Numerické metody lineární algebry

1 Úvod

1.1 Úlohy lineární algebry

1. Řešení soustav lineárních rovnic $\mathbf{A}\vec{x} = \vec{b}$
   - Řešení soustavy s regulární čtvercovou maticí $\mathbf{A}$ řádu $n \times n$ pro 1 nebo více pravých stran
   - Výpočet inverzní matice $\mathbf{A}^{-1}$
   - Výpočet determinantu
   - Řešení soustav n rovnic o m neznámých a singulárních n $\times$ n soustav v nějakém definovaném smyslu (1 řešení + báze nulprostoru, řešení s nejmenší normou, řešení ve smyslu nejmenších čtverců)
2. Hledání vlastních čísel a vlastních vektorů
   - Úplný problém vlastních čísel
   - Částečný problém vlastních čísel

Standardní numerické knihovny - LINPACK (speciální pro řešení soustav lineárních rovnic), EISPACK (speciální pro vlastní čísla a vektory), NAG (obecná), IMSL (obecná)

1.2 Některé základní pojmy a označení

Vektorem zde budeme rozumět sloupcový vektor, horní index T označuje transponovaný vektor (nebo matici), matici označujeme tučným písmem

$$\vec{x} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} = (x_1, x_2, \dots, x_n)^T \qquad \mathbf{A} = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1m} \\ a_{21} & a_{22} & \dots & a_{2m} \\ \vdots & \vdots & \vdots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nm} \end{pmatrix} = (a_{ij})$$

Pokud nebude uvedeno jinak, budeme uvažovat reálné matice a vektory.

Mírou velikosti vektorů a matic je jejich vektorová norma. Vektorová norma musí splnit 3 podmínky a) $\|\vec{x}\| \geq 0$ a $\|\vec{x}\| = 0 \Leftrightarrow \vec{x} = \vec{0}$ b) $\|\lambda\vec{x}\| = |\lambda|\|\vec{x}\|$ a c) $\|\vec{x} + \vec{y}\| \leq \|\vec{x}\| + \|\vec{y}\|$.

Příklady vektorových norem

- maximová - $\|x\|_I = \max\limits_{i=1,\dots,n} |x_i|$
- součtová - $\|x\|_{II} = \sum\limits_{i=1}^{n} |x_i|$
- Eukleidovská - $\|x\|_{III} = \sqrt{\sum\limits_{i=1}^{n} x_i^2}$

Maticová norma

Vektorová norma matice se nazývá maticovou normou, pokud navíc platí pro $\forall$ matice $\mathbf{A}$, $\mathbf{B}$ podmínka 4) pro součin matic

$$\|\mathbf{A} \cdot \mathbf{B}\| \leq \|\mathbf{A}\| \cdot \|\mathbf{B}\|$$

Maticová norma je souhlasná s vektorovou normou, pokud pro $\forall$ $\mathbf{A}$, $\vec{x}$ platí

$$\|\mathbf{A}\vec{x}\| \leq \|\mathbf{A}\| \cdot \|\vec{x}\|$$

Příklady maticových norem

- řádková - $\|\mathbf{A}\|_I = \max\limits_{i=1,\dots,n} \sum\limits_{j=1}^{n} |a_{ij}|$
- sloupcová - $\|\mathbf{A}\|_{II} = \max\limits_{j=1,\dots,n} \sum\limits_{i=1}^{n} |a_{ij}|$
- Eukleidovská - $\|\mathbf{A}\|_{III} = \sqrt{\sum\limits_{i=1}^{n} \sum\limits_{j=1}^{n} a_{ij}^2}$

Každá z uvedených maticových norem je souhlasná se stejně označenou vektorovou normou.

1.3 Metody řešení soustav lineárních rovnic

1. přímé (finitní)
2. iterační
3. gradientní

2 Přímé metody řešení soustav lineárních rovnic

Přímé metody řešení spočívají v úpravě matice na trojúhelníkový (případně diagonální) tvar (přímý běh) a potom řešením soustavy s horní trojúhelníkovou maticí $\mathbf{U}$ nebo dolní trojúhelníkovou maticí $\mathbf{L}$ (zpětný běh). Zpětný běh je podstatně rychlejší než přímý.

2.1 Řešení soustav s trojúhelníkovou maticí

Rovnice s horní trojúhelníkovou maticí $\mathbf{U}$ se řeší postupným prováděním vzorce ve směru klesajícího indexu k

$$x_k = \frac{1}{u_{kk}} \left(b_k - \sum_{j=k+1}^{n} u_{kj} x_j \right)$$

K výpočtu libovolného x_k je třeba nejvýše n vnitřních cyklů (1 násobení + 1 sčítaní), počet operací roste tedy $\sim n^2$ (přesněji $\simeq 0.5 \cdot n^2$ vnitřních cyklů).

2.2 Gaussova a Gauss-Jordanova eliminace

Řeším soustavu rovnic $\mathbf{A}\vec{x} = \vec{b}$. V prvním kroku nechť prvek $a_{11} \neq 0$ (lze vždy dosáhnout přehozením rovnic). Prvek a_{11}, použitý k úpravě rovnic 2, ..., n nazveme <u>hlavním prvkem</u> (pivot).

