;$

Considerăm vectorul și matricea:

$$\boldsymbol{t}^{(r)} = \begin{pmatrix} \mathbf{0} \\ \vdots \\ \mathbf{0} \\ \boldsymbol{t}_{r+1}^{(r)} \\ \vdots \\ \boldsymbol{t}_{n}^{(r)} \end{pmatrix} \in \mathbb{R}^{n} \quad , \quad \boldsymbol{T}_r := \boldsymbol{I}_n + \boldsymbol{t}^{(r)} \boldsymbol{e}_r^{T} \in \mathbb{R}^{n \times n}$$

$$\boldsymbol{t}^{(r)}\boldsymbol{e}_r^T = \begin{pmatrix} \mathbf{0} \\ \vdots \\ \mathbf{0} \\ \boldsymbol{t}_{r+1}^{(r)} \\ \vdots \\ \boldsymbol{t}_n^{(r)} \end{pmatrix} \begin{pmatrix} \mathbf{0} & \cdots & \mathbf{1} & \mathbf{0} \cdots \mathbf{0} \end{pmatrix} = \begin{pmatrix} \mathbf{0} & \mathbf{0} & \cdots & \overbrace{\mathbf{0}}^{\text{col r}} & \cdots & \mathbf{0} & \\ \vdots & & & & & & \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \cdots & \mathbf{0} & \\ \mathbf{0} & \mathbf{0} & \cdots & \boldsymbol{t}_{r+1}^{(r)} & \cdots & \mathbf{0} & -\,\mathrm{lin}\,(\boldsymbol{r}+\mathbf{1}) \\ \vdots & & & & & & \\ \mathbf{0} & \mathbf{0} & \cdots & \boldsymbol{t}_n^{(r)} & \cdots & \mathbf{0} & \end{pmatrix}$$

Matricea $\boldsymbol{T_r}$ este matrice triunghiulară inferior cu ***1*** pe diagonala principală:

$$\begin{array}{cc} & \begin{array}{cccccc} & & & \textbf{col } \boldsymbol{r} & & \end{array} \\ \boldsymbol{T_r} = & \begin{pmatrix} \mathbf{1} & \mathbf{0} & \cdots & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{1} & \cdots & \mathbf{0} & \cdots & \mathbf{0} \\ \vdots & & & & & \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{1} & \cdots & 0 \\ 0 & 0 & \cdots & \boldsymbol{t_{r+1}^{(r)}} & \cdots & 0 \\ \vdots & & & & & \\ 0 & 0 & \cdots & \boldsymbol{t_n^{(r)}} & \cdots & 1 \end{pmatrix} \end{array}$$

Inversa matricei $\boldsymbol{T_r}$ este

$$\boldsymbol{T_r^{-1} = I_n - t^{(r)} e_r^T} \; .$$

$$\boldsymbol{T_r\, T_r^{-1} = (I_n + t^{(r)} e_r^T)(I_n - t^{(r)} e_r^T) = I_n - t^{(r)} e_r^T + t^{(r)} e_r^T - t^{(r)} e_r^T t^{(r)} e_r^T =}$$

$$\boldsymbol{= I_n - t^{(r)}\, t_r^{(r)} e_r^T = I_n \quad (0 = t_r^{(r)} = e_r^T t^{(r)})}$$

Dacă $\boldsymbol{A}$ este o matrice oarecare, vrem să vedem cum se poate construi matricea $\boldsymbol{B=T_r\, A}$ fără a face înmulțire matricială. Vom studia legătura între liniile matricelor $\boldsymbol{A}$ și $\boldsymbol{B}$.

$$e_i^T B = e_i^T (T_r A) = e_i^T (I_n + t^{(r)} e_r^T) A = e_i^T A + e_i^T t^{(r)} e_r^T A =$$
$$= e_i^T A + t_i^{(r)} (e_r^T A)$$

Linia i a noii matrice B se obține din linia i a matricei A la care se adaugă linia r a matricei A înmulțită cu factorul $t_i^{(r)}$.

$$e_i^T B = \begin{cases} e_i^T A & i = 1, \ldots, r \ (t_i^{(r)} = 0) \\ e_i^T A + t_i^{(r)} (e_r^T A) & i = r+1, \ldots, n \end{cases} .$$

