

7. Simetrični problem lastnih vrednosti Relativno robustne reprezentacije

Bor Plestenjak

NLA

5. april 2011

Imamo simetrično matriko A , za katero bi radi izračunali vse lastne pare.

Imamo simetrično matriko A , za katero bi radi izračunali vse lastne pare.

Računanje običajno poteka v treh fazah:

- 1) Redukcija A na tridiagonalno matriko T .
- 2) Računanje lastnih parov T .
- 3) Transformacija lastnih vektorjev T v lastne vektorje A .

Imamo simetrično matriko A , za katero bi radi izračunali vse lastne pare.

Računanje običajno poteka v treh fazah:

- 1) Redukcija A na tridiagonalno matriko T .
- 2) Računanje lastnih parov T .
- 3) Transformacija lastnih vektorjev T v lastne vektorje A .

Časovne zahtevnosti:

- 1) $\mathcal{O}(n^3)$
- 2) Odvisno od metode:
 - Vse preko QR: $\mathcal{O}(n^3)$
 - Lastne vrednosti preko QR, lastni vektorji preko inverzne iteracije: $\mathcal{O}(n^2)$, kadar je potrebna naknadna ortogonalizacija $\mathcal{O}(n^3)$.
 - Deli in vladaj: $\mathcal{O}(n^3)$, če so lastne vrednosti enakomerno razporejene
- 3) $\mathcal{O}(n^3)$

Izračun lastnega vektorja

$$T = \begin{bmatrix} a_1 & b_1 & & \\ b_1 & \ddots & \ddots & \\ & \ddots & \ddots & b_{n-1} \\ & & b_{n-1} & a_n \end{bmatrix}.$$

Naj bo λ lastna vrednost ireducibilne simetrične tridiagonalne matrike T . Za lastni vektor z velja $z_1 \neq 0$ in $z_n \neq 0$.

Izračun lastnega vektorja

$$T = \begin{bmatrix} a_1 & b_1 & & \\ b_1 & \ddots & \ddots & \\ & \ddots & \ddots & b_{n-1} \\ & & b_{n-1} & a_n \end{bmatrix}.$$

Naj bo λ lastna vrednost ireducibilne simetrične tridiagonalne matrike T . Za lastni vektor z velja $z_1 \neq 0$ in $z_n \neq 0$.

$$z_1 = 1$$

$$z_2 = -(a_1 - \lambda)/b_1$$

$$k = 2, \dots, n-1$$

$$z_{k+1} = -(b_{k-1}z_{k-1} + (a_k - \lambda)z_k)/b_k$$

Izračun lastnega vektorja

$$T = \begin{bmatrix} a_1 & b_1 & & \\ b_1 & \ddots & \ddots & \\ & \ddots & \ddots & b_{n-1} \\ & & b_{n-1} & a_n \end{bmatrix}.$$

Naj bo λ lastna vrednost ireducibilne simetrične tridiagonalne matrike T . Za lastni vektor z velja $z_1 \neq 0$ in $z_n \neq 0$.

$$z_1 = 1$$

$$z_2 = -(a_1 - \lambda)/b_1$$

$$k = 2, \dots, n-1$$

$$z_{k+1} = -(b_{k-1}z_{k-1} + (a_k - \lambda)z_k)/b_k$$

Če fiksiramo $z_1 = 1$, potem iz prve do predzadnje enačbe sistema $(T - \lambda I)z = 0$ dobimo lastni vektor z . Pri tem smo opustili zadnjo enačbo.

Izračun lastnega vektorja

$$T = \begin{bmatrix} a_1 & b_1 & & & \\ b_1 & \ddots & \ddots & & \\ & \ddots & \ddots & b_{n-1} & \\ & & b_{n-1} & a_n & \end{bmatrix}.$$

Naj bo λ lastna vrednost ireducibilne simetrične tridiagonalne matrike T . Za lastni vektor z velja $z_1 \neq 0$ in $z_n \neq 0$.

$$z_1 = 1$$

$$z_2 = -(a_1 - \lambda)/b_1$$

$$k = 2, \dots, n-1$$

$$z_{k+1} = -(b_{k-1}z_{k-1} + (a_k - \lambda)z_k)/b_k$$

Če fiksiramo $z_1 = 1$, potem iz prve do predzadnje enačbe sistema $(T - \lambda I)z = 0$ dobimo lastni vektor z . Pri tem smo opustili zadnjo enačbo.

Če je $\tilde{\lambda}$ približek za λ , potem za z velja $(T - \tilde{\lambda}I)z = \delta_n e_n$ za nek δ_n .

