Numerikus módszerek 2.

4. előadás: Hasonlósági transzformáción alapuló SÉP megoldó algoritmusok

Krebsz Anna

ELTE IK

Tartalomjegyzék

Tartalomjegyzék

A Jacobi-módszerrel szimmetrikus mátrixok összes sajátértékét és sajátvektorát határozhatjuk meg.

Emlékeztető:

$$A = A^T \quad \Rightarrow \quad \exists\ Q \text{ ortogonális} : Q^T A\, Q = D = \operatorname{diag}(\lambda_1, \ldots, \lambda_n)$$

Ekkor Q oszlopai az A sajátvektorai.

Ötlet:

A-n ortogonális mátrixokkal (elemi forgatási mátrixokkal) végzünk hasonlósági transzformációkat: $(Q^{(k)})^T A^{(k-1)}\, Q^{(k)}$, mely a $D = \operatorname{diag}(\lambda_1, \ldots, \lambda_n)$ mátrixhoz konvergál.

Példa: forgatásra

Az alábbi mátrix az 1. és 2. tengely által meghatározott síkban az óramutató járásával ellentés φ szöggel történő forgatás mátrixa:

$$Q = \begin{bmatrix} \cos(\varphi) & -\sin(\varphi) & 0 \\ \sin(\varphi) & \cos(\varphi) & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

Ez ellenőrizhető az alábbi szorzásokkal.

$$Q \cdot e_1 = \begin{bmatrix} \cos(\varphi) \\ \sin(\varphi) \\ 0 \end{bmatrix}, \quad Q \cdot e_2 = \begin{bmatrix} -\sin(\varphi) \\ \cos(\varphi) \\ 0 \end{bmatrix}$$

$$Q \cdot e_3 = e_3$$

Algoritmus

$$A^{(0)} := A$$

$k = 1, 2, \ldots$ leállásig

$$A^{(k)} := (Q^{(k)})^T A^{(k-1)} Q^{(k)}, \text{ ahol } Q^{(k)} = Q_{(i,j)}(\varphi_k),$$

$i < j$-re $Q_{(i,j)}(\varphi_k)$-t úgy választjuk, hogy $a_{ij}^{(k)} = a_{ji}^{(k)} = 0$.

$Q_{(i,j)}(\varphi_k)$ az (i,j) pozíciókhoz tartozó $-\varphi_k$ szögű elemi forgatási mátrix. Vagyis egységmátrix, de az $(i,i), (i,j), (j,i), (j,j)$ pozíciókon a következő mátrix elemeit tartalmazza

$$\begin{bmatrix} c & s \\ -s & c \end{bmatrix}, \text{ ahol } c := \cos(\varphi_k), \ \ s := \sin(\varphi_k).$$

A módszer változatai:

Az (i, j) pozíció $(i < j)$ választásától függően:

1. **Klasszikus**: ha $|a_{ij}^{(k-1)}| = \max\{|a_{pq}^{(k-1)}| \; : p < q\}$, akkor (i, j)-t választjuk. Túl sok összehasonlítás kell hozzá, műveletigényes.
2. **Ciklikus:** az $i < j$ és $a_{ij}^{(k-1)} \neq 0$ elemeken megyünk sorban végig, majd elölről kezdjük.
3. **Küszöb:** $\varepsilon_k > 0$ előre adott nullsorozat. Ugyanúgy választunk pozíciót, mint a ciklikus változatnál, de csak olyan pozíciót, melyre $|a_{ij}^{(k-1)}| > \varepsilon_k$.

A k. lépésbeli φ_k meghatározása:
Egyszerűsítsük a jelöléseket.

- $A := A^{(k-1)}$, $B := A^{(k)}$, $Q := Q_{(i,j)}(\varphi_k) \quad \Rightarrow \quad B = Q^T A\, Q$.
- $G := Q^T A$ felhasználásával $B = G\, Q$.

