

3.4 Arnoldijev algoritem za lastne vrednosti

Za nesimetrične matrice podprostor Krilova zgeneriramo z Arnoldijevim postopkom.

Algoritem se konča pri izbranem k ali pa, ko je $h_{j+1,j} = 0$.

$$\begin{aligned}j &= 1, \dots, k \\z &= Av_j \\i &= 1, \dots, j \\h_{ij} &= v_i^T z \\z &= z - h_{ij}v_i \\h_{j+1,j} &= \|z\| \\ \text{če je } h_{j+1,j} &= 0, \text{ potem prekini računanje} \\v_{j+1} &= z/h_{j+1,j}\end{aligned}$$

Če se Arnoldijev postopek izvaja do $j = k$, dobimo

$$AV_k = V_k H_k + h_{k+1,k} v_{k+1} e_k^T = V_{k+1} H_{k+1,k},$$

kjer je H_k $k \times k$ zgornja Hessenbergova, stolpci V_k pa so ON baza za $\mathcal{K}_k(A, v_1)$.

Pri Arnoldiju je (μ, z) Ritzev par, če je $z = V_k s$, kjer je μ lastna vrednost H_k , s pa njen pripadajoči lastni vektor.

Konvergenca Ritzevih lastnih vrednosti pri Arnoldijevi metodi

Imamo

$$AV_k = V_k H_k + h_{k+1,k} v_{k+1} e_k^T = V_{k+1} H_{k+1,k},$$

kjer je H_k $k \times k$ zgornja Hessenbergova, stolpci V_k pa so ON baza za $\mathcal{K}_k(A, v_1)$.

Naj bo (μ, z) Ritzev par, torej $z = V_k s$, kjer je μ lastna vrednost H_k , s pa njen pripadajoči lastni vektor.

Za ostanek potem dobimo

$$r = Az - \mu z = h_{k+1,k} v_{k+1} e_k^T s,$$

torej

$$\|r\| = |h_{k+1,k}| \cdot |e_k^T s|.$$

Če naj bo (μ, z) dober približek za lastni par matrike A , mora biti ostanek majhen in to lahko testiramo brez računanja z .

V primeru nesimetrične matrike majhen ostanek še ni nujno dovolj, saj za matrike, ki se dajo diagonalizirati kot $A = X \Lambda X^{-1}$ velja Bauer-Fikeov izrek

$$\min_i |\lambda_i - \mu| \leq \|r\| \|X\| \|X^{-1}\|.$$

Zgled 1

Vzamemo matriko

$$A = [e_2 \ e_3 \ \cdots \ e_n \ e_1],$$

ki ima ortonormirane stolpce, lastne vrednosti pa so enakomerno razporejene po enotski krožnici v \mathbb{C} .

Če vzamemo $v_1 = e_1$, potem dobimo $v_2 = Av_1 = e_2$, $v_3 = Av_2 = e_3, \dots$

Za $k < n$ je $H_k = A(1 : k, 1 : k)$, kar pomeni, da so vse Ritzve vrednosti enake 0. Šele v koraku $k = n$ dobimo neničelne Ritzve vrednosti.

Tu nimamo težav z občutljivostjo, saj je občutljivost matrike lastnih vektorjev enaka 1.

Vemo, da za podprostor Krilova velja

$$\mathcal{K}_k(\alpha A + \beta I, u_1) = \mathcal{K}_k(A, u_1)$$

za poljuben β in $\alpha \neq 0$. Torej dobimo isti podprostor Krilova ne glede na to, če matriko pomnožimo s skalarjem oziroma uporabimo pomik. To pomeni, da lahko podobne težave pričakujemo v primeru, ko so lastne vrednosti enakomerno razporejene na poljubni krožnici v kompleksni ravnini. [ZgledArn1](#)

Zgled 2

Vzamemo matriko

$$A = \text{diag} \left(\text{diag}(1, 2, \dots, 98), \begin{bmatrix} 100 & 1 \\ -1 & 100 \end{bmatrix} \right),$$

ki ima lastne vrednosti $1, 2, \dots, 98, 100 + i, 100 - i$. [ZgledArn2](#)

Šele po nekaj korakih dobimo kompleksne lastne vrednosti. Tik pred tem, ko se pojavijo kompleksne lastne vrednosti, se za največjo Ritzevo vrednost zdi, da je že skonvergirala.

Zgleda 3 in 4

Vzamemo petdiagonalno matriko s konstantnimi vrednostmi na diagonalah

$$A = \begin{bmatrix} 2 & 1 & -2 & & & \\ 1 & 2 & 1 & -2 & & \\ 2 & 1 & 2 & 1 & -2 & \\ & \cdots & \cdots & \cdots & \cdots & \cdots \end{bmatrix}.$$

ZgledArn3 Konvergence skoraj ni, razen k ekstremnim štirim lastnim vrednostim.

