Alkalmazott algebra - algoritmusok

Ivanyos Gábor

2011 ősz

A hatvány-iteráció

- **Egyszerű konvergencia-tétel - változat:** Tfh. A karakterisztikus polinomja $(x-\lambda)\prod_{i=1}^{n-1}(x-\mu_i)$, ahol $|\mu_i| < |\lambda|$ $(i = 1, \ldots, n-1)$. Legyen $v^0 \notin (A-\lambda I)\mathbb{C}^n$ és $v^k = A^k v^0$. Ekkor a $\langle v^k \rangle$ 1-dim altér "konvergens" és a "határérték" a λ-hoz tartozó sajátaltér.

  Változatok diagonalizálható mátrixokra, pl. a következő:

- **Pozitív szemidefinitre:** Tfh. A pozitív szemidef, $\lambda > 0$ legnagyobb sajátértékkel. Ekkor $\lim_{k\to\infty}(\frac{1}{\lambda}A)^k$ a merőleges vetítés A λ-sajátalterére. Így **majdnem minden** v^0-ra $\langle A^k v^0 \rangle$ egy λ-sajátvektor által gen. altérhez tart:

- **Hatvány-iteráció:** $v^0 :=$ véletlen (egység)vektor, $v^{k+1} := Av^k$, "lenormálva".

  Poz. szemidef A-ra, nagy k-ra v^k közelítő sajátvektor

A QR-felbontás

- A $m \times n$-es valós
- $A = QR$, ahol Q $m \times m$-es ortogonális, R $m \times n$-es felső háromszög

QR-felb. Gram-Schmidt-ortogonalizációval

Egyszerűsítés: $m = n$, A invertálható

- $A^T A$-ra Gram-Schimdt: $G^T A^T AG = I_n$,
- **Emlék:** G felső háromszög
- $Q := AG$ ortogonális: $Q^T Q = (AG)^T AG = G^T A^T AG = I_n$.
- $Q = AG$, $R = G^{-1}$-gyel $A = QR$.

Householder-tükrözések

- **Emlék. (hipersíkra tükrözés):**
  - $u \neq 0$ rögz., $u^{\perp}$-re való tükrözés:
  - $\tau_u : v \mapsto v - 2\frac{(u,v)}{(u,u)}u = v - 2\frac{u^*v}{u^*u}u$

    mátrixa a standard bázisban: $I - \frac{2}{u^*u}uu^*$
  - ortogonális és szimmetrikus egyszerre
- **Szögfelezőre tükrözés:**
  - Tfh. $v, w \neq 0$.
  - $u = \frac{|v|}{|w|}w - v$
  - $u^{\perp} = v$ és w szögfelező hipersíkja
  - $\tau_u : v \leftrightarrow \frac{|v|}{|w|}w$

QR-felbontás Householder-tükrözésekkel

- A $m \times n$-es valós
- $v = A$ első oszlopa, $w = \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}$
- Ha $v \neq 0$, legyen T_1 a $v \leftrightarrow |v|w$ tükr. mátrixa
- Ha $v = 0$, $T_1 = I$
- $T_1 A$ első oszlopa $T_1 v = |v|w$.
- $T_1 = I_n - 2u'u'^T$, ahol u' a szögfelezőre $\perp$ egységvektor
- $T_1 A = A - 2u'u^T A$, $O(n^2)$ művelettel

QR-felbontás Householder-tükrözésekkel II.

- $T_1 A$ első oszlopa rendben
- Rekurzió A jobb alsó $n-1 \times n-1$-es A' blokkjára:

$$A' = T_2' \cdots T_\ell' R',$$

ahol R' $m-1 \times n-1$-es felső háromszög, $\ell \leq \min(m-1, n-1)$.

- legyen $i = 2, 3 \ldots$-re $T_i = \begin{pmatrix} 1 & \\ & \boxed{T_i'} \end{pmatrix}$.
- $R = \begin{pmatrix} ? & ? \\ & \boxed{R'} \end{pmatrix}$ (az első sor $T_1 A$ első sora)
- $T_1 A = T_2 \cdots T_\ell R$, így $A = QR$, ahol $Q = T_1 T_2 \cdots T_\ell$.
- összköltség: $O(\min(m, n) m^2)$.

QR-felbontás - megjegyzések

- **"Egyértelműség":** Tfh. $A = QR$ oszlopai lin. ftlenek, R főátlóbeli elemei pozitívak. Ekkor $A^T A = R^T R$ és R felső $n \times n$-es blokkja éppen $A^T A$ Gram-Schmidt-ortogonalizációjának felel meg.
- **Változat:** $AP = QR$, ahol P permutációs mátrix és R felső delta alakú: valamely k indexre $r_{jj} \neq 0$, ha $j \leq k$; $r_{ij} = 0$ ha $i > j$ vagy ha $i > k$.

