# Lineární algebra III – 7. přednáška 2.12.2013

## Numerický výpočet vlastních čísel obecné matice, QR metoda

Existují různé metody pro výpočet vlastních čísel:
- přímé (málo iterací, obecně pro řádu matice),
- nepřímé iterativní (hodně iterací, využívají specifické vlastnosti daných matic, víc Zdeněk Strakoš příště).

Protože výpočet vlastních čísel je ekvivalentní s hledáním kořenů polynomu, teoreticky každá metoda je iterativní.

Řešíme problém: Obecná nesymetrická matice $A$, chceme nalézt její vlastní čísla a podprostory vl. vektorů.

Zaokrouhlování: Lze udělat podobné odhady, tedy existuje číslo podmíněnosti $k_\lambda(A)$, problematické jsou matice s $k_\lambda(A) = +\infty$ a $\frac{1}{k_\lambda(A)}$ je vzdálenost od nejbližší problem. matice. Nechť $\lambda$ má násobnost jedna, s jednotkovým pravým vlastním vektorem $\mathbf{x}$ a levým vlastním vektorem $\mathbf{y}$.

Potom pro perturbaci $A + \delta A$ a $\lambda + \delta\lambda$ platí:

$$|\delta\lambda| \leq \frac{\|\delta A\|}{|y^T x|} + O(\|\delta A\|^2), \quad \text{tedy} \quad k_\lambda(A) = \frac{1}{|y^T x|}.$$

Secans úhlu mezi $x$ a $y$.

Například pro symetrickou matici je $x = y$, a tedy $k_\lambda(A) = 1$.
Lze provést výrazně složitější analýzu, více například v knize Applied Numerical Linear Algebra.

Jordanova dekompozice: Je nespojitá v koeficientech matice
Například:

$$\begin{pmatrix} 0 & 1 & & & \\ & 0 & 1 & & \\ & & 0 & 1 & \\ & & & 0 & 1 \\ & & & & 0 \end{pmatrix} \text{ a } \begin{pmatrix} \varepsilon & 1 & & & \\ & 2\varepsilon & 1 & & \\ & & 3\varepsilon & 1 & \\ & & & 4\varepsilon & 1 \\ & & & & 5\varepsilon \end{pmatrix}$$

s Jordanovými rozklady

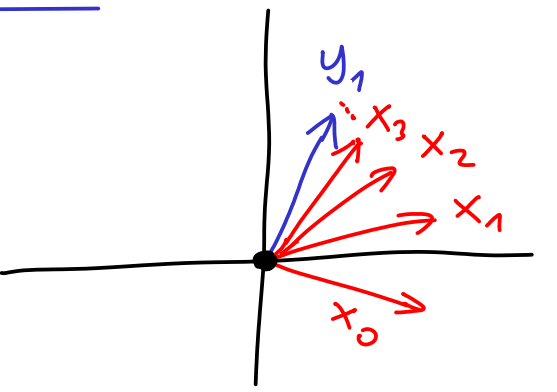
$$\begin{pmatrix} 0 & 1 & & & \\ & 0 & 1 & & \\ & & 0 & 1 & \\ & & & 0 & 1 \\ & & & & 0 \end{pmatrix} \not\sim \begin{pmatrix} \varepsilon & & & & \\ & 2\varepsilon & & & \\ & & 3\varepsilon & & \\ & & & 4\varepsilon & \\ & & & & 5\varepsilon \end{pmatrix}.$$

Proto se numericky používá Schurův rozklad, který je spojitý v koeficientech:

$$A = QUQ^T.$$

# 32 První nápad na algoritmus: Mocninná metoda.

Nechť $y_1, \dots, y_n$ jsou vlastní vektory,
$|\lambda_1| \geq \dots \geq |\lambda_n|$ příslušná vlastní čísla.




## Algoritmus:

Zvolit libovolný vektor $x_0$.

Iterovat $k = 1, \dots$

$$\left[\begin{array}{l} x_k = \dfrac{A x_{k-1}}{\|A x_{k-1}\|}. \leftarrow \text{k-tá approx. } y_1. \\ \tilde{\lambda}_k = x_k^T A x_k. \leftarrow \text{k-tá approx. } \lambda_1. \end{array}\right.$$

Normování zařídí, že konvergujeme k $y_1$.

Předpoklady: $\alpha_1 \neq 0$ a $|\lambda_2| < |\lambda_1|$.

## Příklad:

V přesné aritmetice

$$x_k = A^k x_0 / \|A^k x_0\|.$$

(Normujeme kvůli nepřetečení.)