Od i-té rovnice odečteme 1.rovnici násobenou multiplikátorem $m_i^{(1)} = -a_{i1}/a_{11}$. Modifikovaná soustava bude mít v 1.sloupci pod diagonálou samé 0. Úprava prováděná současně s pravou stranou odpovídá násobení rovnice maticí

$$\mathbf{D_1} = \begin{pmatrix} 1 & 0 & 0 & \dots & 0 \\ -a_{21}/a_{11} & 1 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ -a_{n1}/a_{11} & 0 & 0 & \dots & 1 \end{pmatrix}$$

Rovnice po první úpravě má tvar $\mathbf{D_1}\mathbf{A}\vec{x} = \mathbf{D_1}\vec{b}$, označíme $\mathbf{D_1}\mathbf{A} \equiv \mathbf{A}^{(1)}$ a $\mathbf{D_1}\vec{b} \equiv \vec{b}^{(1)}$.

Po $k-1$ úpravách má matice $\mathbf{A}^{(k-1)}$ tvar

$$\mathbf{A}^{(k-1)} = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1,k-1} & a_{1k} & \dots & a_{1n} \\ 0 & a_{22}^{(1)} & \dots & a_{2,k-1}^{(1)} & a_{2k}^{(1)} & \dots & a_{2n}^{(1)} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & a_{k-1,k-1}^{(k-2)} & a_{k-1,k}^{(k-2)} & \dots & a_{k-1,n}^{(k-2)} \\ 0 & 0 & \dots & 0 & a_{kk}^{(k-1)} & \dots & a_{k,n}^{(k-1)} \\ 0 & 0 & \dots & 0 & a_{k+1,k}^{(k-1)} & \dots & a_{k+1,n}^{(k-1)} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & 0 & a_{nk}^{(k-1)} & \dots & a_{n,n}^{(k-1)} \end{pmatrix}$$

Zde horní index značí počet úprav daného prvku.

Pokud $a_{kk}^{(k-1)} \neq 0$, lze ho zvolit za <u>hlavní prvek</u>, spočítat multiplikátory $m_i^{(k)} = -a_{ik}^{(k-1)}/a_{kk}^{(k-1)}$ pro $i = k+1, \dots,\ n$ a upravit příslušné rovnice.

V k-tém kroku úpravy používám jako hlavní prvek prvek $k-1$-krát upravený (odečítání!) hlavní prvek $\rightarrow$ ztráta přesnosti $\Rightarrow$ výběr hlavního prvku.

Bez výběru hlavního prvku - přímé metody nepoužitelné pro obecné matice!!

Počet operací

Na každou 0 $\rightarrow \leq n$ vnitřních cyklů, potřebuji $n(n-1)$ prvků $\rightarrow$ 0. Celkový počet vnitřních cyklů $\sim n^3$ (přesněji $\simeq 1/3\, n^3$), složitost algoritmu je řádu n^3.

Gauss-Jordanova eliminace

Upravují se všechny prvky mimo diagonálu. Matice se převede na jednotkovou $\mathbf{I}$. Přímo spočtu inverzní matici $\mathbf{A}^{-1}$. Vyšší počet operací $\simeq n^3$ vnitřních cyklů.

2.3 Výběr hlavního prvku (pivoting)

V $\forall$ kroku přímého běhu $\rightarrow$ výběr hlavního prvku.

- *Úplný* výběr hlavního prvku - v celé dosud neupravené části matice, pomalý, strategie - $\max |a_{ij}|$
- *Částečný* výběr hlavního prvku - v daném sloupci (*sloupcový*) nebo řádku (*řádkový*), rychlejší, vylepšená strategie např. sloupcového výběru - při výběru porovnávám velikosti prvků v daném sloupci normované na maximum absolutních hodnot prvků v daném řádku původní matice.

Výběr hlavního prvku $\rightarrow$ použitelnost přímých metod pro většinu matic.
Pro obecné velké matice ($N > 50$) nutná dvojitá přesnost. I tak často problémy u velkých špatně podmíněných matic !!

Problémy:

1. Singulární matice
2. Singulární matice vzniklá ztrátou přesnosti při úpravách
3. Ztráta přesnosti

2.4 LU metoda

$\forall$ matice $\mathbf{A}$ lze rozložit do tvaru $\mathbf{A} = \mathbf{L} \cdot \mathbf{U}$, kde $\mathbf{L}$, $\mathbf{U}$ jsou levá dolní, resp. pravá horní trojúhelníkové matice. Potom řešení najdu postupným řešením 2 soustav s trojúhelníkovou maticí

$$\mathbf{A}\vec{x} = \vec{b} \;\Rightarrow\; (\mathbf{L}\mathbf{U})\vec{x} = \vec{b} \;\Rightarrow\; \mathbf{L}\underbrace{(\mathbf{U}\vec{x})}_{\vec{y}} = \vec{b} \;\Rightarrow\; \mathbf{L}\vec{y} = \vec{b},\; \mathbf{U}\vec{x} = \vec{y}$$

LU dekompozice

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \dots & a_{2n} \\ a_{31} & a_{32} & a_{33} & \dots & a_{3n} \\ \dots & \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & a_{n3} & \dots & a_{nn} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & \dots & 0 \\ l_{21} & 1 & 0 & \dots & 0 \\ l_{31} & l_{32} & 1 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ l_{n1} & l_{n2} & l_{n3} & \dots & 1 \end{pmatrix} \begin{pmatrix} u_{11} & u_{12} & u_{13} & \dots & u_{1n} \\ 0 & u_{22} & u_{23} & \dots & u_{2n} \\ 0 & 0 & u_{33} & \dots & u_{3n} \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & u_{nn} \end{pmatrix}$$