Numerično računanje lastnega vektorja

Podobno, če bi opustili k -to enačbo, bi za ostanek veljalo $(T - \tilde{\lambda}I)z = \delta_k e_k$.

Opustitev k -te enačbe je ekvivalentna temu, da kot približek za lastni vektor vzamemo $z = (T - \tilde{\lambda}I)^{-1}e_k$.

Naj bo $\tilde{\lambda}$ približek za λ_j . Katero enačbo naj opustimo?

Numerično računanje lastnega vektorja

Podobno, če bi opustili k -to enačbo, bi za ostanek veljalo $(T - \tilde{\lambda}I)z = \delta_k e_k$.

Opustitev k -te enačbe je ekvivalentna temu, da kot približek za lastni vektor vzamemo $z = (T - \tilde{\lambda}I)^{-1}e_k$.

Naj bo $\tilde{\lambda}$ približek za λ_j . Katero enačbo naj opustimo?

Iz $e_k = \sum_{i=1}^n (x_i)_k x_i$ sledi

$$z^{(k)} = (T - \tilde{\lambda}I)^{-1}e_k = \frac{(x_j)_k}{\lambda_j - \tilde{\lambda}} \left(x_j + \sum_{i \neq j} \frac{(x_i)_k}{(x_j)_k} \cdot \frac{\lambda_j - \tilde{\lambda}}{\lambda_i - \tilde{\lambda}} x_i \right)$$

Numerično računanje lastnega vektorja

Podobno, če bi opustili k -to enačbo, bi za ostanek veljalo $(T - \tilde{\lambda}I)z = \delta_k e_k$.

Opustitev k -te enačbe je ekvivalentna temu, da kot približek za lastni vektor vzamemo $z = (T - \tilde{\lambda}I)^{-1}e_k$.

Naj bo $\tilde{\lambda}$ približek za λ_j . Katero enačbo naj opustimo?

Iz $e_k = \sum_{i=1}^n (x_i)_k x_i$ sledi

$$z^{(k)} = (T - \tilde{\lambda}I)^{-1}e_k = \frac{(x_j)_k}{\lambda_j - \tilde{\lambda}} \left(x_j + \sum_{i \neq j} \frac{(x_i)_k}{(x_j)_k} \cdot \frac{\lambda_j - \tilde{\lambda}}{\lambda_i - \tilde{\lambda}} x_i \right)$$

Vidimo, da bo $z^{(k)}$ dober približek za lastni vektor, če

- opustimo k -to enačbo, kjer ima lastni vektor x_j maksimalno komponento,
- velja $|\lambda_j - \tilde{\lambda}| \ll |\lambda_i - \tilde{\lambda}|$ za $i \neq j$.

LDL^T in UDU^T faktorizaciji

$$T = \begin{bmatrix} a_1 & b_1 & & \\ b_1 & \ddots & \ddots & \\ & \ddots & \ddots & b_{n-1} \\ & & b_{n-1} & a_n \end{bmatrix} = LD^+L^T = UD^-U^T$$

$$L = \begin{bmatrix} 1 & & & \\ l_1 & 1 & & \\ & \ddots & \ddots & \\ & & l_{n-1} & 1 \end{bmatrix}, D^+ = \begin{bmatrix} d_1^+ & & & \\ & d_2^+ & & \\ & & \ddots & \\ & & & d_n^+ \end{bmatrix}$$

$$U = \begin{bmatrix} 1 & u_1 & & \\ & \ddots & \ddots & \\ & & 1 & u_{n-1} \\ & & & 1 \end{bmatrix}, D^- = \begin{bmatrix} d_1^- & & & \\ & d_2^- & & \\ & & \ddots & \\ & & & d_n^- \end{bmatrix}$$

LDL^T in UDU^T faktorizaciji

$$T = \begin{bmatrix} a_1 & b_1 & & \\ b_1 & \ddots & \ddots & \\ & \ddots & \ddots & b_{n-1} \\ & & b_{n-1} & a_n \end{bmatrix} = LD^+L^T = UD^-U^T$$

$$L = \begin{bmatrix} 1 & & & \\ l_1 & 1 & & \\ & \ddots & \ddots & \\ & & l_{n-1} & 1 \end{bmatrix}, D^+ = \begin{bmatrix} d_1^+ & & & \\ & d_2^+ & & \\ & & \ddots & \\ & & & d_n^+ \end{bmatrix}$$