Állítás:

$$\mathrm{ctg}\,(2\varphi) = \frac{a_{jj} - a_{ii}}{2a_{ij}} \;\; (0 < \varphi < \pi/2) \quad \Rightarrow \quad b_{ij} = b_{ji} = 0.$$

Biz.:

1. $G = Q^T A$ csak az i. és j. sort változtatja:

$$G = Q^T A = \begin{bmatrix} 1 & \dots & & \dots & 0 \\ & c & & -s & \\ & & \ddots & & \\ & s & & c & \\ 0 & \dots & & \dots & 1 \end{bmatrix} \cdot \begin{bmatrix} \vdots & \vdots & & \vdots \\ & a_{il} & & \\ & \vdots & & \\ & a_{jl} & & \\ \vdots & \vdots & & \vdots \end{bmatrix}$$

G i. sorának elemei: $g_{il} = c \cdot a_{il} - s \cdot a_{jl}, \ \ (l = 1, \dots, n)$
G j. sorának elemei: $g_{jl} = s \cdot a_{il} + c \cdot a_{jl}, \ \ (l = 1, \dots, n)$

Látjuk, hogy a Q^T-tal való balról szorzáshoz $6n$ műveletet kell elvégeznünk.

3. $B = GQ$ csak az i. és j. oszlopot változtatja:

$$B = GQ = \begin{bmatrix} \dots & & & & \dots \\ \dots & g_{li} & \dots & g_{lj} & \dots \\ & & & & \\ & & & & \\ \dots & & & & \dots \end{bmatrix} \cdot \begin{bmatrix} 1 & \dots & & \dots & 0 \\ & c & & s & \\ & & \ddots & & \\ & -s & & c & \\ 0 & \dots & & \dots & 1 \end{bmatrix}$$

B i. oszlopának elemei: $b_{li} = c \cdot g_{li} - s \cdot g_{lj}, \ \ (l = 1, \dots, n)$
B j. oszlopának elemei: $b_{lj} = s \cdot g_{li} + c \cdot g_{lj}, \ \ (l = 1, \dots, n)$

Látjuk, hogy a Q-val való jobbról szorzáshoz $6n$ műveletet kell elvégeznünk.

❹

$$\begin{aligned} 0 = b_{ij} &= s \cdot g_{ii} + c \cdot g_{ij} = \\ &= s \cdot (c \cdot a_{ii} - s \cdot a_{ji}) + c \cdot (c \cdot a_{ij} - s \cdot a_{jj}) = \\ &= sc \cdot a_{ii} - s^2 \cdot a_{ji} + c^2 \cdot a_{ij} - sc \cdot a_{jj} = \\ &= sc \cdot (a_{ii} - a_{jj}) + a_{ij} \cdot (c^2 - s^2) \end{aligned}$$

❺

$$2sc \cdot (a_{jj} - a_{ii}) = 2a_{ij} \cdot (c^2 - s^2)$$

❻

$$\operatorname{ctg}(2\varphi) = \frac{c^2 - s^2}{2sc} = \frac{a_{jj} - a_{ii}}{2a_{ij}} =: p$$

A $0 < \varphi < \pi/2$ feltételre φ egyértelműségéhez van szükség. $\square$

Megjegyzések:

- Csak olyan pozíciót választhatunk, ahol $a_{ij}^{(k-1)} \neq 0$.
- A k. lépésben kinullázott $a_{ij}^{(k)}$ a következő lépésben feltöltődik.
- Csak $n = 2$ esetén kapunk véges algoritmust, ilyenkor egy lépésben megkapjuk a diagonális alakot.
- Nincsenek trigonometrikus függvényhívások, φ nem kell:

$$\cos(2\varphi) = \frac{p}{\sqrt{1+p^2}}, \quad c = \sqrt{\frac{1+\cos(2\varphi)}{2}}, \quad s = \sqrt{\frac{1-\cos(2\varphi)}{2}}.$$

- Az iteráció előre haladtával $a_{ij}^{(k)} \to 0$, $\cos(2\varphi_k) \to 1$, így p-ben kicsi számmal osztunk, és s számításánál kiegyszerűsödés lép fel. Más számolási módot kell keresni.