Násobení matic

$$i \;\le\; j \qquad\qquad a_{ij} = u_{ij} + \sum_{k=1}^{i-1} l_{ik}u_{kj}$$

$$i \;>\; j \qquad\qquad a_{ij} = l_{ij}u_{jj} + \sum_{k=1}^{j-1} l_{ik}u_{kj}$$

Croutův algoritmus - postupný výpočet např. odleva po sloupcích a ve sloupcích odshora. Nejdříve

$$u_{ij} = a_{ij} - \sum_{k=1}^{i-1} l_{ik}u_{kj} \qquad\qquad i = 1, \dots, j$$

užívá l z předchozích sloupců a u z předchozích řádků, a potom

$$l_{ij} = \left(a_{ij} - \sum_{k=1}^{j-1} l_{ik}u_{kj}\right)/u_{jj} \qquad i = j+1, \dots, n$$

užívá l z předchozích sloupců a u z naddiagonální části sloupce.
Sloupcové hledání hlavního prvku (úplné nelze)
Prvky a_{ij} použiji jen 1 $\times$, výsledné prvky matic $\mathbf{L}$ a $\mathbf{U}$ se vejdou do 1 matice.
Vlastnosti LU metody:

- Přímá (finitní) metoda, stejně kroků jako přímý běh Gaussovy eliminace
- Hlavní výhoda - při dekompozici nepracuji s pravou stranou rovnice, rychlé výpočty pro postupně získávané pravé strany
- Lze iterativně zpřesnit výsledek

2.5 Iterativní zpřesnění řešení

Hledáme řešení $\vec{x}$ lineární rovnice $\mathbf{A}\vec{x} = \vec{b}$. Získáme nepřesné řešení $\vec{\tilde{x}}$

$$\vec{\tilde{x}} = \vec{x} + \vec{\delta x} \Rightarrow \mathbf{A}(\vec{x} + \vec{\delta x}) = \vec{b} + \vec{\delta b} \Rightarrow \mathbf{A}\vec{\delta x} = \vec{\delta b} = \mathbf{A}\vec{\tilde{x}} - \vec{b}\vec{x} = \vec{\tilde{x}} - \vec{\delta x}$$

Označíme $\vec{x}_0$ nepřesné řešení systému v prvním kroku $\mathbf{A}\vec{x}_0 \simeq \vec{b}$ a pak provádíme iteraci

$$\vec{x}_{i+1} = \vec{x}_i + (\vec{\delta x})_i \qquad\qquad \mathbf{A}(\vec{\delta x})_i = \vec{b} - \mathbf{A}\vec{x}_i$$

2.6 Podmíněnost řešení soustavy lineárních rovnic

Vzhledem k chybám vstupních údajů a zaokrouhlení místo $\mathbf{A}\vec{x} = \vec{b}$ řešíme úlohu

$$(\mathbf{A} + \Delta\mathbf{A})\,(\vec{x} + \Delta\vec{x}) = \vec{b} + \Delta\vec{b}$$

Nejdříve případ $\Delta\mathbf{A} = 0$

$$\begin{aligned} \Delta\vec{x} = \mathbf{A}^{-1}\Delta\vec{b} \;&\Rightarrow\; \|\Delta\vec{x}\| \le \|\mathbf{A}^{-1}\| \cdot \|\Delta\vec{b}\| \\ \mathbf{A}\vec{x} = \vec{b} \;&\Rightarrow\; \|\vec{x}\| \ge \|\vec{b}\|/\|\mathbf{A}\| \end{aligned}$$

Pro relativní chybu řešení tedy platí

$$\frac{\|\Delta\vec{x}\|}{\|\vec{x}\|} \le \|\mathbf{A}\| \cdot \|\mathbf{A}^{-1}\| \cdot \frac{\|\Delta\vec{b}\|}{\|\vec{b}\|}$$

Číslo $C_p = \|\mathbf{A}\| \cdot \|\mathbf{A}^{-1}\|$ se nazývá podmíněnost matice.
Pokud je i $\Delta\mathbf{A} \ne 0$, pak

$$\frac{\|\Delta\vec{x}\|}{\|\vec{x}\|} \le C_p \frac{\frac{\|\Delta\mathbf{A}\|}{\|\mathbf{A}\|} + \frac{\|\Delta\vec{b}\|}{\|\vec{b}\|}}{1 - C_p\frac{\|\Delta\mathbf{A}\|}{\|\mathbf{A}\|}}$$

Pro $C_p \gg 1$ je soustava špatně podmíněná a malé chyby vstupních dat i malé zaokrouhlovací chyby ve výpočtu se projeví velkou chybou řešení.

2.7 Výpočet inverzní matice a determinantu

Gauss-Jordanova eliminace vypočte přímo inverzní matici. U Gaussovy eliminace a LU dekompozice získáme inverzní matici řešením pro n pravých stran, tvořených vektory standardní báze.
Všechny metody rovnocenné jak přesností, tak i počtem n^3 operací.
Determinant nepočítáme ze vzorců (růst zaokrouhlovací chyby), ale využijeme věty, že determinant součinu matic je roven součinu determinantů. Pro LU dekompozici

$$\det(\mathbf{A}) = \det(\mathbf{L}) \cdot \det(\mathbf{U}) = \prod_{j=1}^{n} u_{jj}$$

Gaussova eliminace odpovídá násobení maticemi a pokud se sám řádek nenásobí a pouze se přičítají násobky jiných řádků, je determinant matice každé úpravy $\mathbf{D}_i = 1$ a determinant $\mathbf{A}$ se rovná součinu diagonálních prvků trojúhelníkové matice.
Pokud dochází k přehazování řádků při výběru hlavních prvků, dojde k násobení determinantu -1, je nutno si zapamatovat počet změn znaménka.