$$LD^+L^T$$

$$d_1^+ = a_1$$

$$k = 1, \dots, n-1$$

$$l_k = b_k / d_k^+$$

$$d_{k+1}^+ = a_{k+1} - l_k b_k$$

$$U = \begin{bmatrix} 1 & u_1 & & \\ & \ddots & \ddots & \\ & & 1 & u_{n-1} \\ & & & 1 \end{bmatrix}, D^- = \begin{bmatrix} d_1^- & & & \\ & d_2^- & & \\ & & \ddots & \\ & & & d_n^- \end{bmatrix}$$

$$UD^-U^T$$

$$d_n^- = a_n$$

$$k = n-1, \dots, 1$$

$$u_k = b_k / d_{k+1}^-$$

$$d_k^- = a_k - u_k b_k$$

Zasukan razcep

Na k zgornjih vrsticah naredimo razcep LD^+L^T , na spodnjih $n - k + 1$ pa UD^-U^T . Dobimo $T = N_k D_k N_k^T$, kjer je

$$N_k = \begin{bmatrix} 1 & & & & & & & & \\ & l_1 & & \ddots & & & & & \\ & & \ddots & & & & & & \\ & & & \ddots & & & & & \\ & & & & 1 & & & & \\ & & & & l_{k-1} & & 1 & & u_k \\ & & & & & & & 1 & \ddots \\ & & & & & & & & \ddots & u_{n-1} \\ & & & & & & & & & 1 \end{bmatrix}$$

in $D_k = \text{diag}(d_1^+, \dots, d_{k-1}^+, \gamma_k, d_{k+1}^-, \dots, d_n^-)$.

Zasukan razcep

Na k zgornjih vrsticah naredimo razcep LD^+L^T , na spodnjih $n - k + 1$ pa UD^-U^T . Dobimo $T = N_k D_k N_k^T$, kjer je

$$N_k = \begin{bmatrix} 1 & & & & & & \\ & l_1 & & & & & \\ & & \ddots & & & & \\ & & & \ddots & & & \\ & & & & 1 & & \\ & & & & l_{k-1} & 1 & u_k \\ & & & & & & 1 \\ & & & & & & & \ddots \\ & & & & & & & & \ddots \\ & & & & & & & & & u_{n-1} \\ & & & & & & & & & & 1 \end{bmatrix}$$

in $D_k = \text{diag}(d_1^+, \dots, d_{k-1}^+, \gamma_k, d_{k+1}^-, \dots, d_n^-)$.

Izrek

- 1) $\gamma_k = d_k^+ + d_k^- - a_k$
- 2) Če je matrika T nesingularna, potem velja $1/\gamma_k = e_k^T T^{-1} e_k$.

Lema

Naj bo $N_k D_k N_k^T$ zasukani razcep matrike $T - \tilde{\lambda} I$. Velja

$$\frac{1}{\gamma_k} = \frac{[(x_j)_k]^2}{\lambda_j - \tilde{\lambda}} + \sum_{i \neq j} \frac{[(x_i)_k]^2}{\lambda_i - \tilde{\lambda}}.$$

Lema

Naj bo $N_k D_k N_k^T$ zasukani razcep matrike $T - \tilde{\lambda} I$. Velja

$$\frac{1}{\gamma_k} = \frac{[(x_j)_k]^2}{\lambda_j - \tilde{\lambda}} + \sum_{i \neq j} \frac{[(x_i)_k]^2}{\lambda_i - \tilde{\lambda}}.$$

Naj bo $|\lambda_j - \tilde{\lambda}| \ll |\lambda_i - \tilde{\lambda}|$ za $i \neq j$. Potem:

majhna vrednost $|\gamma_k| \Leftrightarrow$ nadpovprečna vrednost $|(x_j)_k|$

Iz zasukanega razcepa ocenimo, kateri element lastnega vektorja je maksimalen.