Operația T_rA descrie Pasul r al algoritmului de eliminare Gauss dacă:

$$t_{r+1}^{(r)} = -\frac{a_{r+1r}^{(r)}}{a_{rr}^{(r)}}, \cdots, t_i^{(r)} = -\frac{a_{ir}^{(r)}}{a_{rr}^{(r)}}, \cdots, t_n^{(r)} = -\frac{a_{nr}^{(r)}}{a_{rr}^{(r)}}.$$

Algoritmul de eliminare Gauss fără schimbare de ecuații poate fi descris astfel:

$$T_{n-1} \cdots T_2 T_1 A = U \quad \text{cu} \quad T_r = I_n + t^{(r)} e_r^T,$$

$$t^{(r)} = \left(0 \ \cdots 0 \ \left(-\frac{a_{r+1r}^{(r)}}{a_{rr}^{(r)}}\right) \ \cdots \left(-\frac{a_{ir}^{(r)}}{a_{rr}^{(r)}}\right) \ \cdots \left(-\frac{a_{nr}^{(r)}}{a_{rr}^{(r)}}\right) \right)^T.$$

Avem:

$$A = T_1^{-1}T_2^{-1}\cdots T_{n-1}^{-1}\,U = LU \quad , \quad L := T_1^{-1}T_2^{-1}\cdots T_{n-1}^{-1}$$

$$T_1^{-1}T_2^{-1} = (I_n - t^{(1)}e_1^T)(I_n - t^{(2)}e_2^T) = I_n - t^{(1)}e_1^T - t^{(2)}e_2^T + t^{(1)}e_1^T t^{(2)}e_2^T =$$
$$= I_n - t^{(1)}e_1^T - t^{(2)}e_2^T + t^{(1)}t_1^{(2)}e_2^T = I_n - t^{(1)}e_1^T - t^{(2)}e_2^T \quad (t_1^{(2)} = 0)$$

Prin inducție se arată că:

$$L = T_1^{-1}T_2^{-1}\cdots T_{n-1}^{-1} = I_n - t^{(1)}e_1^T - t^{(2)}e_2^T - \cdots - t^{(n-1)}e_{n-1}^T$$

$$
L = \begin{pmatrix}
1 & 0 & \cdots & 0 & \cdots & 0 \\
\dfrac{a_{21}}{a_{11}} & 1 & \cdots & 0 & \cdots & 0 \\
\dfrac{a_{31}}{a_{11}} & \dfrac{a_{32}^{(1)}}{a_{22}^{(1)}} & \cdots & 0 & \cdots & 0 \\
\vdots & & & & & \\
\dfrac{a_{r1}}{a_{11}} & \dfrac{a_{r2}^{(1)}}{a_{22}^{(1)}} & \cdots & 1 & \cdots & 0 \\
\dfrac{a_{r+11}}{a_{11}} & \dfrac{a_{r+12}^{(1)}}{a_{22}^{(1)}} & \cdots & \dfrac{a_{r+1r}^{(r-1)}}{a_{rr}^{(r-1)}} & \cdots & 0 \\
\vdots & & & & & \\
\dfrac{a_{n1}}{a_{11}} & \dfrac{a_{n2}^{(1)}}{a_{22}^{(1)}} & \cdots & \dfrac{a_{nr}^{(r-1)}}{a_{rr}^{(r-1)}} & \cdots & 1
\end{pmatrix}
$$

Descompuneri *QR*

Definiție

Se numește ***matrice ortogonală***, o matrice $\boldsymbol{Q} \in \mathbb{R}^{n \, x \, n}$ care satisface relația:

$$\boldsymbol{Q^T Q = Q Q^T = I_n \quad (Q^{-1} = Q^T)}\,.$$

Matricele ortogonale au următoarele proprietăți:

- Dacă $\boldsymbol{Q}$ este matrice ortogonală atunci şi matricea transpusă $\boldsymbol{Q^T}$ este ortogonală.

$$\boldsymbol{Q^T Q = Q^T (Q^T)^T = Q Q^T = (Q^T)^T Q^T = I_n}$$

- Dacă Q_1 şi Q_2 sunt matrice ortogonale atunci Q_1Q_2 este tot matrice ortogonală.