A stabilabb számítási módot a következőkben vezetjük le:

1. $u := 2a_{ij}$, $v := a_{jj} - a_{ii}$, $w := \sqrt{u^2 + v^2}$

$$c2 := \cos(2\varphi) = \frac{\operatorname{ctg}(2\varphi)}{\sqrt{1+\operatorname{ctg}^2(2\varphi)}} = \frac{\frac{a_{jj}-a_{ii}}{2a_{ij}}}{\sqrt{1+\frac{(a_{jj}-a_{ii})^2}{4a_{ij}^2}}} = \frac{\frac{v}{u}}{\sqrt{1+\frac{v^2}{u^2}}} =$$

$$= \frac{\frac{v}{u}}{\frac{\sqrt{u^2+v^2}}{|u|}} = \frac{v \cdot \operatorname{sign}(u)}{w} \quad \to 1 \ \ (a_{ij} \to 0)$$

2.

$$c = \sqrt{\frac{1+c2}{2}} = \sqrt{\frac{1+\frac{v \cdot \operatorname{sign}(u)}{w}}{2}} = \sqrt{\frac{w + v \cdot \operatorname{sign}(u)}{2w}}.$$

3. $s = \sin(\varphi)$ képletét a kiegyszerűsödés miatt nem számíthatjuk a korábbi képletből, hanem

$$2sc \cdot v = u \cdot c2 \ \ \Rightarrow \ \ s = \frac{u}{2cv} \cdot c2 = \frac{u \cdot \operatorname{sign}(u)}{2cw}.$$

Összegezve:

A $Q^{(k)} = Q_{(i,j)}(\varphi_k)$ előállításhoz szükséges képletek:

1. $u := 2a_{ij}$, $v := a_{jj} - a_{ii}$, $w := \sqrt{u^2 + v^2}$,
2. $$c := \sqrt{\frac{w + v \cdot \operatorname{sign}(u)}{2w}}$$
3. $$s := \frac{u \cdot \operatorname{sign}(u)}{2cw}.$$

Összegezve:

Az $A^{(k)}$ előállításhoz szükséges képletek, vagyis $B = Q^T A\, Q$ számítása két lépésben.

1. $G = Q^T A$ csak az i. és j. sort változtatja:

$$g_{il} = c \cdot a_{il} - s \cdot a_{jl}$$
$$g_{jl} = s \cdot a_{il} + c \cdot a_{jl}, \quad (l = 1, \dots, n)$$

2. $B = G\, Q$ csak az i. és j. oszlopot változtatja:

$$b_{li} = c \cdot g_{li} - s \cdot g_{lj}$$
$$b_{lj} = s \cdot g_{li} + c \cdot g_{lj}, \quad (l = 1, \dots, n)$$

3. A műveletigény egy hasonlósági transzformációra $12n$.

1. Lemma:

$$\|A^{(k)}\|_F = \|A^{(k-1)}\|_F \text{ minden } k\text{-ra.}$$

2. Lemma:

Ha $A = A^T$, akkor $\|A\|_F^2 = \sum_{i=1}^n \lambda_i^2$, így minden k-ra

$$\|A^{(k)}\|_F^2 = \sum_{i=1}^n \lambda_i^2.$$

3. Lemma:

A korábbi jelölésekkel

$$b_{ii}^2 + b_{jj}^2 = a_{ii}^2 + 2a_{ij}^2 + a_{jj}^2.$$

Az 1. és 2. Lemma bizonyítása volt előző félévben gyakorlaton.

3. Lemma biz.:

1. Az ortogonális transzformáció miatt $\|G\, i.\text{oszl}\|_2^2 = \|A\, i.\text{oszl}\|_2^2$, de az oszlopokban csak az i. és j. pozíció változik:

$$i.\ \text{oszlopra:}\ g_{ii}^2 + g_{ji}^2 = a_{ii}^2 + a_{ji}^2,$$

$$j.\ \text{oszlopra:}\ g_{ij}^2 + g_{jj}^2 = a_{ij}^2 + a_{jj}^2.$$

2. Hasonlóan $\|B\, i.\text{sora}\|_2^2 = \|G\, i.\text{sora}\|_2^2$, de a sorokban csak az i. és j. pozíció változik:

$$i.\ \text{sorra:}\ b_{ii}^2 + b_{ij}^2 = g_{ii}^2 + g_{ij}^2,$$

$$j.\ \text{sorra:}\ b_{ji}^2 + b_{jj}^2 = g_{ji}^2 + g_{jj}^2.$$

3. Összegezve:

$$b_{ii}^2 + \underbrace{2b_{ij}^2}_{=0} + b_{jj}^2 = (g_{ii}^2 + g_{ij}^2) + (g_{ji}^2 + g_{jj}^2) = a_{ii}^2 + 2a_{ij}^2 + a_{jj}^2$$