Spet vzamemo petdiagonalno matriko s konstantnimi vrednostmi na diagonalah, tokrat

$$A = \begin{bmatrix} 2 & 1 & -0.4 & & & \\ 0 & 2 & 1 & -0.4 & & \\ 2 & 0 & 2 & 1 & -0.4 & \\ & \cdots & \cdots & \cdots & \cdots & \cdots \end{bmatrix}.$$

ZgledArn4 Ritzeve vrednosti bolje aproksimirajo numerični zaklad matrike kot pa njen spekter.

$$F(A) = \left\{ \frac{x^H A x}{x^H x} : x \in \mathbb{C}^n, x \neq 0 \right\}.$$

Konvergenca

Pri Arnoldijevi metodi je konvergenca odvisna od izbire začetnega vektorja.

Zanima nas konvergenca proti lastnemu paru (λ, x) .

Naj bo $X = [x \ X_2]$ obrnljiva matrika. Potem je

$$X^{-1}AX = \begin{bmatrix} \lambda & 0 \\ 0 & A_{22} \end{bmatrix}.$$

Če s transformacijo X preslikamo začetni vektor u_1 , dobimo $X^{-1}u_1 = \begin{bmatrix} u_\lambda \\ u_r \end{bmatrix}$.

Velja ocena:

$$\tan \theta(x, \mathcal{K}_k) \leq \kappa(X) \min_{\substack{p \in \mathbb{P}_{k-1} \\ p(\lambda)=1}} \frac{\|p(A_{22})u_r\|_2}{|u_\lambda|}.$$

To bo majhno, če bo polinom imel majhne absolutne vrednosti pri vseh ostalih lastnih vrednostih.

Hitrost konvergence

Iščemo polinom p stopnje $k - 1$, za katerega velja $p(\lambda_1) = 1$ in zavzame najmanjšo absolutno vrednost pri ostalih lastnih vrednostih. To je povezano s problemom najboljše enakomerne aproksimacije.

Če je Ω kompaktna množica, lahko definiramo $\|f\|_\infty = \max_{z \in \Omega} |f(z)|$. V našem primeru bomo za Ω vzeli spekter matrike A brez λ_1 .

Sedaj v bistvu iščemo polinom oblike $(z - \lambda_1)q(z)$, kjer je $q \in \mathbb{P}_{k-1}$, ki je najboljša enakomerna aproksimacija za $f(z) = 1$ na Ω oziroma minimizira

$$\|1 - (z - \lambda_1)q(z)\|_\infty$$

po vseh $q \in \mathbb{P}_{k-1}$.

To je spet povezano s polinomi Čebiševa.

EksPLICITNI ponovni zagon

Denimo, da naredimo najprej m korakov Arnoldijeve metode, potem pa metodo ponovno zaženemo z začetnim vektorjem u_+ , ki ga izberemo iz prostora $\text{Lin}(u_1, \dots, u_m)$.

Potem je $u_+ = p(A)u_1$ za nek polinom p stopnje kvečjemu $m - 1$.

Če u_1 razvijemo po bazi lastnih vektorjev kot

$$u_1 = a_1x_1 + \dots + a_nx_n,$$

potem velja

$$u_+ = a_1p(\lambda_1)x_1 + \dots + a_np(\lambda_n)x_n.$$

Pričakujemo lahko, da bo $\mathcal{K}_m(A, u_+)$ vseboval dobre aproksimacije za tiste lastne vrednosti, kjer je absolutna vrednost $p(\lambda)$ velika in slabe aproksimacije za lastne vrednosti, kjer je ta vrednost majhna.

Izbira u_+ v bistvu pomeni, da izberemo polinom, ki dobro separira lastne vrednosti na željene in neželjene.

Arnoldi s polinomskim filtriranjem

Začetni vektor u_1 iterativno menjamo s $q(A)u_1$, kjer je q polinom, ki ima veliko absolutno vrednost pri željenih lastnih vrednostih in majhno pri neželjenih lastnih vrednostih.

Podatke o območju željenih lastnih vrednosti prav tako pridobivamo iterativno z Arnoldijevo metodo.

1. izvedbi m korakov Arnoldijeve metode. Ritzeve vrednosti razdeli na željene μ_1, \dots, μ_k in neželjene μ_{k+1}, \dots, μ_m .
2. izberi tak polinom q stopnje $\leq m$, da je $|g(\mu_i)| \gg |g(\mu_j)|$ za $i = 1, \dots, k$ in $j = k + 1, \dots, m$.
3. vzemi $u_1 = q(A)u_1 / \|q(A)u_1\|$ in nadaljuj v točki 1.