$$(Q_1Q_2)^T(Q_1Q_2) = Q_2^TQ_1^TQ_1Q_2 = Q_2^T I\, Q_2 = I_n$$
$$(Q_1Q_2)(Q_1Q_2)^T = Q_1Q_2Q_2^TQ_1^T = Q_1^T I\, Q_1 = I_n$$

- Dacă $Q \in \mathbb{R}^{n\times n}$ este matrice ortogonală şi $x \in \mathbb{R}^n$ atunci $\|Qx\|_2 = \|x\|_2$.

$$\|Qx\|_2^2 = \left(Qx, Qx\right) = \left(x, Q^TQx\right) = \left(x, x\right) = \|x\|_2^2, \quad \|\cdot\|_2 \geq 0 \Rightarrow$$
$$\|Qx\|_2 = \|x\|_2$$

Fie $\boldsymbol{A}$ o matrice reală pătratică de dimensiune $\boldsymbol{n}$. Pp. că avem:

$$\boldsymbol{A} = \boldsymbol{QR}$$

unde $\boldsymbol{Q}$ este o matrice ortogonală iar $\boldsymbol{R}$ este o matrice superior triunghiulară.

$$\boldsymbol{Ax} = \boldsymbol{b} \quad \Leftrightarrow \quad \boldsymbol{QRx} = \boldsymbol{b} \quad \Leftrightarrow \quad \boldsymbol{Q}^T\boldsymbol{QRx} = \boldsymbol{Q}^T\boldsymbol{b} \quad \Leftrightarrow \quad \boldsymbol{Rx} = \boldsymbol{Q}^T\boldsymbol{b}$$

Algoritmul lui Householder

***Matrice de reflexie* - $P \in \mathbb{R}^{n\times n}$** de forma:

$$P = I_n - 2vv^T \ , \quad v \in \mathbb{R}^n \ , \quad \| v \|_2 = \sqrt{\sum_{j=1}^{n} | v_j |^2} = 1$$

$$vv^T = \begin{pmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{pmatrix} (v_1, v_2, \ldots, v_n) = \begin{pmatrix} v_1^2 & v_1 v_2 & \ldots & v_1 v_n \\ v_2 v_1 & v_2^2 & \ldots & v_2 v_n \\ \vdots & & & \\ v_n v_1 & v_n v_2 & \ldots & v_n^2 \end{pmatrix}.$$

Matricele de reflexie sunt :

simetrice - $\boldsymbol{P} = \boldsymbol{P}^T$ şi

ortogonale - $\boldsymbol{P}\boldsymbol{P}^T = \boldsymbol{P}^T\boldsymbol{P} = \boldsymbol{P}^2 = \boldsymbol{I}_n$.

$$\boldsymbol{P}^T = (\boldsymbol{I}_n - 2\boldsymbol{v}\boldsymbol{v}^T)^T = \boldsymbol{I}_n - 2(\boldsymbol{v}\boldsymbol{v}^T)^T = \boldsymbol{I}_n - 2(\boldsymbol{v}^T)^T\boldsymbol{v}^T = \boldsymbol{I}_n - 2\boldsymbol{v}\boldsymbol{v}^T = \boldsymbol{P}$$

$$P^2 = (I_n - 2vv^T)(I_n - 2vv^T) = I_n - 2vv^T - 2vv^T + 4(vv^T)(vv^T) =$$

$$= I_n - 4vv^T + 4v(v^T v)v^T = I_n - 4vv^T + 4v\|v\|_2^2 v^T =$$

$$= I_n - 4vv^T + 4vv^T = I_n \quad (\|v\|_2 = 1)$$

$$n = 2, \quad v = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad P = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} - 2\begin{pmatrix} 1 \\ 0 \end{pmatrix}(1 \quad 0) = \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix}, \; x \in \mathbb{R}^2, \; y = Px$$

$$x = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}, \quad y = \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} -x_1 \\ x_2 \end{pmatrix}$$

Vectorul $\boldsymbol{y=Px}$ este reflectatul vectorului $\boldsymbol{x}$ în raport cu axa $\boldsymbol{Ox_2}$.

Algoritmul care folosește matricele de reflexie pentru a obține o descompunere ***QR*** pentru o matrice $\boldsymbol{A} \in \mathbb{R}^{\boldsymbol{n \times n}}$ a fost descris de Alston S. Householder în articolul "*Unitary triangularization of a nonsymmetric matrix*" apărut în Journal of the Assoc. of Computing Machinery 5 (1958), 339-342.