2.8 Speciální typy matic

<u>Řídká matice</u> má většinu prvků $= 0$.
Pro řešení soustav s řídkou maticí se často používají <u>gradientní metody</u>, spočívající v minimalizaci $\|\mathbf{A}\vec{x} - \vec{b}\|^2_{III}$. Pro řídkou matici je totiž počet operací pro výpočet $\mathbf{A}\vec{x} \sim n$, a ne $\sim n^2$ jako pro plnou matici.
Matice $\mathbf{A}$ je <u>pásová</u>, pokud $a_{ij} = 0$ pro $|i - j| > p$. Tridiadonální matice pro $p = 1$, pětidiagonální matice pro $p = 2$.

<u>Soustavy s tridiagonální maticí</u>

$$\begin{pmatrix} a_1 & b_1 & 0 & 0 & \dots & 0 & 0 & 0 \\ c_2 & a_2 & b_2 & 0 & \dots & 0 & 0 & 0 \\ 0 & c_3 & a_3 & b_3 & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & c_{n-1} & a_{n-1} & b_{n-1} \\ 0 & 0 & 0 & 0 & \dots & 0 & c_n & a_n \end{pmatrix} \cdot \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_{n-1} \\ x_n \end{pmatrix} = \begin{pmatrix} f_1 \\ f_2 \\ f_3 \\ \vdots \\ f_{n-1} \\ f_n \end{pmatrix}$$

Tridiagonální zapíšeme do 3 vektorů $\vec{a}$, $\vec{b}$, $\vec{c}$. V praxi téměř vždy tridiagonální matice, u kterých výběr hlavního prvku není potřebný (silně regulární matice).

Řešení: Předpokládáme zpětný běh $x_k = \mu_k x_{k+1} + \rho_k$. Dosadíme

$$c_i \left(\mu_{i-1} x_i + \rho_{i-1} \right) + a_i x_i + b_i x_{i+1} = f_i$$

po úpravě

$$x_i = \frac{-b_i}{c_i \mu_{i-1} + a_i} \; x_{i+1} + \frac{f_i - c_i \rho_{i-1}}{c_i \mu_{i-1} + a_i}$$

výsledek

$$\mu_i = \frac{-b_i}{c_i \mu_{i-1} + a_i} \qquad\qquad \rho_i = \frac{f_i - c_i \rho_{i-1}}{c_i \mu_{i-1} + a_i}$$

Startování $c_1 = 0,\ b_n = 0$ ($\mu_0,\ \rho_0,\ x_{n+1}$) libovolné.

Blokově tridiagonální matice - $\mathbf{A}_i,\ \mathbf{B}_i,\ \mathbf{C}_i$ - malé matice $\Rightarrow$ $\boldsymbol{\mu_i},\ \boldsymbol{\rho_i}$ - malé matice

2.9 Úlohy se žádným řešením nebo ∞ řešeními

m rovnic o n neznámých, lineárně závislé $n \times n$ systémy.

Metoda SVD (singular value decomposition) - pokud $\mathbf{A}\vec{x} = \vec{b}$ má ∞ řešení, určí řešení s nejmenší Eukleidovskou normou a bázi nulprostoru, pokud $\neg\exists$ řešení, najde řešení ve smyslu nejmenších čtverců - vektor $\vec{x}$ minimalizující $\|\mathbf{A}\vec{x} - \vec{b}\|_{III}$.

SVD - $\mathbf{A},\ \mathbf{U}$ matice $m \times n$, $\mathbf{W},\ \mathbf{V},\ \mathbf{I}$ matice $n \times n$, $\mathbf{W}$ diagonální, $\mathbf{I}$ jednotková, $\mathbf{U},\ \mathbf{V}$ ortogonální ($\mathbf{U}^T \cdot \mathbf{U} = \mathbf{V}^T \cdot \mathbf{V} = \mathbf{I}$)

$$\mathbf{A} = \mathbf{U} \cdot \mathbf{W} \cdot \mathbf{V}^T \quad \Rightarrow \quad \vec{x} = \mathbf{V} \cdot [\mathrm{diag}(1/w_j)] \cdot \mathbf{U}^T \cdot \vec{b}$$

Pokud $w_j = 0$ ($w_j \simeq 0$) nahradíme $1/w_j \to 0$ (signalizuje singulárnost matice).

2.10 Rychlost řešení soustav lineárních rovnic (matice N x N) přímými metodami

- Gauss–Jordanova eliminace
  Při výpočtu se provádí $\sim N^3$ vnitřních cyklů (v každém cyklu jedno násobení a jedno sčítání)

- Gaussova eliminace a LU dekompozice
  Přímý běh je podstatně náročnější na počet operací. U obou metod je k výpočtu přímého běhu potřeba $\sim \frac{1}{3} N^3$ vnitřních cyklů. Při zpětném běhu $\sim \frac{1}{2} N(N-1)$ v Gaussově eliminaci. Při LU dekompozici jsou zapotřebí 2 zpětně běhy, ale přímý běh je nepatrně kratší, protože se neupravuje pravá strana, operací je u Gaussovy eliminace i LU dekompozice stejně. Pro výpočet inverzní matice jsou pracnosti Gauss–Jordanovy eliminace, Gaussovy eliminace i LU dekompozice stejné.

- Metoda řádu < 3.
  Byla dokázána (Strassen) existence metody, kde počet operací $\sim N^{\log_2 7}$ ($\log_2 7 \simeq 2.807$) a tedy roste s dimenzí N matice pomaleji než u klasických metod, kde počet operací $\sim N^3$, Tato metoda vyžaduje komplikovanou průběžnou archivaci napočítaných hodnot. Pro malé matice je tato metoda podstatně pomalejší než klasické metody a její výhody se projeví až matice řádu $N \gg 1000$.