$\square$

Tétel: a Jacobi-módszer konvergencia tétele

A klasszikus Jacobi-módszerrel generált $(A^{(k)})$ sorozat olyan diagonális mátrixhoz konvergál, melynek átlójában az A sajátértékei állnak.

Biz.: Vezessük be a következő jelölést a főátlón kívüli elemek négyzetösszegére:

$$N(A) := \sum_{i=1}^{n} \sum_{j=1, j\neq i}^{n} a_{ij}^2 = \|A\|_F^2 - \sum_{l=1}^{n} a_{ll}^2.$$

A következő lépésekben bizonyítjuk a tételt:

1. Belátjuk, hogy $N(A^{(k)}) = N(A^{(k-1)}) - 2 \cdot (a_{ij}^{(k-1)})^2$.
2. Majd $N(A^{(k)}) \leq N(A) \left(1 - \frac{2}{n(n-1)}\right)^k$, $n > 2$-re.
3. Gersgorin-tétellel belátjuk, hogy az átlóbeli elemek a sajátértékekhez konvergálnak.

Biz. folyt.:

1. Belátjuk, hogy $N(A^{(k)}) = N(A^{(k-1)}) - 2 \cdot (a_{ij}^{(k-1)})^2$.

Felhasználva az 1. és 3. Lemmát, továbbá azt, hogy a módszer egy lépése során csak az i. és j. sorok és oszlopok változnak:

$$\begin{aligned}
N(A^{(k-1)}) - N(A^{(k)}) &= \\
&= \left(\|A^{(k-1)}\|_F^2 - \sum_{l=1}^{n} (a_{ll}^{(k-1)})^2 \right) - \left(\|A^{(k)}\|_F^2 - \sum_{l=1}^{n} (a_{ll}^{(k)})^2 \right) = \\
&= \sum_{l=1}^{n} (a_{ll}^{(k)})^2 - \sum_{l=1}^{n} (a_{ll}^{(k-1)})^2 = \\
&= (a_{ii}^{(k)})^2 + (a_{jj}^{(k)})^2 - (a_{ii}^{(k-1)})^2 - (a_{jj}^{(k-1)})^2 = \\
&= 2 \cdot (a_{ij}^{(k-1)})^2
\end{aligned}$$

Biz. folyt.:

2. A klasszikus Jacobi-módszer esetén $|a_{ij}^{(k-1)}|$ a maximális abszolút értékű elem $A^{(k-1)}$-ben, így

$$N(A^{(k-1)}) \leq n(n-1) \cdot |a_{ij}^{(k-1)}|^2 \quad \Rightarrow \quad |a_{ij}^{(k-1)}|^2 \geq \frac{1}{n(n-1)} N(A^{(k-1)}).$$

Írjuk fel $N(A^{(k)})$-ra a rekurziót

$$\begin{aligned} N(A^{(k)}) &= N(A^{(k-1)}) - 2(a_{ij}^{(k-1)})^2 \leq \\ &\leq N(A^{(k-1)}) - \frac{2}{n(n-1)} N(A^{(k-1)}) = \\ &= N(A^{(k-1)}) \left(1 - \frac{2}{n(n-1)}\right). \end{aligned}$$

Biz. folyt.: Kibontva a rekurziót

$$N(A^{(k)}) \leq N(A)\left(1 - \frac{2}{n(n-1)}\right)^k \ \rightarrow 0 \ \ (k \to \infty) \quad (n > 2).$$

Mivel $\left(1 - \frac{2}{n(n-1)}\right) < 1$, az $(A^{(k)})$ sorozat diagonális mátrixhoz konvergál.