Transformarea matricei ***A*** într-una superior triunghiulară se face în ***(n-1)*** pași, la fiecare pas folosindu-se o matrice de reflexie.

Pas 1: $\boldsymbol{A^{(1)} = P_1 A}$ (matricea $\boldsymbol{P_1}$ se alege astfel încât col. ***1*** să fie transformată în formă superior triunghiulară)

Pas 2: $\boldsymbol{A^{(2)} = P_2 A^{(1)} = P_2 (P_1 A)}$ ($\boldsymbol{P_2}$ transformă col. ***2*** în formă superior triunghiulară, fără să schimbe col. ***1***)

Pas *r*: $\boldsymbol{A^{(r)} = P_r A^{(r-1)} = P_r (P_{r-1} \dots P_1 A)}$ (se transformă col ***r*** în formă superior triunghiular fără să schimbe primele ***(r-1)*** coloane)

Descompunerea QR construită cu algoritmul Householder este următoarea:

$$P_{n-1} \cdots P_r \cdots P_2 P_1 A = \tilde{Q} A = R$$

unde

$$\tilde{Q} = P_{n-1} \cdots P_r \cdots P_2 P_1$$

$\tilde{Q}$ este matrice ortogonală ca produs de matrice ortogonale.

$$\tilde{Q} A = R \quad \Leftrightarrow \quad \tilde{Q}^T Q A = \tilde{Q}^T R \quad \Leftrightarrow \quad A = \tilde{Q}^T R = QR$$

$$Q = \tilde{Q}^T = (P_{n-1} \cdots P_r \cdots P_2 P_1)^T = P_1 P_2 \cdots P_r \cdots P_{n-1}$$

Pasul *r*

La intrarea în pasul ***r*** matricea ***A*** are forma:

$$
\boldsymbol{A} = \begin{pmatrix}
a_{11} & a_{12} & \cdots & a_{1r} & \cdots & a_{1n} \\
0 & a_{22} & \cdots & a_{2r} & \cdots & a_{2n} \\
\vdots & & & & & \\
0 & 0 & \cdots & a_{rr} & \cdots & a_{rn} \\
0 & 0 & \cdots & a_{r+1r} & \cdots & a_{r+1n} \\
\vdots & & & & & \\
0 & 0 & \cdots & a_{ir} & \cdots & a_{in} \\
\vdots & & & & & \\
0 & 0 & \cdots & a_{nr} & \cdots & a_{nn}
\end{pmatrix}
$$

Pasul ***r*** constă în:

$$\boldsymbol{A := P_r A}$$

$$\boldsymbol{P_r = I_n - 2v^r (v^r)^T \ , \quad v^r \in R^n \ , \quad \| v^r \|_2 = 1}$$

unde vectorul $\boldsymbol{v}^{\boldsymbol{r}}$ se alege astfel ca matricea ***A*** să aibă şi coloana ***r*** în formă superior triunghiulară:

$$A = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1r} & \cdots & a_{1n} \\ 0 & a_{22} & \cdots & a_{2r} & \cdots & a_{2n} \\ \vdots & & & & & \\ 0 & 0 & \cdots & a_{rr} & \cdots & a_{rn} \\ 0 & 0 & \cdots & 0 & \cdots & a_{r+1n} \\ \vdots & & & & & \\ 0 & 0 & \cdots & 0 & \cdots & a_{in} \\ \vdots & & & & & \\ 0 & 0 & \cdots & 0 & \cdots & a_{nn} \end{pmatrix}$$

Calculul matricei P_r

Pentru simplitate vom nota $\boldsymbol{P_r=P}$, $\boldsymbol{v^r=v}$.

$$Ae_r = \begin{pmatrix} a_{1r} \\ a_{2r} \\ \vdots \\ a_{r-1r} \\ a_{rr} \\ a_{r+1r} \\ \vdots \\ a_{ir} \\ \vdots \\ a_{nr} \end{pmatrix} \rightarrow (PA)e_r = \overline{A}e_r = \begin{pmatrix} \overline{a}_{1r} = a_{1r} \\ \overline{a}_{2r} = a_{2r} \\ \vdots \\ \overline{a}_{r-1r} = a_{r-1r} \\ \overline{a}_{rr} = k \\ 0 \\ \vdots \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