3 Gradientní metody

Lineární rovnici $\mathbf{A}\vec{x} = \vec{b}$ řešíme například minimalizací funkce
$f(\vec{x}) = \frac{1}{2}|\mathbf{A}\vec{x} - \vec{b}|^2$. V každém kroku λ takové, aby $f(\vec{x} + \lambda\vec{u})$ bylo minimální. Tedy

$$\lambda = \frac{-\vec{u}\nabla f}{|\mathbf{A}\vec{u}|^2} \quad \text{kde} \quad \nabla f(\vec{x}) = \mathbf{A}^T(\mathbf{A}\vec{x} - \vec{b})$$

Pro řídké matice se složitost násobení vektoru maticí snižuje z počtu operací $\sim N^2$ na počet operací $\sim N$.

<u>*Pozn.*</u> Existuje řada moderních často používaných gradientních metod.

4 Iterační metody řešení soustav lineárních rovnic

4.1 Některé typy matic

Matice $\mathbf{A}$ typu $n \times n$ je <u>**diagonálně dominantní**</u>, pokud

$$|a_{ii}| > \sum_{j=1, j\neq i}^{n} |a_{ij}| \qquad \forall i = 1,\ 2, \ldots,\ n$$

Symetrická matice $\mathbf{A}$ typu $n \times n$ je <u>**pozitivně definitní**</u>, pokud pro $\forall \vec{x} \neq \vec{0}$ je skalární součin $(\vec{x}, \mathbf{A}\vec{x}) > 0$.

4.2 Iterační proces

Řešení $\vec{x}$ lineární rovnice $\mathbf{A}\vec{x} = \vec{b}$ odhadneme vektorem $\vec{x}^{(0)}$ a další přiblížení k přesnému řešení vypočteme pomocí předpisu

$$\vec{x}^{(k+1)} = \mathbf{B}_k \vec{x}^{(k)} + \vec{c}^{(k)}$$

Pro řešení $\vec{x}$ musí platit

$$\vec{x} = \mathbf{B}_k \vec{x} + \vec{c}_k.$$

Odtud vyplývá $\vec{x}^{(k+1)} - \vec{x} = \mathbf{B}_k(\vec{x}^{(k)} - \vec{x}) = \mathbf{B}_k \mathbf{B}_{k-1}(\vec{x}^{(k-1)} - \vec{x}) = \ldots$.
Pro konvergenci iteračního procesu je tedy nutné a stačí, aby

$$\lim_{k\to\infty} \mathbf{B}_k \mathbf{B}_{k-1} \ldots \mathbf{B}_0 = \mathbf{0}.$$

Iterační metody dělíme na metody stacionární, které mají matici $\mathbf{B}$ konstantní ($\mathbf{B}_k = \mathbf{B}$), a na metody nestacionární.

4.3 Příklad nestacionární iterační metody

Příkladem nestacionární iterační metody je **řízená relaxace**. Pro jednoduchost ji ukážeme pro matici $\mathbf{A}$, která má diagonální členy jednotkové ($\mathbf{a}_{ii} = 1$). Nechť po k-té iteraci je ze složek rezidua $|\mathbf{A}\vec{x} - \vec{b}|$ maximální i-tá složka. Budeme tedy nulovat i-tou složku rezidua $k+1$-ní iterací

$$x_i^{(k+1)} = b_i - a_{i1}x_1^{(k)} - \ldots - a_{ii-1}x_{i-1}^{(k)} - a_{ii+1}x_{i+1}^{(k)} - \ldots - a_{in}x_n^{(k)}$$

Matice $\mathbf{B}_k$ a vektor $\vec{c}_k$ mají tvar

$$\mathbf{B}_k = \begin{pmatrix} 1 & & & & & & \\ & \ddots & & & & & \\ & & 1 & & & & \\ -a_{i1} & \dots & -a_{ii-1} & 0 & -a_{ii+1} & \dots & -a_{in} \\ & & & & 1 & & \\ & & & & & \ddots & \\ & & & & & & 1 \end{pmatrix}, \qquad \vec{c}_k = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ b_i \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

Tato metoda se však nehodí pro využití na počítači, protože vyžaduje v každém kroku hledání rovnice, kde odchylka od řešení je maximální, a to je časově náročné.

4.4 Stacionární iterační metody

Pro všechna vlastní čísla λ matice $\mathbf{B}$ (*tedy čísla, pro která existuje takový* $\vec{v}$, *že platí* $\mathbf{B}\vec{v} = \lambda\vec{v}$) musí platit $|\lambda| < 1$.

<u>Věta</u> Nutnou a postačující podmínkou konvergence metody je podmínka, aby spektrální poloměr $\varrho(\mathbf{B})$ byl

$$\varrho(\mathbf{B}) = \max_{i=1,\dots,n} |\lambda_i| < 1$$

<u>Věta</u> Pokud v některé maticové normě platí

$$\|\mathbf{B}\| < 1 \ ,$$

pak iterace konverguje.