3. Az 1. és 2. Lemmából

$$\sum_{i=1}^{n} \lambda_i^2 = \|A\|_F^2 = \|A^{(k)}\|_F^2 = \lim_{k\to\infty} \sum_{l=1}^{n} (a_{ll}^{(k)})^2.$$

Mivel a Gersgorin-körök sugarai 0-hoz tartanak, így $A^{(k)}$ állóbeli elemei a sajátértékekhez konvergálnak. $\square$

Példa: A Jacobi módszer alkalmazása

Végezzük el a következő mátrixon a Jacobi-módszer egy lépését az $(i, j) = (1, 2)$ pozíciónak megfelelően!

$$A = \begin{bmatrix} -1 & 4 & 0 \\ 4 & 5 & 0 \\ 0 & 0 & 3 \end{bmatrix}$$

Mivel csak két átlón kívüli elem nem nulla, ezért a Jacobi-módszer egy lépésben előállítja a diagonális alakot. Ezzel mindhárom sajátértéket és a hozzájuk tartozó sajátvektorokat is megkapjuk.

Írjuk fel az $A^{(1)} = Q^T A Q$ mátrixot, ahol $c := \cos\varphi, \quad s := \sin\varphi$.

$$A^{(1)} = \begin{bmatrix} c & -s & 0 \\ s & c & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} -1 & 4 & 0 \\ 4 & 5 & 0 \\ 0 & 0 & 3 \end{bmatrix} \cdot \begin{bmatrix} c & s & 0 \\ -s & c & 0 \\ 0 & 0 & 1 \end{bmatrix} =$$

$$= \begin{bmatrix} -c-4s & 4c-5s & 0 \\ -s+4c & 4s+5c & 0 \\ 0 & 0 & 3 \end{bmatrix} \cdot \begin{bmatrix} c & s & 0 \\ -s & c & 0 \\ 0 & 0 & 1 \end{bmatrix} =$$

$$= \begin{bmatrix} -c^2-4sc-4sc+5s^2 & -sc-4s^2+4c^2-5sc & 0 \\ -sc+4c^2-4s^2-5sc & -s^2+4sc+4sc+5c^2 & 0 \\ 0 & 0 & 3 \end{bmatrix} =$$

$$= \begin{bmatrix} -c^2+5s^2-8sc & 4c^2-4s^2-6sc & 0 \\ 4c^2-4s^2-6sc & -s^2+5c^2+8sc & 0 \\ 0 & 0 & 3 \end{bmatrix}$$

Kézi számolásnál alkalmazhatjuk az egyszerűbb nem stabil képleteket.

$$p := \cot 2\varphi = \frac{3}{4}$$

$$\cos 2\varphi = \frac{p}{\sqrt{1+p^2}} = \frac{\frac{3}{4}}{\sqrt{1+\frac{9}{16}}} = \frac{3}{4} \cdot \frac{4}{5} = \frac{3}{5}$$

$$c := \cos \varphi = \sqrt{\frac{1+\cos 2\varphi}{2}} = \frac{2}{\sqrt{5}} \quad \to c^2 = \frac{4}{5}$$

$$s := \sin \varphi = \sqrt{\frac{1-\cos 2\varphi}{2}} = \frac{1}{\sqrt{5}} \quad \to s^2 = \frac{1}{5}$$

Példa a Jacobi-módszerre

A Q transzformációs mátrix oszlopai az A sajátvektorai.

$$Q = \frac{1}{\sqrt{5}} \cdot \begin{bmatrix} 2 & 1 & 0 \\ -1 & 2 & 0 \\ 0 & 0 & \sqrt{5} \end{bmatrix}$$

A kapott értékeket az $A^{(1)}$ mátrixba behelyettesítve

$$-c^2 + 5s^2 - 8sc = -\frac{4}{5} + \frac{5}{5} + 8 \cdot \frac{2}{\sqrt{5}} \cdot \frac{1}{\sqrt{5}} = -3$$

$$-s^2 + 5c^2 + 8sc = -\frac{1}{5} + \frac{20}{5} + \frac{16}{5} = 7$$

A hasonlósági transzformáció után kapott mátrix a következő.

$$A^{(1)} = \begin{bmatrix} -3 & 0 & 0 \\ 0 & 7 & 0 \\ 0 & 0 & 3 \end{bmatrix}$$

Ellenőrizzük a 3. Lemma állítását a transzformációs lépésre.