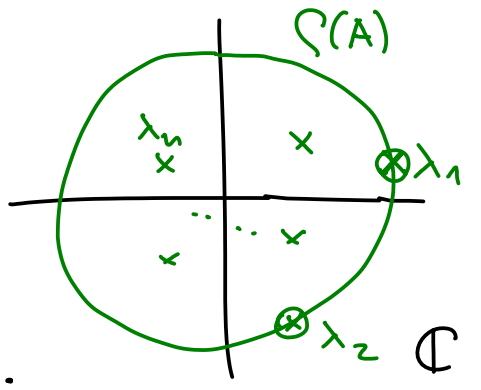
Pokud $x_0 = \alpha_1 y_1 + \dots + \alpha_n y_n$,

$$A^k x_0 = \alpha_1 \lambda_1^k y_1 + \dots + \alpha_n \lambda_n^k y_n =$$

$$= \alpha_1 \lambda_1^k \left( y_1 + \frac{\alpha_2 \lambda_2^k}{\alpha_1 \lambda_1^k} y_2 + \dots + \frac{\alpha_n \lambda_n^k}{\alpha_1 \lambda_1^k} y_n \right) \to \alpha_1 \lambda_1^k y_1.$$

Rychlost konvergence je lineární, s koeficientem $\dfrac{|\lambda_2|}{|\lambda_1|}$.

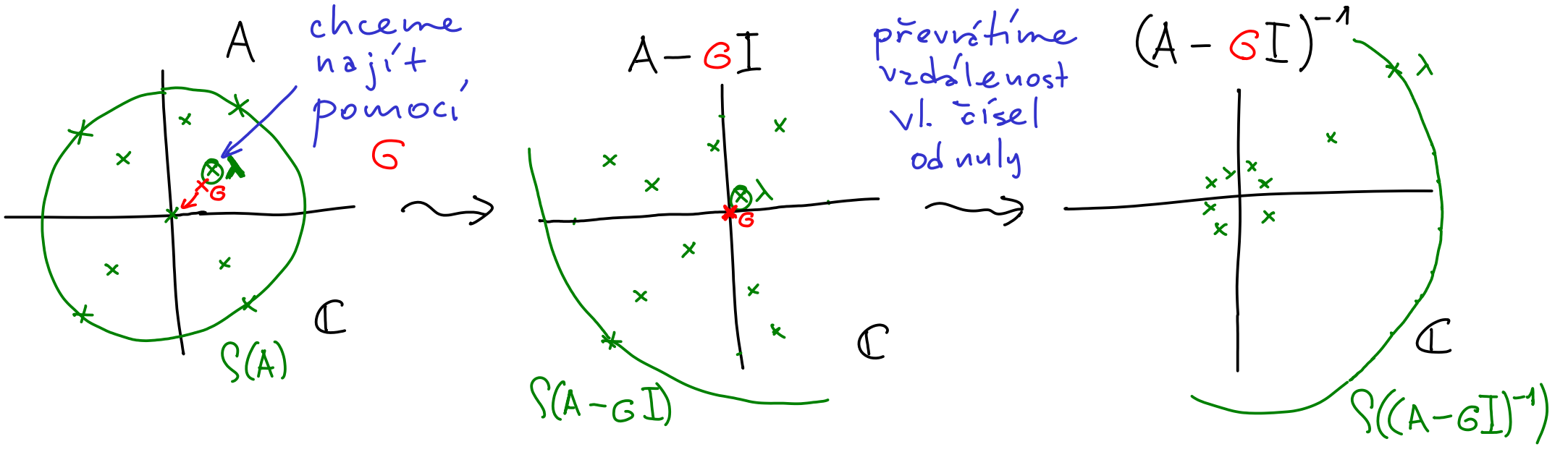
Potíž: Konvergence je moc pomalá. Ohledně předpokladů, $\alpha_1 \neq 0$ platí skoro vždy, ale $|\lambda_2| = |\lambda_1|$ často.
Dále předpokládáme, že $A$ je diagonalizovatelná.
A umíme nalézt pouze největší vlastní číslo $\lambda_1$.




Inverzní mocninná metoda řeší tyto problémy částečně.
$A + \sigma I$ posouvá vlastní čísla o $\sigma$, $A^{-1}$ má převrácená vlastní čísla.
Pokud máme odhad $\sigma$, jak vypadá vlastní číslo zhruba, aplikujeme mocninnou metodu $(A - \sigma I)^{-1}$, konvergujeme velice rychle. Klíčové je: Matice $A$ a $(A - \sigma I)^{-1}$ mají stejné vlastní vektory, rychlost konvergence je $|\lambda - \sigma| / |\lambda' - \sigma|$.

↑ Druhé nejbližší vlastní číslo k $\sigma$.

Ale: musíme znát $\sigma$, což typicky neznáme.



Typicky se používají heuristiky pro volbu $\sigma$, více později.

Důležitá aplikace: Můžeme znát vlastní číslo $\lambda$, třeba z nějaké jiné metody jako metoda bisekce pro symetrické matice, ale nemusíme znát příslušný vlastní vektor $x$. Potom ho inverzní iterace dokáže enormně rychle najít.

Ortogonální iterace: najde více vlastních vektorů a čísel.

Iterujeme několik vektorů. Aby všechny nekonvergovali k $y_1$, udržujeme je ortogonálně. V maticovém zápisu:

Nechť je $Z_0$ libovolná ortogonální matice $n \times p$.