<u>Odhad chyby</u> Chceme dosáhnout přesnosti ε a tedy iterovat, dokud nebude

$$\|\vec{x}^{(k)} - \vec{x}\| \leq \varepsilon \ .$$

Výraz $\|\vec{x}^{(k)} - \vec{x}\|$ lze odhadnout

$$\|\vec{x}^{(k)} - \vec{x}\| = \|\vec{x}^{(k)} - \mathbf{A}^{-1}\vec{b}\| = \|\mathbf{A}^{-1}(\mathbf{A}\vec{x}^{(k)} - \vec{b})\| \leq \|\mathbf{A}^{-1}\| \, \|\mathbf{A}\vec{x}^{(k)} - \vec{b}\|$$

4.5 Prostá iterace

Lineární rovnici $\mathbf{A}\vec{x} = \vec{b}$ převedeme na tvar

$$\vec{x} = (\mathbf{I} - \mathbf{A})\vec{x} + \vec{b} ,$$

kde $\mathbf{I}$ je jednotková matice.
Prostá iterace je tedy dána iteračním vzorcem

$$\vec{x}^{(k+1)} = (\mathbf{I} - \mathbf{A})\vec{x}^{(k)} + \vec{b} .$$

Označíme $\mathbf{B} = \mathbf{I} - \mathbf{A}$, pak lze přesnost k-té iterace odhadnout výrazem

$$\|\vec{x}^{(k)} - \vec{x}\| \leq \|\mathbf{B}\|^k \left[\|\vec{x}^{(0)}\| + \frac{\|\vec{b}\|}{1 - \|\mathbf{B}\|} \right]$$

Metoda prosté iterace se pro systémy lineárních rovnic prakticky nepoužívá.

4.6 Jacobiho metoda

Předpokladem metody je, že matice $\mathbf{A}$ má nenulové diagonální prvky $a_{ii} \neq 0$.
Složky $k+1$ Jacobiho iterace řešení jsou dány

$$x_i^{(k+1)} = -\frac{a_{i1}}{a_{ii}} x_1^{(k)} - \ldots - \frac{a_{ii-1}}{a_{ii}} x_{i-1}^{(k)} - \frac{a_{ii+1}}{a_{ii}} x_{i+1}^{(k)} - \ldots - \frac{a_{in}}{a_{ii}} x_n^{(k)} + \frac{b_i}{a_{ii}}$$

Matici $\mathbf{A}$ lze zapsat ve tvaru

$$\mathbf{A} = \mathbf{D} + \mathbf{L} + \mathbf{R} ,$$

kde $\mathbf{D}$ je diagonální, $\mathbf{L}$ je dolní trojúhelníková a $\mathbf{R}$ horní trojúhelníkovou matici ($\mathbf{L}$ a $\mathbf{R}$ mají nulovou diagonálu).

Jacobiho iteraci lze zapsat ve tvaru

$$\vec{x}^{(k+1)} = -\mathbf{D}^{-1}(\mathbf{L} + \mathbf{R})\vec{x}^{(k)} + \mathbf{D}^{-1}\vec{b}$$

Věta Pokud je matice $\mathbf{A}$ diagonálně dominantní, pak Jacobiho metoda konverguje.
Důkaz Pro diagonálně dominantní matici platí

$$\sum_{j=1, j\neq i}^{n} \frac{|a_{ij}|}{|a_{ii}|} < 1$$

a tedy maximová norma matice $\|\mathbf{D}^{-1}(\mathbf{L} + \mathbf{R})\| < 1$.

4.7 Gauss–Seidelova metoda

Gauss–Seidelova metoda je podobná Jacobiho metodě, ale na rozdíl od ní používá při výpočtu složek vektoru $x_i^{(k+1)}$ již dříve vypočtené složky $k+1$ iterace. Iterace je dána vztahem

$$x_i^{(k+1)} = -\frac{a_{i1}}{a_{ii}} x_1^{(k+1)} - \ldots - \frac{a_{ii-1}}{a_{ii}} x_{i-1}^{(k+1)} - \frac{a_{ii+1}}{a_{ii}} x_{i+1}^{(k)} - \ldots - \frac{a_{in}}{a_{ii}} x_n^{(k)} + \frac{b_i}{a_{ii}}$$

Vztah lze zapsat vektorově

$$\vec{x}^{(k+1)} = -(\mathbf{D}+\mathbf{L})^{-1}\mathbf{R}\vec{x}^{(k)} + (\mathbf{D}+\mathbf{L})^{-1}\vec{b}$$

Věta Pro konvergenci Gauss–Seidelovy metody stačí, když platí libovolná z následujících dvou podmínek:

1. $\mathbf{A}$ je diagonálně dominantní matice.
2. Matice $\mathbf{A}$ je pozitivně definitní (symetrická s kladnými vlastními čísly).

4.8 Superrelaxační metoda

Gauss–Seidelovy metoda konverguje pro poměrně širokou třídu matic, ale její konvergence může být velmi pomalá. Označíme-li $\Delta x_i^{(k)} = x_i^{(k+1)} - x_i^{(k)}$ rozdíl mezi iteracemi Gauss–Seidelovou metodou, pak je superrelaxační metoda dána vztahem

$$x_i^{(k+1)} = x_i^{(k)} + \omega \Delta x_i^{(k)}$$

kde relaxační faktor ω je z intervalu $(0,2)$, obvykle $\omega \in \langle 1, 2)$.
Relaxační faktor slouží k urychlení konvergence metody a jeho optimální hodnotu lze vypočítat ze vztahu

$$\omega_{opt} = \frac{2}{1+\sqrt{1-\varrho^2(\mathbf{B})}} \qquad \mathbf{B} = -(\mathbf{D}+\mathbf{L})^{-1}\mathbf{R}$$

$\mathbf{B}$ je iterační matice Gauss–Seidelovy metody. Gauss–Seidelova metoda je tedy speciálním případem superrelaxační metody s $\omega = 1$.