Izračun lastnega vektorja iz zasukanega razcepa

$$N_k = \begin{bmatrix} 1 & & & & & & \\ & h_1 & & & & & \\ & & \ddots & & & & \\ & & & \ddots & & & \\ & & & & 1 & & \\ & & & l_{k-1} & 1 & u_k & \\ & & & & & 1 & \ddots \\ & & & & & & \ddots & u_{n-1} \\ & & & & & & & 1 \end{bmatrix},$$

$$D_k = \text{diag}(d_1^+, \dots, d_{k-1}^+, \gamma_k, d_{k+1}^-, \dots, d_n^-)$$

Če je $T - \tilde{\lambda}I = N_k D_k N_k^T$, ima sistem $(T - \tilde{\lambda}I)z = \gamma_k e_k$ enolično rešitev, ki jo dobimo z:

$$z_k = 1$$

$$j = k-1, \dots, 1$$

$$z_j = -l_j z_{j+1}$$

$$j = k+1, \dots, n$$

$$z_j = -u_{j-1} z_{j-1}$$

Izračun lastnega vektorja iz zasukanega razcepa

$$N_k = \begin{bmatrix} 1 & & & & & & \\ & h_1 & & & & & \\ & & \ddots & & & & \\ & & & \ddots & & & \\ & & & & 1 & & \\ & & & l_{k-1} & 1 & u_k & \\ & & & & & 1 & \ddots \\ & & & & & & \ddots & u_{n-1} \\ & & & & & & & 1 \end{bmatrix},$$

$$D_k = \text{diag}(d_1^+, \dots, d_{k-1}^+, \gamma_k, d_{k+1}^-, \dots, d_n^-)$$

Če je $T - \tilde{\lambda}I = N_k D_k N_k^T$, ima sistem $(T - \tilde{\lambda}I)z = \gamma_k e_k$ enolično rešitev, ki jo dobimo z:

$$z_k = 1$$

$$j = k-1, \dots, 1$$

$$z_j = -l_j z_{j+1} \quad (\text{v primeru } z_{j+1} = 0 \text{ pa } z_j = -b_{j+1} z_{j+2} / b_j)$$

$$j = k+1, \dots, n$$

$$z_j = -u_{j-1} z_{j-1} \quad (\text{v primeru } z_{j-1} = 0 \text{ pa } z_j = -b_{j-2} z_{j-2} / b_{j-1})$$

Ponovitev algoritma

Naj bo $\tilde{\lambda}$ približek za lastno vrednost matrike T . Lastni vektor z izračunamo po naslednjem postopku.

Algoritem za izračun lastnega vektorja

- 1) Izračunaj razcepa $T - \tilde{\lambda}I = LD^+L^T$ in $T - \tilde{\lambda}I = UD^-U^T$
- 2) Izračunaj $\gamma_1, \dots, \gamma_n$ za zasukane razcepe in poišči minimalni $|\gamma_k|$
- 3) Reši sistem $(T - \tilde{\lambda}I)z = \gamma_k e_k$ in normiraj z

Zahtevnost tega algoritma je $12n + \mathcal{O}(1)$ operacij.

Če lahko vnaprej ocenimo k (npr. z Gerschgorinovimi krogi), se zahtevnost še zmanjša.

Relativno robustna reprezentacija matrike

Tridiagonalno matriko T lahko predstavimo z vektorjema diagonalnih in obdiagonalnih elementov a in b . Majhne relativne spremembe a in b lahko povzročijo velike relativne spremembe lastnih vrednosti in vektorjev.

Če je T pozitivno definitna, jo lahko predstavimo z vektorjema diagonalnih in obdiagonalnih elementov faktorja Choleskega. Majhne relativne spremembe teh elementov povzročijo majhne relativne spremembe lastnih parov, zato je takšna predstavitev relativno robustna.

Relativno robustna reprezentacija matrike

Tridiagonalno matriko T lahko predstavimo z vektorjema diagonalnih in obdiagonalnih elementov a in b . Majhne relativne spremembe a in b lahko povzročijo velike relativne spremembe lastnih vrednosti in vektorjev.

Če je T pozitivno definitna, jo lahko predstavimo z vektorjema diagonalnih in obdiagonalnih elementov faktorja Choleskega. Majhne relativne spremembe teh elementov povzročijo majhne relativne spremembe lastnih parov, zato je takšna predstavitev relativno robustna.

Definicija

Množica števil $\{p_i\}$, ki določa matriko T , je **relativno robustna reprezentacija (RRR)**, če se lastni pari matrike $T + \delta T$, ki jo določa množica zmotenih elementov $\{p_i(1 + \epsilon_i)\}$, relativno malo razlikujejo od lastnih parov matrike T .

Če reprezentacija z visoko relativno natančnostjo določa le lastne vrednosti $(\lambda_j, \dots, \lambda_k)$, pravimo da je delna $RRR(j, \dots, k)$.

Če je T pozitivno definitna, lahko namesto razcepa Choleskega uporabimo LDL^T razcep (izognemo se kvadratnim korenem), ki je prav tako RRR.

Če je T nedefinitna, lahko uporabimo razcep $T - \sigma I = LDL^T$ za izbrani premik σ .

