Lineární algebra - iterační metody

Numerické metody

7. dubna 2018

FJFI ČVUT v Praze

Úvod
Rozdělení
Metody

Zastavení
SOR
Programy

Úvod

Úvod - LAR

Mějme základní úlohu

$$\mathbf{A}\vec{x} = \vec{b}, \tag{1}$$

kde $\mathbf{A} \in \mathbb{R}^{n,n}$ je regulární matice, $\vec{b} \in \mathbb{R}^n$ je vektor pravé strany a $\vec{x} \in \mathbb{R}^n$ je hledaný vektor řešení.

Provedeme nyní ekvivaletní úpravy

$$\begin{aligned}
\mathbf{A}\vec{x} = \vec{b} \quad &\Leftrightarrow \quad \vec{0} = \vec{b} - \mathbf{A}\vec{x} \quad \Leftrightarrow \quad \vec{0} = \vec{b} - \mathbf{A}\vec{x} - \vec{x} + \vec{x} \quad \Leftrightarrow \\
&\Leftrightarrow \quad \vec{x} = \vec{b} - \mathbf{A}\vec{x} + \vec{x} \quad \Leftrightarrow \quad \vec{x} = \vec{b} - \mathbf{A}\vec{x} + \mathbb{I}\vec{x} \quad \Leftrightarrow \\
&\Leftrightarrow \quad \vec{x} = \vec{b} + (\mathbb{I} - \mathbf{A})\,\vec{x}.
\end{aligned}$$

tedy $\vec{x}$, které splňuje (1), splňuje i tuto rovnici.

Úvod - Iterace

Na rovnici

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

není nic zajímavého, dokud nezadefinujeme rekuretní tvar

$$\vec{x}_k = \vec{b} + (\mathbb{I} - \mathbf{A})\,\vec{x}_{k-1}\,, \qquad \forall k \in \mathbb{N}$$

a počateční odhad $\vec{x}_0$.

Tato posloupnost vektorů, pak konverguje ke skutečnému řešení

$$\lim_{k\to\infty} \vec{x}_k = \vec{x}$$

za určitých podmínek kladených na matici $\mathbf{A}$.

Úvod - Prostá iterace

Metodě založené na rovnici

$$\vec{x}_k = \vec{b} + (\mathbb{I} - \mathbf{A})\,\vec{x}_{k-1}\,, \qquad \forall k \in \mathbb{N}$$

a počatečnímu odhadu $\vec{x}_0$ říkáme **metoda prosté iterace**.

Tato metoda je velice jednoduchá, ale konverguje velice pomalu a pro málo typů matic.

Rozdělení

Rozdělení - Princip

Nyní budeme uvažovat, že matici $\mathbf{A}$ rozdělíme na součet dvou libovolných matic $\mathbf{M}$ a $\mathbf{N}$, tedy

$$\mathbf{A} = \mathbf{M} + \mathbf{N}.$$

Úloha (1) přejde tedy do tvaru

$$\mathbf{M}\vec{x} = \vec{b} - \mathbf{N}\vec{x},$$

kde označíme $\vec{x}$ na pravé straně jako $\vec{x}_k$ a na levé $\vec{x}_{k-1}$, čímž dostaneme rekurentní předpis pro řešení

$$\mathbf{M}\vec{x}_k = \vec{b} - \mathbf{N}\vec{x}_{k-1}. \tag{2}$$

Abychom takovouto soustavu vyřešili, je nutné aby matice $\mathbf{M}$ byla regulární, to je také jediný požadavek na rozložení.

Rozdělení - Matice

Nyní se podíváme na základní rozklad matice $\mathbf{A}$.

Matici $\mathbf{A}$ rozdělíme na součet tří matic

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

kde $\mathbf{L}$ je dolní $\triangle$ matice (bez diagonály), $\mathbf{D}$ je diagonální a $\mathbf{U}$ je horní $\triangle$ matice (bez diagonály).

Jedná se pouze o součet matic (tedy ne jako u **LU dekompozice**, kde to byl součin), tedy pouze o vyjmutí prvků z matice $\mathbf{A}$.

Rozdělení - Matice

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

$$\mathbf{A} = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1(n-1)} & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2(n-1)} & a_{2n} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{(n-1)1} & a_{(n-1)2} & \cdots & a_{(n-1)(n-1)} & a_{(n-1)n} \\ a_{n1} & a_{n2} & \cdots & a_{n(n-1)} & a_{nn} \end{pmatrix}$$

$$\mathbf{L} = \begin{pmatrix} 0 & 0 & \cdots & 0 & 0 \\ a_{21} & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{(n-1)1} & a_{(n-1)2} & \cdots & 0 & 0 \\ a_{n1} & a_{n2} & \cdots & a_{n(n-1)} & 0 \end{pmatrix}$$

Rozdělení - Matice

$$\mathbf{D} = \begin{pmatrix} a_{11} & 0 & \cdots & 0 & 0 \\ 0 & a_{22} & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & a_{(n-1)(n-1)} & 0 \\ 0 & 0 & \cdots & 0 & a_{nn} \end{pmatrix}$$

$$\mathbf{U} = \begin{pmatrix} 0 & a_{12} & \cdots & a_{1(n-1)} & a_{1n} \\ 0 & 0 & \cdots & a_{2(n-1)} & a_{2n} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & a_{(n-1)n} \\ 0 & 0 & \cdots & 0 & 0 \end{pmatrix}$$

Rozdělení - Možnosti

Nabízí se nám tři možnosti (s ohledem na požadavek regulárnost), jak volit matici $\mathbf{M}$ a $\mathbf{N}$

1. $\mathbf{M} = \mathbf{D}$ a $\mathbf{N} = \mathbf{L} + \mathbf{U}$,
2. $\mathbf{M} = \mathbf{L} + \mathbf{D}$ a $\mathbf{N} = \mathbf{U}$,
3. $\mathbf{M} = \mathbf{D} + \mathbf{U}$ a $\mathbf{N} = \mathbf{L}$.

