

Arnoldiho a Lanczosova metoda

Částečný problém vlastních čísel

- Ne vždy je potřeba (a někdy to není ani technicky možné) nalézt celé spektrum dané matice (velké řídké matice).
- Úloze, ve které chceme aproximovat pouze **malou část spektra**, budeme říkat **částečný problém vlastních čísel**.
- Dvě základní metody,
Arnoldiho metoda → **obecné matice**
Lanczosova metoda → **hermitovské matice**.
- Stačí, když jsme schopni **realizovat násobení matice vektorem**.

Metody Krylovových podprostorů

- Dána **čtvercová regulární** $A \in \mathbb{R}^{n \times n}$, startovací **vektor** v .
- **Velké řídké matice** $\rightarrow$ jedinou operací, kterou jsme schopni s A provést, je **násobení** Av , A nemusí být dána explicitně.
- Využití této operace: budování posloupnosti podprostorů

$$\mathcal{K}_k(A, v) \equiv \text{span}\{v, Av, A^2v, \dots, A^{k-1}v\},$$

Krylovovy podprostory.

- **Problém** (řešení soustavy či problém vlastních čísel)
projektujeme na tyto podprostory a tím získáme postupné aproximace řešení.

Metody Krylovových podprostorů

Příklad - systém lineárních rovnic

- $Ax = b$, hledáme aproximaci x_k

$$x_k \in \mathcal{K}_k(A, b).$$

- Aby x_k jednoznačné $\rightarrow$ nutné předepsat **k podmínek**.
- Definujme k -té reziduum vztahem $r_k \equiv b - Ax_k$,

$$x_k \in \mathcal{K}_k(A, b) \quad \Rightarrow \quad r_k \in b + A\mathcal{K}_k(A, b).$$

Pokud $\|r_k\| = 0 \Rightarrow x_k$ je přesné řešení.

- Logický požadavek: $\|r_k\|$ co nejmenší v rámci daných omezení
 $\Leftrightarrow$ **ortogonální projekce** b na $A\mathcal{K}_k(A, b)$,

$$r_k \perp A\mathcal{K}_k(A, b).$$

Metody Krylovových podprostorů

Systém lineárních rovnic

- Pokud

$$\dim(\mathcal{K}_k(A, b)) = k,$$

pak je x_k určeno jednoznačně.

- b projektujeme do stále většího a většího podprostoru a **norma rezidua se zmenšuje**.
- Roste-li dimenze $\mathcal{K}_k(A, b)$, dostáváme pro $k = n$

$$A\mathcal{K}_n(A, b) = \mathbb{R}^n,$$

Potom je

$$r_n \perp \mathbb{R}^n \quad \Rightarrow \quad r_n = 0 \quad \Rightarrow \quad x_n = x.$$

Metody Krylovových podprostorů

Příklad - problém vlastních čísel $Ay = \lambda y$

- Vlastní vektor a příslušné vlastní číslo (vlastní pár) **budeme aproximovat** dvojicí (y_j, μ_j) tak, aby

$$y_j \in \mathcal{K}_k(A, v), \quad Ay_j - \mu_j y_j \perp \mathcal{K}_k(A, v),$$

v je daný startovací vektor.

- **Pokud** $\dim \mathcal{K}_k(A, v) = k$ a je-li A diagonalizovatelná, potom existuje **právě k párů** (y_j, μ_j) splňující tuto podmínku $\rightarrow$ získáme aproximace k vlastních čísel a vektorů.
- **Roste-li dimenze** Krylovových podprostorů, získáme pro $k = n$ všechny vlastní vektory a vlastní čísla matice A .

Dimenze Krylovových podprostorů

$$\mathcal{K}_k(A, v) \equiv \text{span}\{v, Av, A^2v, \dots, A^{k-1}v\}$$

- **Maximální dimenzi**, jež mohou Krylovovy podprostory generované maticí A a vektorem v dosáhnout, budeme nazývat **stupněm v vzhledem k A** ,

$$d(A, v) \equiv \min\{k; \dim \mathcal{K}_k(A, v) = \dim \mathcal{K}_{k+1}(A, v)\}.$$

- $d(A, v)$ je **dimenze nejmenšího** Krylovova podprostoru $\mathcal{K}_k(A, v)$ **invariantního** vzhledem k násobení maticí A

$$x \in \mathcal{K}_k(A, v) \Rightarrow Ax \in \mathcal{K}_k(A, v).$$

$d(A, v)$ je omezeno stupněm minimálního polynomu.