Iteruj $k = 1, \dots$ (iterujeme vektory matic)

- $Y_k = A Z_{k-1}$.
- QR rozklad $Y_k = Z_k R_k$. (ortogonalizace)

Sloupcové vektory $Z_k$ konvergují k vlastním vektorům $p$ největších vlastních čísel, analýza se udělá podobně. Pokud $p=n$ a vlastní čísla mají různé absolutní hodnoty, potom $Z_k^T A Z_k$ konverguje k Schurově rozkladu.

QR metoda – spojení obou metod { ortogonální iterace / inverzní mocninné metody

$A_0 = A$.

iteruj $k = 0, \dots$

- QR rozklad $A_k = Q_k R_k$.
- $A_{k+1} = R_k Q_k$. ← prohození matic

Metoda vypadá magicky, ale ...

Matice $A_{k+1}$ a $A_k$ jsou podobné:

$$A_{k+1} = R_k Q_k = \underbrace{Q_k^T Q_k}_{id} R_k Q_k = Q_k^T A_k Q_k.$$

Lemma: Platí $A_k = Z_k^T A Z_k$, kde $Z_k$ je z ortogonální iterace.

Důkaz: Indukcí, zjevně pravda na začátku. Nechť $A_k = Z_k^T A Z_k$.

Pak $A Z_k = Z_{k+1} R_{k+1}$, tedy $Z_k^T A Z_k = \underbrace{Z_k^T Z_{k+1}}_{Q} \underbrace{R_{k+1}}_{R}$. To je

součin ortogonální matice a horní trojúhelníkové, tedy QR dekompozice, jednoznačně určené až na násobení sloupců $Q$ a řádků $R$ $-1$.

34

Potom ale $Z_{k+1}^T A Z_{k+1} = (Z_{k+1}^T A Z_k)(Z_k^T Z_{k+1}) = RQ = A_{k+1}$. □

($Z_k Z_k^T$ … $R$, úpravou $AZ_k = Z_{k+1}R$; $Q$)

Lze přidat posunutí $G_k$ do algoritmu:

QR rozklad $A_k - G_k I = Q_k R_k$. ← Platí podobné vlastnosti jako předtím.

$A_{k+1} = R_k Q_k + G_k I$.

☼ Když $G_k$ je přesně vlastní číslo, potom $R_k$ je singulární.

Kdyby $(R_k)_{n,n} = 0$, potom $A_{k+1} = \begin{pmatrix} A'_{k+1} & * \\ 0 \cdots 0 & G_k \end{pmatrix}$, tedy $G_k$ je vlastní číslo a dále pokračujeme s menší $A'_{k+1}$.

Konvergujeme při malých hodnotách.

Platí, že jedna iterace QR metody odpovídá inverzní iteraci na $((A_k - G_k)^T)^{-1}$. (Důkaz v Applied Numerical Linear Algebra.)

Dobrá heuristika je volba $G_k = (A_k)_{n,n}$, vyjde kvadratická konvergence.

Pár poznámek k praktické implementace QR metody:

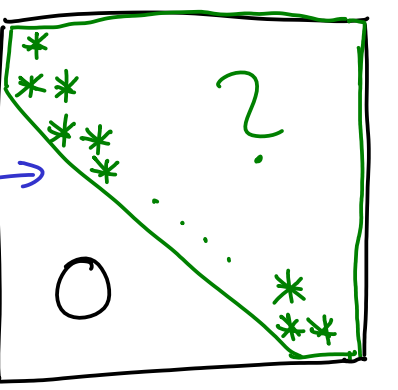
Zcela ignorujeme, že vlastní čísla mohou být komplexní.

Zrovna tak nevíme, co znamé konvergence při dostatečně malých hodnotách v posledním řádku, bez odhadu chyby perturbace.

Kdybychom implementovali přímočaře, složitost by byla zhruba $O(n^4)$ (pro konstantně mnoho iterací na vlastní číslo), chceme $O(n^3)$.

Ke zrychlení se kroky aplikují implicitně, navíc převedeme matici $A$ do horní Hessenbergovy formy. Dále pro symetrickou matici dostaneme dokonce tridiagonální matici.

Horní Hessenbergova:




Nulová v dolním trojúhelníku vyjma první diagonály.

Lemma: Existuje ortogonální $Q$, že $H = QAQ^T$ je horní Hessenbergova.

Důkaz: Aplikujeme $n-1$ Householderových reflexí podobně jako u QR, ale z obou stran. Proto nemůžeme vynulovat tolik.

$Q_1 \cdot A = \begin{pmatrix} * & ? \\ * & \\ 0 & \\ \vdots & \\ 0 & \end{pmatrix} \Rightarrow Q_1 A Q_1^T = \begin{pmatrix} * & ? \\ * & \\ 0 & \\ \vdots & \\ 0 & \end{pmatrix}$.

Transpozice mění první sloupec.

A tak dál, postupně nulujeme sloupce… □