  Előállítás: Az eredeti módszer azzal, hogy ha csupa 0 maradékú oszlop jönne, kicseréljük egy olyannal, aminek a maradéka nem csupa 0 (addig, amíg van ilyen).

QR-felbontás - alkalmazások

- determináns, rang, képtér
- numerikusan stabilabb, mint Gauss-elim, vagy Gram-Schimdt
- LSI-ben SVD helyett: könnyebben számolható, gyakran viszonylag tűrhető alacsony rangú közelítés kapható R szabdalásával

A QR-algoritmus

- $A_1 = A$
- $A_1 = Q_1 R_1$ (Q_1 ortog., R_1 felső háromszög)

$\vdots$

- $A_i = Q_i R_i$ (Q_i ortog., R_i felső háromszög)
- $A_{i+1} := R_i Q_i$
- $A_{i+1} = Q_{i+1} R_{i+1}$ (Q_{i+1} ortog., R_{i+1} felső háromszög)

- **Észrevétel:** $A_{i+1} = Q_i^T A_i Q_i$
- tehát $A_1, A_2, \ldots, A_i, A_{i+1}, \ldots$ mátrixok hasonlók
- sőt, ortog. konjugáló mátrixszal

A QR-algoritmus II.

- $A_1 = A$

  $\vdots$
- $A_i = Q_i R_i$
- $A_{i+1} := R_i Q_i$

  $\vdots$

- Alkalmas feltételek mellett konvergál A (egy) Schur-felbontásához.
- **Spec. eset.:** Ha A pozitív def. valós szimm.,

  - a határérték diagonális, sőt, - a Q_i-k szorzatának határértéke diagonalizálja A-t.
- Vannak javított, adott körülményekhez igazított változatok

Hessenberg-mátrixok

- **Def.:** $A = (a_{ij})$ $n \times n$-es mátrix felső Hessenberg-alakú, ha $i > j + 1$ esetén $a_{ij} = 0$:
$$A = \begin{pmatrix} * & * & * & * & * \\ * & * & * & * & * \\ & * & * & * & * \\ & & * & * & * \\ & & & * & * \end{pmatrix}$$
- **Tul.:** A $n \times n$-es felső Hessenberg, B $n \times n$-es felső háromszög $\Rightarrow$ AB és BA is felső Hessenberg.
Biz.: $B = (b_{ij})$, $b_{ij} = 0$, ha $i > j$. Legyen $AB = (c_{ij})$. $c_{ij} = \sum_{k=1}^{n} a_{ik} b_{kj}$. Ha $i > k + 1$, akkor $a_{ik} = 0$ és ha $k > j$, akkor $b_{kj} = 0$. Így $a_{ik} b_{kj} = 0$, kivéve esetleg $i \leq k + 1$ és $k \leq j$. Ha $i > j + 1$, nincs ilyen k, tehát $c_{ij} = 0$.

  BA-ra hasonló biz.

Hessenberg-alakra hozás Householder-tükrözésekkel

- A $n \times n$-es
- **Első lépés:** $T_1 = \begin{pmatrix} 1 & \\ & \boxed{T_1'} \end{pmatrix}$, ahol T_1' az A alsó $n-1$ soros részének az első oszlopát rendbe tevő Householder-tükrözés (vagy I_{n-1}). Ekkor $T_1 A \begin{pmatrix} * & * & * & * & * \\ * & * & * & * & * \\ & * & * & * & * \\ & * & * & * & * \\ & * & * & * & * \end{pmatrix}$ alakú, és ez megmarad a T_1-gyel való jobbról szorzással is: az nem bántja az első oszlopot.
- A QR-felbontáshoz hasonló rekurzióval vagy iterációval:
- **Végeredmény:** $A = T_1 \cdots T_{n-3} H T_{n-3} \cdots T_1$, ahol H felső Hessenberg.
- **Költség:** $O(n^3)$.