5 Hledání vlastních čísel a vektorů

5.1 Úvod

Nechť pro číslo λ $\exists \vec{x} \neq \vec{0}$ takový, že $\mathbf{A}\vec{x} = \lambda\vec{x}$. Pak λ je vlastní číslo a vektor $\vec{x}$ je vlastní vektor matice $\mathbf{A}$.

2 typy úloh

1. Úplný problém vlastních čísel – hledání všech vlastních čísel matice a popřípadě i příslušných vlastních vektorů
2. Částečný problém vlastních čísel – hledání 1 nebo několika vlastních čísel (obvykle největších)

Charakteristický polynom matice $\mathbf{A}$ – determinant $\det(\mathbf{A} - \lambda\mathbf{I})$.

Pozn. Je-li $\mathbf{A}$ matice $n \times n$, je charakteristický polynom n-tého stupně, a tedy má n kořenů (mohou být i vícenásobné). Ke každému vlastnímu číslu $\exists$ alespoň 1 vlastní vektor. Počet l lineárně nezávislých (LN) vlastních vektorů je $l \leq k$ (k je násobnost vlastního čísla).

Pozn. Matice defektní má $< n$ lineárně nezávislých vlastních vektorů. Příklad

$$\mathbf{A} = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}, \qquad \det(\mathbf{A} - \lambda\mathbf{I}) = (1-\lambda)^2 = 0, \quad \text{tedy } \lambda_{1,2} = 1.$$

Vektor $\vec{x} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$ je jediným vlastním vektorem $\mathbf{A}$.

Pozn. Reálná matice může mít komplexně sdružená vlastní čísla a vlastní vektory. Například

$$\mathbf{A} = \begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix}, \qquad \det(\mathbf{A} - \lambda\mathbf{I}) = (1-\lambda)^2 + 1 = 0, \quad \text{tedy } \lambda_{1,2} = 1 \pm i.$$

Vlastní vektory jsou $\vec{x}_1 = \begin{pmatrix} 1 \\ -i \end{pmatrix}$ a $\vec{x}_2 = \begin{pmatrix} 1 \\ i \end{pmatrix}$.

Normální matice $\mathbf{A}$ je taková, že $\mathbf{A}^{\mathrm{T}}\mathbf{A} = \mathbf{A}\mathbf{A}^{\mathrm{T}}$. Normální matice řádu n má n LN vlastních vektorů.

Pozn. Symetrická matice ($\mathbf{A} = \mathbf{A}^{\mathrm{T}}$) má všechna vlastní čísla reálná.

Pozn. Trojúhelníkové matice mají všechna vlastní čísla na diagonále.

Věta Podobné matice $\mathbf{A}$ a $\mathbf{P}^{-1}\mathbf{A}\mathbf{P}$ mají stejná vlastní čísla (stejné spektrum).
Odvození

$$\begin{aligned}\det(\mathbf{P}^{-1}\mathbf{A}\mathbf{P} - \lambda\mathbf{I}) &= \det[\mathbf{P}^{-1}(\mathbf{A} - \lambda\mathbf{I})\mathbf{P}] = \det(\mathbf{P}^{-1})\det(\mathbf{A} - \lambda\mathbf{I}) \cdot \\ &\cdot \det(\mathbf{P}) = \det(\mathbf{A} - \lambda\mathbf{I})\end{aligned}$$

Pokud je vektor $\vec{x}$ vlastním vektorem matice $\mathbf{A}$, pak vektor $\mathbf{P}^{-1}\vec{x}$ je vlastním vektorem matice $\mathbf{P}^{-1}\mathbf{A}\mathbf{P}$.

Věta Ke každá matici $\exists$ jí podobná matice v Jordanově normálním tvaru

$$\mathbf{J} = \begin{pmatrix} \mathbf{J}_1 & 0 & \dots & 0 \\ 0 & \mathbf{J}_2 & & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \dots & \mathbf{J}_s \end{pmatrix}, \quad \text{kde} \quad \mathbf{J}_i = \begin{pmatrix} \lambda & 0 & 0 & \dots & 0 \\ 1 & \lambda & 0 & & 0 \\ 0 & 1 & \lambda & & 0 \\ \vdots & & & \ddots & \vdots \\ 0 & 0 & 0 & \dots & \lambda \end{pmatrix}.$$

Pozn. Pro $\forall$ normální matici $\exists$ podobná matice diagonální.

Pozn. Neexistuje finitní postup jak provést transformaci matice na Jordanův normální tvar.

Numerické metody řešení úplného problému vlastních čísel

1. Sekvence elementárních transformací $\longrightarrow$ na přibližně diagonální tvar (příp. Jordanův normální tvar) nebo přibližně speciální typ (např. tridiagonální nebo Hessenbergova matice).

2. Rozložíme matici $\mathbf{A}$ na součin dvou matic $\mathbf{A} = \mathbf{F}_L \cdot \mathbf{F}_R$. Matice $\tilde{\mathbf{A}} = \mathbf{F}_R\mathbf{F}_L$ je podobná matici $\mathbf{A}$.
   Odvození $\mathbf{F}_R\mathbf{F}_L = \mathbf{F}_L^{-1}\mathbf{F}_L\mathbf{F}_R\mathbf{F}_L = \mathbf{F}_L^{-1}\mathbf{A}\mathbf{F}_L$.

5.2 Jacobiho transformace (metoda)

Jacobiho metoda nalezne $\forall$ vlastní čísla a vektory **symetrické** matice. Cílem je postupně zmenšovat čísla mimo diagonálu.