Aplicând proprietatea matricelor ortogonale :

$$Q \in \mathbb{R}^{n\times n},\ x \in \mathbb{R}^{n},\ \| Qx \|_2 = \| x \|_2$$

pentru matricea $Q=P$ şi vectorul $x=Ae_r$ avem:

$$\| PAe_r \|_2^2 = a_{1r}^2 + a_{2r}^2 + \cdots + a_{r-1r}^2 + k^2 =$$

$$\| Ae_r \|^2 = a_{1r}^2 + a_{2r}^2 + \cdots + a_{r-1r}^2 + a_{rr}^2 + a_{r+1r}^2 + \cdots + a_{ir}^2 + \cdots + a_{nr}^2$$

Din relația de mai sus rezultă:

$$k^2 = \sigma = a_{rr}^2 + a_{r+1r}^2 + \cdots + a_{ir}^2 + \cdots + a_{nr}^2 = \sum_{i=r}^{n} a_{ir}^2 \Rightarrow k = \pm\sqrt{\sigma}$$

Determinarea vectorului ***v*** ce definește matricea ***P:***

$$(\boldsymbol{PA})\boldsymbol{e}_r = (\boldsymbol{I}_n - 2\boldsymbol{v}\boldsymbol{v}^T)(\boldsymbol{A}\boldsymbol{e}_r) = \boldsymbol{A}\boldsymbol{e}_r - 2(\boldsymbol{v}\boldsymbol{v}^T)(\boldsymbol{A}\boldsymbol{e}_r) =$$

$$= \boldsymbol{A}\boldsymbol{e}_r - 2\boldsymbol{v}(\boldsymbol{v}^T(\boldsymbol{A}\boldsymbol{e}_r)) = \boldsymbol{A}\boldsymbol{e}_r - (2\alpha)\boldsymbol{v} = \boldsymbol{A}\boldsymbol{e}_r - \boldsymbol{u}$$

unde cu α și ***u*** am notat:

$$\alpha := v^T(Ae_r) = \left((Ae_r), v\right)_{\mathbb{R}^n} = \left(\begin{pmatrix} a_{1r} \\ a_{2r} \\ \vdots \\ a_{ir} \\ \vdots \\ a_{nr} \end{pmatrix}, \begin{pmatrix} v_1 \\ v_2 \\ \vdots \\ v_i \\ \vdots \\ v_n \end{pmatrix}\right)_{\mathbb{R}^n} =$$

$$= a_{1r}v_1 + a_{2r}v_2 + \cdots + a_{ir}v_i + \cdots + a_{nr}v_n$$

$$u := (2\alpha)\, v = Ae_r - (PA)e_r = \begin{pmatrix} a_{1r} \\ a_{2r} \\ \vdots \\ a_{r-1r} \\ a_{rr} \\ a_{r+1r} \\ \vdots \\ a_{ir} \\ \vdots \\ a_{nr} \end{pmatrix} - \begin{pmatrix} a_{1r} \\ a_{2r} \\ \vdots \\ a_{r-1r} \\ k \\ 0 \\ \vdots \\ 0 \\ \vdots \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ a_{rr} - k \\ a_{r+1r} \\ \vdots \\ a_{ir} \\ \vdots \\ a_{nr} \end{pmatrix}$$

Cu aceste notații, matricea ***P*** devine:

$$\boldsymbol{P} = \boldsymbol{I}_n - 2(\frac{1}{2\alpha})\boldsymbol{u}(\frac{1}{2\alpha})\boldsymbol{u}^T = \boldsymbol{I}_n - (\frac{1}{2\alpha^2})\boldsymbol{u}\boldsymbol{u}^T = \boldsymbol{I}_n - \frac{1}{\beta}\boldsymbol{u}\boldsymbol{u}^T$$

$$\beta := 2\alpha^2$$

Pentru a cunoaște matricea ***P*** trebuie să mai determinăm constanta β. Din condiția:

$$\|\boldsymbol{v}\|_2^2 = 1 \Rightarrow \|\frac{1}{2\alpha}\boldsymbol{u}\|_2^2 = 1 \Rightarrow \frac{1}{4\alpha^2}\|\boldsymbol{u}\|_2^2 = 1 \Rightarrow 2\beta = \|\boldsymbol{u}\|_2^2$$