Ob primerno izbranem σ dosežemo, da je zgornji razcep delna RRR za iskane lastne vrednosti.

Medtem ko lastne vrednosti lahko izračunamo z visoko relativno natančnostjo npr. z bisekcijo, pa za natančne lastne vektorje potrebujemo, da je lastna vrednost relativno dobro izolirana, kar lahko dosežemo s premiki.

RRR premaknjene matrike

Denimo, da poznamo RRR matrike $T + \tau I$ oblike LDL^T .

Potrebujemo $L^+ D^+ L^{+T}$ in $U^- D^- U^{-T}$ razcepa premaknjene matrike $T + \tau I - \mu I = LDL^T - \mu I$.

RRR premaknjene matrike

Denimo, da poznamo RRR matrike $T + \tau I$ oblike LDL^T .

Potrebujemo $L^+ D^+ L^{+T}$ in $U^- D^- U^{-T}$ razcepa premaknjene matrike $T + \tau I - \mu I = LDL^T - \mu I$.

Z naslednjimi algoritmi ohranimo relativno robustno reprezentacijo.

dstqds algoritem

$$\begin{aligned} s_1 &= -\mu \\ i &= 1, \dots, k-1 \\ d_i^+ &= s_i + d_i \\ l_i^+ &= l_i d_i / d_i^+ \\ s_{i+1} &= l_i^+ l_i s_i - \mu \\ d_n^+ &= s_n + d_n \end{aligned}$$

dqds algoritem

$$\begin{aligned} p_n &= d_n - \mu \\ i &= n-1, \dots, 1 \\ d_{i+1}^- &= d_i l_i^2 + p_{i+1} \\ t &= d_i / d_{i+1}^- \\ p_i &= p_{i+1} i - \mu \\ d_1^- &= p_1 \end{aligned}$$

izračun γ

$$\begin{aligned} i &= 1, \dots, n \\ \gamma_i &= s_i + p_i + \mu \end{aligned}$$

Naj bo $\tilde{\lambda}$ približek za lastno vrednost matrike LDL^T , ki je RRR.

Algoritem GetVec

- 1) Preko dstqds izračunaj razcep $LDL^T - \tilde{\lambda}I = L^+ D^+ L^{+T}$
- 2) Preko dqds izračunaj razcep $LDL^T - \tilde{\lambda}I = U^- D^- U^{-T}$
- 3) Izračunaj $\gamma_1, \dots, \gamma_n$ za zasukane razcepe in poišči minimalni $|\gamma_k|$
- 4) Reši sistem $N_k D_k N_k^T z = \gamma_k e_k$ in normiraj z

Naj bo $\tilde{\lambda}$ približek za lastno vrednost λ_j matrike $T = LDL^T$.

Za lastni vektor izračunan preko inverzne iteracije, velja

$$|\sin \angle(z, x_j)| \leq \frac{\mathcal{O}(nu \|T\|)}{\text{gap}(\tilde{\lambda})},$$

kjer je $\text{gap}(\tilde{\lambda}) = \min_{i \neq j} |\tilde{\lambda} - \lambda_i|$,

Za lastni vektor, izračunan preko RRR, pa velja

$$|\sin \angle(z, x_j)| \leq \frac{\mathcal{O}(nu)}{\text{relgap}(\tilde{\lambda})},$$

kjer je $\text{relgap}(\tilde{\lambda}) = \frac{\text{gap}(\tilde{\lambda})}{|\tilde{\lambda}|}$.

Naj bo $\tilde{\lambda}$ približek za lastno vrednost λ_j matrike $T = LDL^T$.

Za lastni vektor izračunan preko inverzne iteracije, velja

$$|\sin \angle(z, x_j)| \leq \frac{\mathcal{O}(nu \|T\|)}{\text{gap}(\tilde{\lambda})},$$

kjer je $\text{gap}(\tilde{\lambda}) = \min_{i \neq j} |\tilde{\lambda} - \lambda_i|$,

Za lastni vektor, izračunan preko RRR, pa velja

$$|\sin \angle(z, x_j)| \leq \frac{\mathcal{O}(nu)}{\text{relgap}(\tilde{\lambda})},$$

kjer je $\text{relgap}(\tilde{\lambda}) = \frac{\text{gap}(\tilde{\lambda})}{|\tilde{\lambda}|}$.

Kadar je $|\tilde{\lambda}|$ dosti manjša kot $\|T\|$, je RRR lahko veliko natančnejši.