To je posebej primerno zato, ker se z večanjem dimenzije podprostora Krilova zahtevnost Arnoldijeve metode zelo poveča. V vsakem koraku je namreč potrebno novi vektor ortogonalizirati na vse prejšnje. Zaradi tega je več ponovnih zagonov boljša rešitev.

Arnoldi z implicitnim zagonom (Sorensen (1992))

Če delamo ponovni zagon samo z enim vektorjem, lahko izgubimo preveč podatkov. Bolje je nadaljevati s primerno izbranim k -razsežnim podprostorom.

Pri IRA (Implicitly Restarted Arnoldi) naredimo $M = k + m$ korakov Arnoldija. Dobimo

$$AV_M = V_M H_M + h_{M+1,M} v_{M+1} e_M^T = V_{M+1} H_{M+1,M}.$$

Na H_M izvedemo m korakov QR algoritma s premiki $\theta_1, \dots, \theta_m$ in dobimo

$$H_M^{(m)} = Q^H H_M Q.$$

Sedaj za nov začetni podprostor \tilde{V}_k vzamemo prvih k stolpcev $V_M Q$, za \tilde{H}_k pa vodilno podmatriko $H_M^{(m)}$.

Izkaže se, da je

$$\tilde{v}_1 = \gamma (A - \theta_1 I) \cdots (A - \theta_m I) v_1,$$

stolpci \tilde{V}_k pa so ON baza za $\mathcal{K}_k(A, \tilde{v}_1)$.

Algoritem za IRA

izberi v_1, k, m

izvedi k korakov Arnoldija $\implies V_{k+1}, H_{k+1,k}$.

$$r = h_{k+1,k}v_{k+1}$$

dokler $\|r\| > \epsilon$ ponavljaj

izvedi m korakov Arnoldija $\implies V_{M+1}, H_{M+1,M}$.

izberi m premikov $\theta_1, \dots, \theta_m$ (npr. neželene Ritzve vrednosti)

naredi m korakov implicitnega QR za H_M s premiki $\theta_1, \dots, \theta_m \implies Q, \tilde{V}_{M+1}, H_M^{(m)}$

$$V_k = \tilde{V}_k$$

$$H_k = H_k^{(m)}$$

$$r = h_{k+1,k}^{(m)}\tilde{v}_{k+1} + h_{M+1,M}v_{M+1}e_M^T Q e_k$$

$$h_{k+1,k} = \|r\|$$

$$v_{k+1} = r/h_{k+1,k}$$

Premakni in obrni (Shift-and-invert)

Če iščemo lastne vrednosti v bližini τ , potem si lahko pomagamo s podprostorom Krilova, ki ga generira matrika $(A - \tau I)^{-1}$.

Pri tej metodi je potrebno produkt z matriko $(A - \tau I)^{-1}$ izračunati točno.

Pomagamo si lahko npr. z LU razcepom. Če je $P(A - \tau I) = LU$, potem moramo v vsakem koraku Arnoldija namesto $v_{j+1} = (A - \tau I)^{-1}v_j$ novi vektor izračunati kot

$$Ly = Pv_j$$

$$Uv_{j+1} = y.$$

Če je A razpršena matrika, to ponavadi velja tudi za L in U , ki pa lahko imata več neničelnih elementov kot A . Najbolj zahtevna operacija pri premakni in obrni je izračun LU razcepa matrike $A - \sigma I$. Ker pa je konvergenca potem lahko zelo hitra in porabimo veliko manj korakov Arnoldijevega algoritma, se pristop premakni in obrni večinoma splača, kadar iščemo notranje lastne vrednosti.

Če iščemo kompleksne lastne vrednosti, je najbolje vzeti kompleksni τ . Vendar, potem moramo računati v kompleksni aritmetiki. Ali je možno priti do željenih kompleksnih lastnih vrednosti samo z realno aritmetiko?

Iterativne metode v Matlabu

V Matlabu je na voljo metoda `eigs`, ki računa lastne vrednosti s pomočjo Arnoldijeve metode z implicitnim ponovnim zagonom iz paketa ARPACK.

- `d = eigs(A)` vrne 6 po absolutni vrednosti največjih lastnih vrednosti matrike A .
- `[V,D] = eigs(A)` vrne vektor lastnih vektorjev V in diagonalno matriko lastnih vrednosti D za 6 po absolutni vrednosti največjih lastnih vrednosti A .
- `[V,D,flag] = eigs(A)` deluje tako kot zgoraj, le da dodatno vrne še ali so lastne vrednosti skonvergirale (`flag=0`) ali ne (`flag≠0`).
- `eigs(A,B)` rešuje splošeni problem lastnih vrednosti $Ax = \lambda Bx$.
- `eigs(A,k)` vrne k največjih lastnih vrednosti.
- `eigs(A,k,sigma)` vrne k največjih lastnih vrednosti, pri čemer `sigma` pomeni:
 - če je `sigma` skalar, iščemo lastne vrednosti v bližini skalarja (preko premakni in obrni)
 - `'lm'`: največja absolutna vrednost,
 - `'sm'`: najmanjša absolutna vrednost,
 - `'lr'`: največji realni del,
 - `'li'`: največji imaginarni del,
 - `'si'`: najmanjši imaginarni del.