$$\| \mathbf{u} \|_2^2 = \left\| \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ a_{rr} - k \\ a_{r+1r} \\ \vdots \\ a_{ir} \\ \vdots \\ a_{nr} \end{pmatrix} \right\|_2^2 = (a_{rr} - k)^2 + a_{r+1r}^2 + \cdots + a_{ir}^2 + \cdots + a_{nr}^2 =$$

$$= a_{rr}^2 + a_{r+1r}^2 + \cdots + a_{ir}^2 + \cdots + a_{nr}^2 - 2ka_{rr} + k^2 =$$

$$= \sigma - 2ka_{rr} + \sigma = 2(\sigma - ka_{rr})$$

de unde obținem:

$$\beta = \sigma - k a_{rr}$$

Vom alege semnul constantei ***k*** astfel încât ***β*** să fie cât mai mare posibil deoarece constanta ***β*** apare în operația de împărțire. Avem:

$$\beta \quad \text{"}mare\text{"} \;\rightarrow\; \beta = \sigma - k a_{rr} \ge \sigma \quad (\sigma \ge 0) \quad \rightarrow$$

$$\textbf{semn}\, k = -\textbf{semn}\, a_{rr}$$

Ce înseamnă ***β =0*** ?

$$\beta = \frac{1}{2} \| u \|_2^2 = 0 \rightarrow \| u \|_2^2 = 0 \rightarrow u = 0 \rightarrow$$

$$\rightarrow a_{rr} = k, a_{r+1r} = 0, \dots, a_{ir} = 0, \dots, a_{nr} = 0$$

Cum $\boldsymbol{a_{rr}=k}$ şi **semn** $\boldsymbol{k}$ **= -semn** $\boldsymbol{a_{rr}}$ obținem:

$$a_{ir} = 0, \quad \forall i = r, \dots, n$$

adică avem coloana $\boldsymbol{r}$ deja în formă superior triunghiulară, se poate trece la pasul următor. În acest caz matricea $\boldsymbol{A}$ este singulară.

Ne interesează cum se efectuează operația $\boldsymbol{A=P_rA}$ fără a face înmulțire matricială. Vom pune în evidentă schimbările în raport cu coloanele.

$$(PA)e_j = \text{ noua col. } j \text{ a matr. } A = \left(I_n - \frac{1}{\beta}uu^T\right)(Ae_j) =$$

$$= Ae_j - \frac{1}{\beta}(uu^T)(Ae_j) = Ae_j - \frac{1}{\beta}u(u^T(Ae_j)) =$$

$$= Ae_j - \frac{\gamma_j}{\beta}u$$

$$\gamma_j := u^T(Ae_j) = \left(\begin{pmatrix} a_{1j} \\ a_{2j} \\ \vdots \\ a_{rj} \\ a_{r+1\,j} \\ \vdots \\ a_{nj} \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ \vdots \\ a_{rr} - k \\ a_{r+1\,r} \\ \vdots \\ a_{nr} \end{pmatrix} \right)_{\mathbb{R}^n} =$$

$$= a_{rj}(a_{rr} - k) + \cdots + a_{ij}a_{ir} + \cdots + a_{nj}a_{nr} = \sum_{i=1}^{n} u_i a_{ij} = \sum_{i=r}^{n} u_i a_{ij}$$

$$(u_i = 0,\ i = 1, \ldots, r-1,\ u_r = a_{rr} - k,\ u_i = a_{ir},\ i = r+1, \ldots, n)$$

Noua coloană ***j*** se obține din vechea coloană ***j*** din care scădem vectorul ***u*** înmulțit cu constanta γ_j. Ne interesează ca primelor ***(r-1)*** coloane să nu li se schimbe forma superior triunghiulară deja obținută.