Báze Krylovových prostorů

- Abychom mohli s Krylovovými prostory pracovat $\rightarrow$ **potřebujeme rozumnou bázi**.
Báze $v, Av, \dots, A^{k-1}v$ **není vhodná k výpočtům** (špatná podmíněnost), viz mocninná metoda.
- Pro efektivní a numericky stabilní práci je nutné umět počítat **dobře podmíněnou bázi** (ideálně **ortogonální**).
- Chceme **bázové vektory** $v_1, \dots, v_k$ podprostoru $\mathcal{K}_k(A, v)$, které **vyhovují podmínkám**

$$v_{i+1} \in \mathcal{K}_{i+1}(A, v) \setminus \mathcal{K}_i(A, v), \quad \text{span}\{v_1, \dots, v_i\} = \mathcal{K}_i(A, v)$$

Bez újmy na obecnosti $\|v_i\| = 1, 1 \leq i \leq k$.

Arnoldiho algoritmus

Základní myšlenka

- Nechť $v_1, \dots, v_k$ splňují

$$\begin{aligned} v_{i+1} &\in \mathcal{K}_{i+1}(A, v) \setminus \mathcal{K}_i(A, v), \\ \text{span}\{v_1, \dots, v_i\} &= \mathcal{K}_i(A, v), \quad i = 1, \dots, k-1, \end{aligned}$$

a nechť $k < d(A, v)$.

- Potom

$$Av_k \in \mathcal{K}_{k+1}(A, v) \setminus \mathcal{K}_k(A, v).$$

- Idea **projekce**,

$$w = Av_k - \sum_{i=1}^k h_{i,k} v_i, \quad v_{k+1} = \frac{w}{\|w\|},$$

kde $h_{i,k}$ jsou vhodně zvolené koeficienty.

- $v_1, \dots, v_k, v_{k+1}$ splňují opět danou podmínku.

Podrobněji

- Krok 1:

$$v_1 \equiv \frac{v}{\|v\|}.$$

- Krok 2: urči w tak, aby $w \perp v_1$ a normuj

$$w = Av_1 - h_{1,1}v_1, \quad h_{1,1} = \langle Av_1, v_1 \rangle$$

$$h_{2,1} = \|w\|, \quad v_2 = \frac{w}{h_{2,1}},$$

$$Av_1 = h_{1,1}v_1 + h_{2,1}v_2.$$

- Krok $k + 1$:

$$w = Av_k - \sum_{i=1}^k h_{i,k}v_i, \quad h_{i,k} = \langle Av_k, v_i \rangle,$$

$$h_{k+1,k} = \|w\|, \quad v_{k+1} = \frac{w}{h_{k+1,k}}.$$

$$Av_k = h_{1,k}v_1 + h_{2,k}v_2 + \cdots + h_{k,k}v_k + h_{k+1,k}v_{k+1}.$$

Arnoldiho algoritmus

Algebraická formulace

$$Av_1 = h_{1,1} v_1 + h_{2,1} v_2,$$

$$Av_2 = h_{1,2} v_1 + h_{2,2} v_2 + h_{3,2} v_3,$$

$$\vdots$$

$$Av_k = h_{1,k} v_1 + \cdots + h_{k,k} v_k + h_{k+1,k} v_{k+1}$$

tj.