$$b_{ii}^2 + b_{jj}^2 = a_{ii}^2 + 2a_{ij}^2 + a_{jj}^2 \quad (i < j)$$

$$A = \begin{bmatrix} -1 & 4 & 0 \\ 4 & 5 & 0 \\ 0 & 0 & 3 \end{bmatrix}, \quad B = A^{(1)} = Q^T A Q = \begin{bmatrix} -3 & 0 & 0 \\ 0 & 7 & 0 \\ 0 & 0 & 3 \end{bmatrix}$$

$$\begin{aligned} a_{11}^2 + 2a_{12}^2 + a_{22}^2 &= (-1)^2 + 5^2 + 2 \cdot 4^2 = 1 + 25 + 32 = 58 \\ b_{11}^2 + b_{22}^2 &= (-3)^2 + 7^2 = 58 \end{aligned}$$

$\square$

Tartalomjegyzék

Ötlet:

1. Készítsük el az A mátrix LU-felbontását: $A = LU$.
2. A kapott mátrixokból állítsuk elő $B := UL$-t.
3. Ekkor A és B hasonlók ($B = L^{-1} A\, L$), a sajátértékeik azonosak.

Algoritmus: LU-algoritmus

1. $A_1 := A$, $k = 1, 2, \ldots$, leállásig:
2. $A_k = L_k \cdot U_k$, LU-felbontás előállítása
3. $A_{k+1} := U_k \cdot L_k$.

Tétel: LU-algoritmus

Az LU-algoritmussal generált (A_k) sorozatra

1. $$A_{k+1} = L_k^{-1} A_k L_k,$$

2. $$A_{k+1} = \widetilde{L}_k^{-1} A \widetilde{L}_k = \widetilde{U}_k A \widetilde{U}_k^{-1}$$
   $$\text{ahol } \widetilde{L}_k = L_1 L_2 \ldots L_k \text{ és } \widetilde{U}_k = U_k U_{k-1} \ldots U_1$$

3. $$A^{k+1} = \widetilde{L}_k \widetilde{U}_k.$$

Biz.: Indukcióval. (Hf.)

Tétel: LU-algoritmus konvergencia tétele

1. Tegyük fel, hogy az A sajátértékeire $|\lambda_1| > |\lambda_2| > \ldots > |\lambda_n| > 0$, továbbá
2. A diagonalizálható, azaz $A = SDS^{-1}$.
3. S-nek és S^{-1}-nek létezik LU-felbontása.

Ekkor az LU-algoritmussal generált (A_k) sorozat felső háromszögmátrixhoz konvergál, melynek átlójában a sajátértékek vannak.

Nem bizonyítjuk.

Ötlet:

1. Készítsük el az A mátrix QR-felbontását: $A = QR$.
2. A kapott mátrixokból állítsuk elő $B := RQ$-t.
3. Ekkor A és B hasonlók ($B = Q^T AQ$), a sajátértékeik azonosak.

Algoritmus: QR-algoritmus

1. $A_1 := A$, $k = 1, 2, \ldots$, leállásig:
2. $A_k = Q_k \cdot R_k$, QR-felbontást állítsuk elő
3. $A_{k+1} := R_k \cdot Q_k$.

Tétel: QR-algoritmus

Az QR-algoritmussal generált (A_k) sorozatra

1. $$A_{k+1} = Q_k^T A_k Q_k,$$

2. $$A_{k+1} = \widetilde{Q}_k^T A \widetilde{Q}_k = \widetilde{R}_k A \widetilde{R}_k^{-1}$$
   ahol $\widetilde{Q}_k = Q_1 Q_2 \dots Q_k$ és $\widetilde{R}_k = R_k R_{k-1} \dots R_1$

3. $$A^{k+1} = \widetilde{Q}_k \widetilde{R}_k.$$

Biz.: Indukcióval. (Hf.)

Tétel: QR-algoritmus konvergencia tétele

1. Tegyük fel, hogy az A sajátértékeire $|\lambda_1| > |\lambda_2| > \ldots > |\lambda_n| > 0$,
2. továbbá A diagonalizálható, azaz $A = SDS^{-1}$.
3. S^{-1}-nek létezik LU-felbontása.