V každém kroku nalezneme v absolutní hodnotě maximální číslo mimo diagonálu $|a_{pq}| = \max_{i,j<i} |a_{ij}|$. Pootočíme osy p, q tak, aby se matice $[a_{pp}\ a_{pq};\ a_{qp}\ a_{qq}]$ transformovala na diagonální.

Jeden (k-tý) krok iterace Jacobiho metodou má tvar

$$\mathbf{A}^{(k)} = \mathbf{T}_{p,q}^{\mathrm{T}} \mathbf{A}^{(k-1)} \mathbf{T}_{p,q}, \qquad \text{kde} \qquad \mathbf{T}_{p,q} = \begin{bmatrix} 1 & & & & & & 0 \\ & \ddots & & & & & \\ & & c & \dots & -s & & \\ & & \vdots & 1 & \vdots & & \\ & & s & \dots & c & & \\ & & & & & \ddots & \\ 0 & & & & & & 1 \end{bmatrix}.$$

V této matici platí $c = \cos\varphi$ a $s = \sin\varphi$. Po k-té iterace je prvek matice

$$a_{pq}^{(k)} = (c^2 - s^2) a_{pq}^{(k-1)} - cs(a_{pp}^{(k-1)} - a_{qq}^{(k-1)})$$

Aby se prvek $a_{pq}^{(k)} = 0$, úhel φ musí splňovat vztah

$$\mathrm{tg} 2\varphi = \frac{2a_{pq}}{a_{pp} - a_{qq}}$$

Důkaz konvergence Při posloupnosti Jacobiho transformací nejsou 0 mimo diagonálu trvalé. Důkaz konvergence je založen na konvergenci sumy mimodiagonálních prvků k 0.

Označme $t(\mathbf{A}) = \sum_{i,j=1;i\neq j}^{n} a_{ij}^2$. Pak

$$t(\mathbf{A}^{(k)}) = t(\mathbf{A}^{(k-1)}) - 2a_{pq}^2 \qquad \vee \qquad a_{pq}^2 \geq \frac{t(\mathbf{A}^{(k-1)})}{n(n-1)}$$

Tedy

$$t(\mathbf{A}^{(k)}) \leq \left(1 - \frac{2}{n(n-1)}\right) t(\mathbf{A}^{(k-1)}) \leq \left(1 - \frac{2}{n(n-1)}\right)^k t(\mathbf{A})$$

Posloupnost je tedy shora odhadnuta geometrickou posloupnosti s kvocientem < 1.

5.3 LU rozklad pro úplný problém vlastních čísel

Tato metoda konverguje velmi pomalu, na výpočet je třeba velký počet operací.

Matice $\mathbf{A}$ je 0-tým krokem iterace, tj. $\mathbf{A}_0 \equiv \mathbf{A}$. V k-tém kroku rozložíme matici $\mathbf{A}_k = \mathbf{L}_k\mathbf{U}_k$. Vytvoříme matici $\mathbf{A}_{k+1} = \mathbf{U}_k\mathbf{L}_k$, ta je podobná matici $\mathbf{A}_k$.

Pokud posloupnost $\mathbf{B}_k = \mathbf{L}_0\mathbf{L}_1\dots\mathbf{L}_k \to$ k regulární matici $\Rightarrow$ matice $\mathbf{A}_k \to$ k horní trojúhelníkové matici, která má na diagonále vlastní čísla.

Existují ale speciální rozklady matic vhodné pro hledání vlastních čísel a vektorů. Pro tyto rozklady konverguje obdobný postup rychle.

5.4 Částečný problém vlastních čísel

Hledáme vlastní číslo největší v absolutní hodnotě.
Zvolíme libovolný vektor $\vec{x}^{(0)}$. A dále počítáme iterace

$$\vec{x}^{(k+1)} = \frac{1}{\varrho_k}\mathbf{A}\vec{x}^{(k)}\ , \quad \text{kde} \quad \varrho_k = \vec{e}_1^{\,\mathrm{T}}\mathbf{A}\vec{x}^{(k)} \quad (\text{ příp.} \quad \varrho_k = \|\mathbf{A}\vec{x}^{(k)}\|)$$

Pak platí

$$\lim_{k\to\infty} \varrho_k = \lambda_1 \quad \vee \quad \lim_{k\to\infty} \vec{x}^{(k)} = \vec{x}_1$$

Pokud chceme další vlastní číslo, redukujeme matici na řád $(n-1)$. Je-li vektor $\vec{x}_1 = (u_1, \dots, u_n)^{\mathrm{T}}$, platí

$$\mathbf{P} = \begin{pmatrix} u_1 & 0 & \dots & 0 \\ u_2 & 1 & & 0 \\ \vdots & & \ddots & \vdots \\ u_n & 0 & \dots & 1 \end{pmatrix}, \qquad \mathbf{P}^{-1}\mathbf{A}\mathbf{P} = \begin{pmatrix} \lambda_1 & \vec{q}^{\,\mathrm{T}} \\ \vec{0} & \mathbf{B} \end{pmatrix}.$$

dále hledáme maximální vlastní číslo matice $\mathbf{B}$.

Pozn. Nevýhodou je postupná ztráta přesnosti.

Pozn. Pro výpočet nejmenšího vlastního čísla lze nalézt největší vlastní číslo matice $\mathbf{A}^{-1}$. Pro hledání vlastního čísla v určité oblasti lze provést posun, neboť $(\mathbf{A} + \mu\mathbf{I})\vec{x} = (\lambda + \mu)\vec{x}$.