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

$$V_k \equiv [v_1, \dots, v_k], \quad H_k = \begin{bmatrix} h_{1,1} & h_{1,2} & \cdots & h_{1,k} \\ h_{2,1} & h_{2,2} & \cdots & h_{2,k} \\ & \ddots & \ddots & \vdots \\ & & h_{k,k-1} & h_{k,k} \end{bmatrix}.$$

Arnoldiho algoritmus

Algoritmus počítající ortogonální bázi Krylovova prostoru, který jsme právě odvodili nazveme **Arnoldiho algoritmem**.

```
1: input  $A, v$ 
2:  $v_1 = \frac{v}{\|v\|}$ 
3: for  $k = 1, 2, \dots$  do
4:    $w = Av_k$ 
5:   for  $i = 1 : k$  do
6:      $h_{i,k} = \langle w, v_i \rangle$ 
7:      $w = w - h_{i,k} v_i$ 
8:   end for
9:    $h_{k+1,k} = \|w\|$ 
10:   $v_{k+1} = \frac{w}{h_{k+1,k}}$ 
11: end for
```

$$A V_k = V_k H_k + h_{k+1,k} v_{k+1} e_k^T.$$

Ukončení Arnoldiho algoritmu

- K ukončení Arnoldiho algoritmu dochází tehdy, je-li $h_{k+1,k} = 0$, tj. $\|w\| = 0$ a platí

$$Av_k = \sum_{i=1}^k h_{i,k} v_i.$$

- Jinak řečeno, $Av_k \in \mathcal{K}_k(A, v)$ a platí

$$\begin{aligned}\mathcal{K}_k(A, v) &= \text{span}\{v_1, \dots, v_k\} \\ &= \text{span}\{v_1, \dots, v_k, Av_k\} = \mathcal{K}_{k+1}(A, v)\end{aligned}$$

a tudíž je $k = d(A, v)$.

- Arnoldiho algoritmus **najde ortogonální bázi celého** invariantního prostoru $\mathcal{K}_d(A, v)$.

Arnoldiho algoritmus počítá po sloupcích QR rozklad

- Arnoldi $\leftrightarrow$ Gram-Schmidt. Lze na Arnoldiho algoritmus nahlížet jako na **QR rozkladu nějaké matice?**
- Matici $[Av_1, \dots, Av_k]$ rozšíříme o první sloupec v_1 , platí

$$[v_1, Av_1, \dots, Av_k] = V_{k+1} \begin{bmatrix} 1 & h_{1,1} & h_{1,2} & \dots & h_{1,k} \\ & h_{2,1} & h_{2,2} & \dots & h_{2,k} \\ & & \ddots & \ddots & \vdots \\ & & & h_{k,k-1} & h_{k,k} \\ & & & & h_{k+1,k} \end{bmatrix}.$$

- Arnoldiho algoritmus počítá **postupně po sloupcích QR rozklad** matice $[v_1, Av_1, \dots, Av_k]$, která na počátku výpočtu není známa. Tato matice vzniká postupně po sloupcích.

Arnoldiho algoritmus

a vlastní čísla matice A

- Vektory počítané Arnoldiho algoritmem splňují

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

a pro $k = d(A, v)$ je $h_{d+1,d} = 0$,

$$AV_d = V_d H_d.$$

- Nechť platí $H_d y = \mu y$, $y \neq 0$, potom

$$\underbrace{A V_d y}_x = V_d H_d y = \mu \underbrace{V_d y}_x.$$

- **Množina vlastních čísel** H_d je **podmnožinou** vlastních čísel A , vlastní vektory A lze spočítat z vlastních vektorů H_d .
- A regulární $\Rightarrow H_d$ regulární.

Arnoldiho algoritmus

a aproximace vlastních čísel matice A

- **Pro** $k < d$ je $h_{k,k+1} \neq 0$. Vlastní čísla a vektory H_k poskytují pouze **aproximace vlastních čísel** A . Přenásobením vztahu y (vlastní vektor H_k) dostáváme

$$\begin{aligned} AV_k y &= V_k H_k y + h_{k+1,k} v_{k+1} e_k^T y \\ &= \mu V_k y + h_{k+1,k} v_{k+1} e_k^T y. \end{aligned}$$