Pentru ***j=1,…,(r-1)*** avem:

$$\gamma_j := u^T(Ae_j) = \left(\begin{pmatrix} a_{1j} \\ a_{2j} \\ \vdots \\ a_{jj} \\ a_{j+1j} = 0 \\ \vdots \\ a_{rj} = 0 \\ \vdots \\ a_{nj} = 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ a_{rr} - k \\ a_{r+1r} \\ \vdots \\ a_{nr} \end{pmatrix} \right)_{\mathbb{R}^n} =$$

$$= a_{1j}0 + \cdots + a_{jj}0 + \cdots + 0(a_{rr} - k) + \cdots + 0a_{ir} + \cdots + 0a_{nr} = 0$$

Din faptul că $\gamma_j = 0,\ j = 1,...,r-1$ rezultă că primele ***(r-1)*** coloane ale matricei ***A*** nu se schimbă când facem operația ***A=P$_r$A***, rămân în formă superior triunghiulară.

Algoritmul de trecere de la matricea ***A*** la matricea ***P$_r$A*** este următorul:

$$(P_r A)e_j = \begin{cases} Ae_j & \textbf{pentru } j = 1,\ldots,r-1 \\ (a_{1r}, a_{2r}, \ldots, a_{r-1r}, k, 0, \ldots, 0)^T & \textbf{pentru } j = r \\ Ae_j - \dfrac{\gamma_j}{\beta} u & \textbf{pentru } j = r+1,\ldots,n \end{cases}$$

$$\gamma_j = (Ae_j, u)_{\mathbb{R}^n} = \sum_{i=r}^{n} u_i a_{ij}$$

$$u_i = 0, \quad i = 1, \ldots, r-1, \quad u_r = a_{rr} - k, \qquad u_i = a_{ir}, \quad i = r+1, \ldots, n$$

Operaţia de transformare a vectorului termenilor liberi

b:=P$_r$b:

$$P_r b = (I_n - \frac{1}{\beta}(uu^T))b = b - \frac{1}{\beta}(uu^T)b = b - \frac{1}{\beta}u\,(u^T b) = b - \frac{\gamma}{\beta}u$$

$$\gamma = u^T b = (b, u)_{\mathbb{R}^n} = \sum_{i=r}^{n} u_i b_i$$

Algoritmul lui Householder

$$\widetilde{Q} = I_n;$$

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

$$\begin{cases} \textbf{calculează matricea } P_r \textbf{ (constanta } \beta \textbf{ şi vectorul } u\textbf{)} \\ A = P_r * A; \\ b = P_r * b; \\ \widetilde{Q} = P_r * \widetilde{Q}; \end{cases}$$

La sfârșitul acestui algoritm, în matricea $\boldsymbol{A}$ vom avea matricea superior triunghiulară $\boldsymbol{R}$, în vectorul $\boldsymbol{b}$ vom avea $Q^T b^{\text{init}}$, b^{init} este vectorul inițial al termenilor liberi, iar matricea $\widetilde{Q}$ va conține matricea Q^T din factorizarea QR a matricei $\boldsymbol{A}$.

Algoritmul Householder detaliat

$\widetilde{Q} = I_n;$

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

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

- $\sigma = \sum_{i=r}^{n} a_{ir}^2;$
- **if** $(\sigma \leq \varepsilon)$ **break ;** // $r = r+1 \leftrightarrow P_r = I_n$ (A **singulară**)
- $k = \sqrt{\sigma};$
- **if** $(a_{rr} > 0)\ k = -k;$
- $\beta = \sigma - k\, a_{rr};$
- $u_r = a_{rr} - k;\ \ u_i = a_{ir}\ , i = r+1,\dots,n;$

$// \; A = P_r * A$

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

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

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

$$* \textbf{for } i = r, \dots, n$$

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

$//$ **transformarea coloanei** r **a matricii** A

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

$// \; b = P_r * b$

- $\gamma = (\gamma / \beta) = (b, u)/\beta = \left(\sum_{i=r}^{n} u_i b_i\right)/\beta;$
- **for** $i = r, \dots, n \quad b_i = b_i - \gamma * u_i;$

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

- **for** $j = 1, \ldots, n$

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

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

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

Numărul de operații efectuate:

A (adunări, scăderi):

$$(n-1)+2n(n-1)+\frac{n(n-1)(2n-1)}{3}=\frac{(n-1)(n+1)(2n+3)}{3}=$$

$$=\frac{2}{3}n^3+O(n^2)$$

M (înmulțiri, împărțiri):

$$4(n-1)+3n(n-1)+\frac{n(n-1)(2n-1)}{6}=\frac{2}{3}n^3+O(n^2)$$

R (radicali): ***(n-1)***