Matriko lahko podamo kot funkcijo, ki za x vrne Ax oziroma $(A - \sigma I)^{-1}x$ v primeru `shift-and-invert`.

Harmonične Ritzve vrednosti in Arnoldi

Tudi pri Arnoldiju lahko uporabljamo harmonične Ritzve vrednosti.

Če uporabimo premik σ , je θ harmonična Ritzeva vrednost in u harmonični vektor, če je

$$Au - \theta u \perp (A - \sigma I)\mathcal{K}_k(A, v_1).$$

Lema 1. *Po k korakih Arnoldija dobimo $AV_k = V_k H_k + h_{k+1,k} v_{k+1} e_k^T = V_{k+1} H_{k+1,k}$. Če je ϑ lastna vrednost posplošenega problema lastnih vrednosti*

$$\vartheta(H_{k+1,k} - \sigma I_{k+1,k})^H (H_{k+1,k} - \sigma I_{k+1,k})y = (H_k - \sigma I_k)y,$$

potem je $\sigma + 1/\vartheta$ harmonična Ritzeva vrednost matrike A .

Izkaže se, da so harmonični Ritzevi vektorji kvalitetnejše aproksimacije za lastne vektorje, kot so harmonične Ritzve vrednosti za lastne vrednosti. Zato kot približek za lastno vrednost vzamemo Rayleighov kvocient harmoničnega vektorja.

Harmonični vektorji so posebej primerni za ponovni zagon. Če računamo notranje lastne vrednosti, bo tudi v primeru, ko je Ritzeva vrednost blizu tarče, Ritzev vektor lahko daleč od ustreznega lastnega vektorja. Pri harmoničnih vrednostih in vektorjih se to ne more zgoditi, saj ima harmonični vektor vedno močno komponento v smeri iskanega lastnega vektorja.

Računanje kompleksnih lastnih vrednosti

Naj bo A realna nesimetrična matrika, radi pa bi uporabili premik $\sigma = \sigma_1 + i\sigma_2$.

Namesto kompleksne aritmetike pri direktni uporabi metode premakni in obrni lahko npr. uporabimo tudi realni operator

$$B_+ = \operatorname{re}[(A - \sigma I)^{-1}] = \frac{1}{2} \left((A - \sigma I)^{-1} + (A - \bar{\sigma} I)^{-1} \right).$$

Velja namreč:

- Lastni vektorji matrike B_+ se ujemajo z lastnimi vektorji matrike A .
- Lastni vrednosti μ_i^+ ustreza

$$\mu_i^+ = \frac{1}{2} \left(\frac{1}{\lambda_i - \sigma_i} + \frac{1}{\lambda_i - \bar{\sigma}_i} \right).$$

Računanje več lastnih vrednosti

Če nas zanima več kot le ena lastna vrednost, potem je ena izmed možnosti uporaba tehnik t.i. zaklepanja (lock-and-restart).

Denimo, da smo že izračunali lastne vektorje x_1, \dots, x_{j-1} . Sedaj to vzamemo za prvih $j - 1$ vektorjev v_1, \dots, v_{j-1} , izberemo v_j in nadaljujemo s podprostorom, ki je kombinacija že izračunanih lastnih vektorjev in podprostora Krilova $\mathcal{K}(A, v_j)$.

Po k korakih dobimo podprostor, ki ga razpenjamo vektorji

$$v_1, \dots, v_{j-1}, v_j, Av_j, \dots, A^{k-j}v_{k-j}.$$

Za matriko H_k velja, da je podmatrika $H_k(1 : j, 1 : j)$ zgornja trikotna, kot približke za lastne vrednosti pa jemljemo v pošteve le Ritzove vrednosti spodnje podmatrike $H_k(j + 1 : k, j + 1 : k)$, ki je zgornja Hessenbergova.

Alternativna možnost je, da nadaljujemo računanje od začetka z matriko

$$\tilde{A} = (I - U_{j-1}U_{j-1}^H)A(I - U_{j-1}U_{j-1}^H),$$

kjer stolpci matrike $U_{j-1} = [u_1 \ \cdots \ u_{j-1}]$ sestavljajo ortonormirano bazo za invariantni podprostor, ki smo ga že našli.