- Jinak řečeno, vektor $x \equiv V_k y$ splňuje

$$Ax - \mu x = h_{k+1,k} v_{k+1} e_k^T y$$

tj. reziduum $Ax - \mu x$ je násobkem v_{k+1} ,

$$\|Ax - \mu x\| = h_{k+1,k} |e_k^T y|.$$

Arnoldiho metoda

- **Arnoldiho metoda** na výpočet aproximací vlastních čísel **využívá Arnoldiho algoritmus** k výpočtu **matice** H_k . Poté spočte vlastní čísla a vlastní vektory y matice H_k , $V_k y$ je pak aproximace vlastního vektoru A .
- Pro obecnou nesymetrickou matici je problematické říci cokoliv o **blízkosti spočtené aproximace** vlastního čísla μ k nejbližšímu vlastnímu číslu matice A , víme jen

$$\|Ax - \mu x\| = h_{k+1,k} |e_k^T y|.$$

- Vlastní čísla $H_k \rightarrow$ **Ritzova čísla**,
vektory $V_k y \rightarrow$ **Ritzovy vektory**.

Arnoldiho algoritmus

aplikovaný na symetrickou matici

- Arnoldiho algoritmus na **symetrickou** regulární A ,

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

- Přenásobme maticí V_k^T , ortogonalita sloupců, symetrie,

$$V_k^T AV_k = H_k,$$

$$H_k = V_k^T AV_k = V_k^T A^T V_k = (V_k^T AV_k)^T = H_k^T.$$

- H_k je symetrická, **horní Hessenbergova** $\Rightarrow$ **tridiagonální**,

$$H_k = \begin{bmatrix} \alpha_1 & \beta_2 & & & \\ \beta_2 & \alpha_2 & \beta_3 & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & \beta_k \\ & & & \beta_k & \alpha_k \end{bmatrix} \equiv T_k.$$

Lanczosův algoritmus

- Platí

$$AV_k = V_k T_k + \beta_{k+1} v_{k+1} e_k^T.$$

Rozepišme po sloupcích,

$$\beta_{j+1} v_{j+1} = Av_j - \alpha_j v_j - \beta_j v_{j-1}, \quad j = 1, \dots, k.$$

- **Ortogonalní bázi** Krylovova prostoru lze pro symetrickou matici A počítat **tříčlennou rekurencí**.
- β_{j+1} je normalizační koeficient (vždy kladný), α_j určíme z podmínky $v_{j+1} \perp v_j$, platí

$$\alpha_j = \langle Av_j, v_j \rangle.$$

Ortogonalita k ostatním vektorům báze je **automaticky** splněna **díky symetrii** matice A .

Lanczos's algorithm

```
input  $A, v$   
 $v_0 := 0$   
 $v_1 := v / \|v\|$   
 $\beta_1 := 0$   
for  $k = 1, 2, \dots$  do  
     $w := Av_k - \beta_k v_{k-1}$   
     $\alpha_k := v_k^* w$   
     $w := w - \alpha_k v_k$   
     $\beta_{k+1} := \|w\|$   
     $v_{k+1} := w / \beta_{k+1}$   
end for
```

Lanczosova metoda

na výpočet aproximací vlastních čísel

- Lanczosova metoda = Arnoldiho metoda pro symetrickou A .
- Nechť $k < d$ a nechť (μ, y) je vlastní pár T_k , $T_k y = \mu y$,

$$x \equiv V_k y.$$

Protože volíme y tak, aby $\|y\| = 1$, je i $\|x\| = 1$.

- Z $AV_k = V_k T_k + \beta_{k+1} v_{k+1} e_k^T$ plyne

$$Ax = \mu x + \beta_{k+1} v_{k+1} e_k^T y$$

a pro normu rezidua platí

$$\|Ax - \mu x\| = \beta_{k+1} |e_k^T y|.$$

- **Díky symetrii** matice A lze tento vztah **dále analyzovat**.