Končni algoritem (MRRR)

- 1) Poišči τ da bo LDL^T razcep matrike $T + \tau I$ RRR.
- 2) Izračunaj $T + \tau I = L_0 D_0 L_0^T$
- 3) Relativno natančno izračunaj lastne vrednosti $\tilde{\lambda}_1, \dots, \tilde{\lambda}_n$ matrike $L_0 D_0 L_0^T$.
- 4) $l = 1, m = n$
- 5) Razdeli lastne vrednosti $\tilde{\lambda}_l, \dots, \tilde{\lambda}_m$ na izolirane in gruče.
- 6) Za vsako izolirano lastno vrednost $\tilde{\lambda}_j$ izračunaj z_j preko GetVec.
- 7) Za vsako gručo $\tilde{\lambda}_j, \dots, \tilde{\lambda}_{j+k-1}$
 - Preko dstqds izračunaj $L_0 D_0 L_0^T - \tau_s I = L_S D_S L_S^T$, kjer τ_s izbereš tako, da
 - bo v $L_S D_S L_S^T$ vsaj ena lastna vrednost izolirana,
 - bo $L_S D_S L_S^T$ delna RRR($j, \dots, j+k-1$).
 - Relativno natančno izračunaj lastne vrednosti $\mu_j, \dots, \mu_{j+k-1}$ matrike $L_S D_S L_S^T$.
 - Določi $\tilde{\lambda}_i = \mu_i$ za $i = j, \dots, j+k-1$
 - $l = j, m = j+k-1, L_0 = L_s, D_0 = D_s$, rekurzivno se vrni na korak 5).

Kriterij za izolirano lastno vrednost oz. gručo

Za lastno vrednost $\tilde{\lambda}_j$ pravimo, da je relativno izolirana, če velja

$$\text{relgap}(\tilde{\lambda}_j) \geq \delta.$$

Lastne vrednosti $\tilde{\lambda}_j, \dots, \tilde{\lambda}_{j+k-1}$ tvorijo gručo, če velja

$$\begin{aligned} \text{reldis}(\tilde{\lambda}_i, \tilde{\lambda}_{i+1}) &< \delta \quad \text{za} \quad i = j, \dots, j+k-2 \\ \text{reldis}(\tilde{\lambda}_{j-1}, \tilde{\lambda}_j) &\geq \delta \\ \text{reldis}(\tilde{\lambda}_{j+k-1}, \tilde{\lambda}_{j+k}) &\geq \delta, \end{aligned}$$

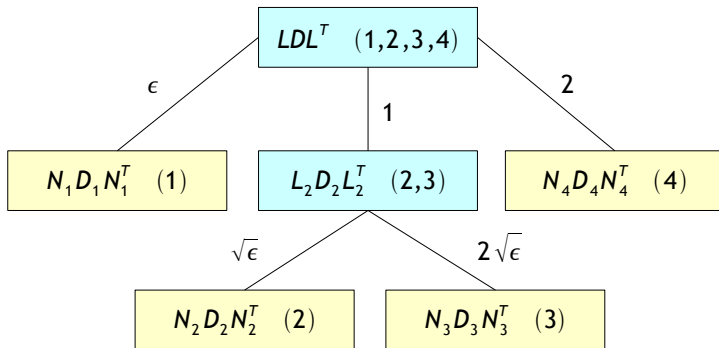
kjer je

$$\text{reldis}(\lambda, \mu) = \frac{|\lambda - \mu|}{|\lambda|}.$$

Za δ vzamemo npr. 10^{-3} .

Predstavitveno drevo

Matrika ima lastne vrednosti ϵ , $1 + \sqrt{\epsilon}$, $1 + 2\sqrt{\epsilon}$, 2 .



- Listi predstavljajo izračune lastnih vektorjev (GetVec).
- Povezave predstavljajo prehod na delno RRR premaknjene matrike (dstqds).

- Avtor RRR algoritma je Inderjit S. Dhillon (1997)
- Podrobnosti sta v člankih objavila Dhillon in Parlett (2000, 2003, 2004)
- Zahtevnost metode za izračun k lastnih parov je $\mathcal{O}(kn)$.
- Razvili so tudi varianto za singularni razcep bidiagonalne matrike
- V LAPACKU je prisoten od verzije 3.0 dalje (1999)
- Predvideva se, da bo sčasoma postal osnovna metoda za reševanje simetričnega problema lastnih vrednosti
- Metoda je večinoma hitrejša od ostalih in potrebuje najmanj dodatnega spomina