QR-algoritmus Hessenberg-mátrixokkal

- Ha A felső Hessenberg- alakú, a QR-felbontásánál a T_i Householder-tükrözések csak az i-edik és $i+1$-edik sort érintik:
$$T_i = \begin{pmatrix} 1 & & & & & & \\ & \ddots & & & & & \\ & & 1 & & & & \\ & & & * & * & & & \\ & & & * & * & & & \\ & & & & & 1 & & \\ & & & & & & \ddots & \\ & & & & & & & 1 \end{pmatrix}$$
- T_i-vel való szorzás kölsége $O(n)$, QR-felb. összköltsége $O(n^2)$.
- Ha $A_i = Q_i R_i$ f. Hess., akkor Q_i is, és így $A_{i+1} = R_i Q_i$ is.
(invertálható esetre: $Q_i = A_i R_i^{-1}$)
- érdemes QR-algoritmus előtt f. Hessenberg-alakra hozni

QR-algoritmus szimmetrikus tridiagonális mátrixokra

- A szimmetrikus $\longrightarrow U^TAU$ felső Hessenberg és szimmetrikus
  ($U = T_1 \cdots T_{n-2}$ Hessenberg-alakra hozó tükrözések)
- Szimmetrikus Hessenberg-mátrixok = tridiagonális mátrixok:

$$\begin{pmatrix} * & * & & & \\ * & * & * & & \\ & * & * & * & \\ & & * & * & * \\ & & & * & * \end{pmatrix}$$

- Tridiag. mátrix QR-felbontása:
  - Hoseholder-tükrözések: 2 sort érintenek
  - 1 sorban max 3 nemnulla elem
  - Összköltség: $O(n)$
- $A_i = Q_iR_i$ szimm. tridiag. $\Rightarrow A_{i+1} = R_iQ_i = Q_i^TA_iQ_i$ is szimm. tridiag.

Hessenberg-alakra hozó bázis

- U ortog, U oszlopai: $v_1, \dots, v_n$ ortonormált bázis
- $U^T AU$ felső Hessenberg $\Leftrightarrow j = 1, \dots, n-1$-re:

$$A\langle v_1, \dots, v_j \rangle \leq \langle v_1, \dots, v_j, v_{j+1} \rangle$$

- **Egyenértékű:** $j = 1, \dots, n-1$-re:

$$Av_j \in \langle v_1, \dots, v_j, v_{j+1} \rangle$$

Biz.: $i < j$-re $Av_i \in \langle v_1, \dots, v_i, v_{i+1} \rangle \leq \langle v_1, \dots, v_j \rangle$

- **Eljárás** ($\sim$ Arnoldi)
  - v_1 tetsz. egységvektor
  - Tfh. $v_1, \dots, v_j$ már megvan
  - Ha $Av_j \in \langle v_1, \dots, v_j \rangle$, v_{j+1} tetsz. $v_1, \dots, v_j$-re $\perp$ egységvektor.
  - Különben v_{j+1} legyen Av_j-nek $v_1, \dots, v_j$-re $\perp$ része, egységhosszúra normálva

Lánczos algoritmusa

- Arnoldi szimmetrikus A-ra
- **Tudjuk:** az eredmény szimm. Hessenberg, azaz tridiag:

$$U^T A U = \begin{pmatrix} a_1 & b_1 & & & \\ b_1 & a_2 & b_2 & & \\ & b_2 & a_3 & \ddots & \\ & & \ddots & \ddots & b_{n-1} \\ & & & b_{n-1} & a_n \end{pmatrix} = \begin{pmatrix} \cdot & \ddots & & & & \\ \ddots & \ddots & b_{j-1} & & & \\ & b_{j-1} & a_j & b_j & & \\ & & b_j & a_{j+1} & b_{j+1} & \\ & & & \ddots & \ddots & \ddots \\ & & & & \ddots & \cdot \end{pmatrix}$$

- Tehát $Av_j = b_{-1} v_{j-1} + a_j v_j + b_j v_{j+1}$

Lánczos algoritmusa II.

$$Av_j = b_{-1}v_{j-1} + a_j v_j + b_j v_{j+1}$$

- $v_0 := 0,\ b_0 := 0.$
- v_1 véletlen egységvektor
- $j = 1, \ldots, n-1$:
  - $w'_j := Av_j - b_{j-1}v_{j-1}$
  - $a_j := v_j^T w_j$
  - $w_{j+1} := w'_j - a_j v_j$
  - $b_j := |w_{j+1}|$
  - Ha $b_j \neq 0$, akkor $v_{j+1} = \frac{1}{b_j} w_{j+1}$
  - Ha $b_j = 0$, akkor v_{j+1} véletlen egységvektor (?)
- **Költség:** $O(n^2) + n - 1$ vektor szorzása A-val

  Jó, ha A ritka vagy 2 ritka szorzata (pl. LSI).
- Nem igazán robusztus: v_i-k ortogonalitása elromolhat ...