Analýza Lanczosovy metody

Blížkost aproximace

- A je symetrická $\rightarrow A = U\Lambda U^T$, kde U je ortogonální a Λ je diagonální reálná,

$$\|Ax - \mu x\| = \|U\Lambda U^T x - \mu x\| = \|\Lambda U^T x - \mu U^T x\|.$$

- Označme $w \equiv U^T x$, platí

$$\begin{aligned}\|Ax - \mu x\| &= \|(\Lambda - \mu I)w\| \\ &\geq \min_{\lambda_j} |\lambda_j - \mu| \|w\| \\ &= \min_{\lambda_j} |\lambda_j - \mu| \|x\|.\end{aligned}$$

- Jinak řečeno,

$$\min_{\lambda_j} |\lambda_j - \mu| \leq \frac{\|Ax - \mu x\|}{\|x\|} = \beta_{k+1} \frac{|e_k^T y|}{\|x\|}.$$

Analýza Lanczosovy metody

Přesná versus konečná aritmetika

- **Pomocí** čísla

$$\beta_{k+1} |e_k^T y|$$

Ize kontrolovat kvalitu příslušné **aproximace** μ .

- Při odvození odhadu jsme použili **vlastnost**

$$1 = \|y\| = \sqrt{y^T V_k^T V_k y} = \|x\|,$$

neplatí obecně **při počítání v konečné aritmetice**, ztráta ortogonality mezi sloupci V_k může způsobit $\|y\| \gg \|x\|$.

- **Christopher C. Paige** → **netriviální analýza**, ukázal že

$$\min_{\lambda_j} |\lambda_j - \mu| \leq \beta_{k+1} |e_k^T y|$$

platí i pro vypočtené hodnoty v konečné aritmetice (až na možnou malou nepřesnost $\sim \mathbf{u}$), navzdory ztrátě ortogonality a velikosti $\|x\|$.

Jacobiho matice

v Lanczosově algoritmu

- Matice

$$T_k = \begin{bmatrix} \alpha_1 & \beta_2 & & & \\ \beta_2 & \alpha_2 & \beta_3 & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & \beta_k \\ & & & \beta_k & \alpha_k \end{bmatrix},$$

kde $\alpha_i \in \mathbb{R}$, $i = 1, \dots, k$ a $\beta_i > 0$, $i = 2, \dots, k$ nazveme **Jacobiho maticí**.

- Tyto matice se v průběhu minulého století staly středem zájmu díky tomu, že se vyskytují v obecné teorii **ortogonálních polynomů** a **Gaussovy kvadratury**.
- Výskyt Jacobiho matic v Lanczosově algoritmu naznačuje **možný blízký vztah** mezi tímto algoritmem a zmíněnou teorií.

Některé vlastnosti Jacobiho matic

- **Vlastní čísla** Jacobiho matic se **ostře prokládají**.
- Všechna **leží** pro $k < d$ v otevřeném **intervalu** (λ_1, λ_n) .
- Pro $k < d$ jsou **vlastní čísla** T_k **různá od vlastních čísel** A .

Cvičení

7.5 Uvažujme charakteristické polynomy Jacobiho matic J_k ,

$$\chi_0(\lambda) \equiv 1, \quad \chi_k(\lambda) = \det(\lambda I - J_k), \quad k = 1, 2, \dots$$

Ukažte, že platí $\chi_0 = 1$, $\chi_1 = \lambda - \alpha_1$,

$$\chi_k(\lambda) = (\lambda - \alpha_k)\chi_{k-1}(\lambda) - \beta_k^2\chi_{k-2}(\lambda), \quad k = 2, 3, \dots$$

7.7 Ukažte pomocí předchozí rekurence, že dvě po sobě jdoucí Jacobiho matice nemohou mít stejná vlastní čísla.

7.8 Dokažte následující vztah mezi v_k generovanými Lanczosovým algoritmem s počátečním vektorem v a charakteristickými polynomy $\chi_k(\lambda)$ odpovídajících Jacobiho matic:

$$v_k = \frac{1}{\beta_1\beta_2\cdots\beta_k}\chi_{k-1}(A)v, \quad k = 1, 2, \dots,$$

kde $\chi_0 = 1$.

Cvičení

- Necht y je vlastní vektor matice T_k , $k < d$. Ukažte, že jeho poslední složka je nenulová, tj. platí $e_k^T y \neq 0$.

Důsledek: Pro $k < d$ je

$$\|Ax - \mu x\| = \beta_{k+1} |e_k^T y| \neq 0.$$

tj. μ není vlastním číslem matice A .