Ekkor az QR-algoritmussal generált (A_k) sorozat felső háromszögmátrixhoz konvergál, melynek átlójában a sajátértékek vannak.

Nem bizonyítjuk.

Megjegyzések:

- A szimmetrikus mátrixra a QR-felbontás rekurziója invariáns.

  $$A_{k-1}^T = A_{k-1} \;\; \Rightarrow \;\; A_k^T = \left(Q_k^T A_{k-1} Q_k\right)^T = Q_k^T A_{k-1} Q_k = A_k$$

  Ekkor az (A_k) sorozat diagonális mátrix-hoz konvergál. A hibabecsléshez Gergorin-tétel használható.
- Telt mátrix esetén a QR-felbontás műveletigénye túl nagy ($O(n^3)$), ezért célszerű előbb felső Hessenberg-alakra hozni.
- Az algoritmus a Hessenberg-alakot megtartja. (Beadható Hf.)
- Felső Hessenberg-alakra a QR-felbontást $n-1$ db 2×2-es Householder-transzformációval $O(n)$ művelettel lehet megoldani.
- Szimmetrikus esetben a felső Hessenberg-alak tridiagonális alakot jelent.

Ötlet: Legyen H irreducibilis felső Hessenberg-alak. Ha λ sajátértéke H-nak, akkor $\det(H - \lambda I) = 0$ miatt a $H - \lambda I = QR$ felbontásában $r_{nn} = 0$, így R utolsó sora 0.

$$B := RQ + \lambda I\text{-t}$$

előállítva RQ utolsó sora is 0, így $b_{nn} = \lambda$ lesz. Vagyis megkaptunk egy sajátértéket. Nyilván a gyakorlatban így nem használható, mert nem ismerjük a pontos sajátértéket.

QR-algoritmus σ_k eltolással

1. Legyen H_1 az A-hoz hasonló felső Hessenberg-alak, $k = 1, 2, \ldots,$ leállásig:
2. Válasszunk egy σ_k értéket (shiftet),
3. készítsük a $H_k - \sigma_k I = Q_k R_k$ QR-felbontást,
4. állítsuk elő a $H_{k+1} := R_k Q_k + \sigma_k I$ mátrixot.

Állítás:

H_k és H_{k+1} hasonló.

Biz.:

$$\begin{aligned} H_{k+1} &= R_k Q_k + \sigma_k I = Q_k^T Q_k R_k Q_k + Q_k^T \sigma_k Q_k = \\ &= Q_k^T (Q_k R_k + \sigma_k I) Q_k = Q_k^T H_k Q_k. \end{aligned}$$

Megjegyzések:

- Ha valamely i-re $h_{i,i-1}^{(k)} = 0$, akkor a feladat felbomlik két kisebb méretűre.
- Eltolási paraméternek $\sigma_k := h_{nn}^{(k)}$-t javasolják, ezt Rayleigh-shiftelésnek nevezik.
- A fenti eltolás nem mindig működik (pl. többszörös sajátértékek esetén), ilyenkor a jobb alsó 2×2-es részmátrix azon sajátértékét használjuk σ_k-nak, amelyik közelebb van $h_{nn}^{(k)}$-koz. Ezt nevezik Wilkinson-shiftelésnek.

- Előfordulhat, hogy a 2×2-es jobb alsó blokk sajátértékei komplexek, ezért komplex eltolást kéne alkalmazni. Ilyenkor két lépést lehet egybefogni, ezzel valós aritmetikában végrehajtva a dupla lépéses implicit QR-algoritmust kapjuk.
- Ha a mátrixnak vannak komplex sajátértékei, akkor 2×2-es blokkok maradnak az átlóban.
- A QR-algoritmus konvergenciája a $|\lambda_{i+1}|/|\lambda_i|$ hányadostól függ. Ezt lehet az eltolással javítani. Eltolás esetén másodrendű a konvergencia sebessége. Ha még szimmetrikus is a mátrix, akkor harmadrendű a módszer.

Köszönöm a figyelmet!