SVD közelítő kiszámítása

- A $m \times n$-es, $m \geq n$
- **Teljes SVD:** $A = M'\Sigma' M^T$
- M az $A^T A$-t diagonalizáló ortog: $M^T A^T A M = \Sigma'^2$
- M' az AM-ből
- **Kézenfekvő megközelítés** M kiszámítása $A^T A$ diagonalizálásával
- **Például:**
  - Először tridiagonalizáljuk $A^T A$-t (Householder-tükrözésekkel v. Lánczossal)
  - Majd QR-algoritmus
- Nem elég stabil

SVD kiszámítása II.

- Egyszerűsítő feltevés: $m = n$.
- **Észrevétel:** U_1, U_2 unitér

  A SVD-je $\longleftrightarrow$ $U_1 A U_2$ SVD-je

  különösen, ha U_1, U_2 könnyen számolható

  pl. Householder-tükrözések szorzata
- Először bidiagonlizáljuk A-t: $U_1 A U_2 = \begin{pmatrix} * & * & & & \\ & * & * & & \\ & & * & * & \\ & & & * & * \\ & & & & * \end{pmatrix}$
- **Megj.:** Bidiagonális A-re $A^T A$ tridiagonális

SVD kiszámítása III - bidiagonalizálás

- T_1 Householder-tükrözéssel:

$$T_1 A = \begin{pmatrix} * & * & * & * & * \\ & * & * & * & * \\ & * & * & * & * \\ & * & * & * & * \\ & * & * & * & * \end{pmatrix},$$

- Majd a második oszlopnál kezdődő Z_1 Householder-tükrözéssel

$$T_1 A Z_1 = \begin{pmatrix} * & * & & & \\ & * & * & * & * \\ & * & * & * & * \\ & * & * & * & * \\ & * & * & * & * \end{pmatrix}.$$

SVD kiszámítása IV - bidiagonalizálás II

- Első menet után:

$$T_1AZ_1 = \begin{pmatrix} * & * & & & \\ & * & * & * & * \\ & * & * & * & * \\ & * & * & * & * \\ & * & * & * & * \end{pmatrix}.$$

- Következő menet:

$$T_2T_1AZ_1Z_2 = \begin{pmatrix} * & * & & & \\ & * & * & & \\ & & * & * & * \\ & & * & * & * \\ & & * & * & * \end{pmatrix}.$$

- Végül $T_{n-1}\cdots T_1AZ_1\cdots Z_{n-2}$ bidiagonális.
- **Megj.** Az eljárás hasonlít a Hessenberg-alakra hozáshoz

SVD kiszámítása IV - A^TA helyett másik szimmetrizált

- Tfh. $M'^T A M = \Sigma'$, M, M' ortog.
- $M^T A^T M' = \Sigma'^T = \Sigma'$
- $\frac{1}{\sqrt{2}}\begin{pmatrix} M & M \\ M' & -M' \end{pmatrix}$ ortog. és
- $\frac{1}{2}\begin{pmatrix} M^T & M'^T \\ M^T & -M'^T \end{pmatrix}\begin{pmatrix} & A^T \\ A & \end{pmatrix}\begin{pmatrix} M & M \\ M' & -M' \end{pmatrix} =$
  $\frac{1}{2}\begin{pmatrix} M'^T A & M^T A^T \\ -M'^T A & M^T A^T \end{pmatrix}\begin{pmatrix} M & M \\ M' & -M' \end{pmatrix} =$
  $\frac{1}{2}\begin{pmatrix} M'^T AM + M^T A^T M' & M'^T AM - M^T AM' \\ -M'^T AM + M^T A^T M' & -M'^T AM - M^T AM' \end{pmatrix} = \begin{pmatrix} \Sigma' & \\ & -\Sigma' \end{pmatrix}$,
- azaz $\begin{pmatrix} & A^T \\ A & \end{pmatrix}$ sajátértékei A szing. értékei $\pm$ előjellel, és $\frac{1}{\sqrt{2}}\begin{pmatrix} M & M \\ M' & -M' \end{pmatrix}$ oszlopai adják a sajátvektorok egy ortonormált bázisát (ld. zh091030, 6. feladat).

SVD kiszámítása V

- A szing. értékei $\longleftrightarrow$ $\begin{pmatrix} & A^T \\ A & \end{pmatrix}$ sajátértékei (előjel erejéig)
- A SVD-ja $\longleftrightarrow$ $\begin{pmatrix} & A^T \\ A & \end{pmatrix}$ sajátvektorai
- **Észrevétel:** Ha A bidiag., akkor alaklmas permutáció P permutációmátrixszal $P^T \begin{pmatrix} & A^T \\ A & \end{pmatrix} P$ szimm. tridiag, 0 főátlóval.
- Az $\begin{pmatrix} & A^T \\ A & \end{pmatrix}$ mátrixot csak implicite használják az algoritmusok
- Numerikusan stabilabb.