Metody

Metody - Jacobiho metoda

Pokud použijime první rozdělení, tedy $\mathbf{M} = \mathbf{D}$ a $\mathbf{N} = \mathbf{L} + \mathbf{U}$, získáme Jacobiho metodu.

Ta je výhodná především kvůli tomu, že jsme schopni jednoduše matici $\mathbf{M}$ invertovat, jelikož je to diagonální matice. Můžeme tedy rovnici (2) přepsat rovnou jako

$$\vec{x}_k = \mathbf{M}^{-1} \left(\vec{b} - \mathbf{N}\vec{x}_{k-1} \right),$$

resp.

$$\vec{x}_k = \mathbf{D}^{-1} \left(\vec{b} - (\mathbf{L} + \mathbf{U})\, \vec{x}_{k-1} \right).$$

Metody - Inverze diagonální matice

Diagonální matici $\mathbf{D}$ invertuje jednoduše tak, že na diagonálu umístíme převrácené hodnoty prvků, tedy

$$\mathbf{D} = \begin{pmatrix} d_{11} & 0 & \cdots & 0 \\ 0 & d_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & d_{nn} \end{pmatrix} \quad \Rightarrow \quad \mathbf{D}^{-1} = \begin{pmatrix} \frac{1}{d_{11}} & 0 & \cdots & 0 \\ 0 & \frac{1}{d_{22}} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \frac{1}{d_{nn}} \end{pmatrix}.$$

Metody - Gaussova-Seidelova metoda

Body 2 a 3 pokryjeme Gaussovou-Seidelovou metodou, jelikož jsou velice podobné. Matice $\mathbf{M}$ bude vždy trojúhelníková matice, na vyřešení soustavy

$$\mathbf{M}\vec{x}_k = \vec{b} - \mathbf{N}\vec{x}_{k-1},$$

tedy budeme potřebovat metody probírané minulou hodinu (přímé metody).

Matici $\mathbf{N}$ a vektor $\vec{x}_{k-1}$ známe a jejich násobek dá opět vektor, označíme si tedy pravou stranu jako $\vec{e} = \vec{b} - \mathbf{N}\vec{x}_{k-1}$.

Metody - Gaussova-Seidelova metoda

Potřebujeme tedy vyřešit soustavu

$$\mathbf{M}\vec{x}_k = \vec{e},$$

kde $\mathbf{M}$ je trojúhelníková. V klasické aritmetice bychom tuto matici invertovali a získali. V aritmetice s konečnou přesností ale raději použijeme zpětný běh, kvůli stabilitě.

Zastavení

Zastavení - Úvod

Jak jsem si řekli, vektor $\vec{x}_k$ konverguje k vektoru řešení $\vec{x}$. V numerice ale nemůžeme počítat do nekonečna a i kdybychom to dělali, tím že počítáme s konečnou přesností, nemuseli bychom se dostat k přesnému řešení.

Zastavení - Podmínky

Musíme tedy položit kritérium, kdy iterace zastavit. Nabízí se nám spousta možností

1. počet iterací
2. rozdíl po sobě jdoucích řešení $\vec{x}_k$ a $\vec{x}_{k-1}$
3. rozdíl po sobě jdoucích řešení $\vec{x}_k$ a $\vec{x}_{k-1}$ podělený $\vec{x}_k$
4. výpočet
$$||\mathbf{A}^{-1}|| \left|\left|\mathbf{A}\vec{x}_k - \vec{b}\right|\right|$$
5. rozdíl oproti skutečnému řešení, tedy $\vec{b} - \mathbf{A}\vec{x}_k$

Zastavení - Počet iterací

Jedná se velice jednoduchý postup, kdy zafixujeme počet opakování.

Nemáme ale odhad, jak přesný máme výsledek.

Navíc pro jednoduché úlohy může dojít zbytečným iteracím navíc, jelikož už nemůže získat lepší výsledek.

Zastavení - Rozdíl dvou řešení

Jedná se opět o jednoduchý zastavovací algoritmus.

Jeho myšlenka spočívá v tom, že pokud se dvě po sobě řešení neliší v normě více jak o ε (nějaké malé reálné číslo), tak může iterace zastavit.

Tedy dokud platí $||\vec{x}_k - \vec{x}_{k-1}|| > \varepsilon$, pak pokračuj s iteracemi.

Zastavení - Rozdíl dvou řešení

Problém nastává pro různě "velké" vektory. Neboť vektory

$$\vec{v}_1 = \begin{pmatrix} 100 \\ 100 \\ 100 \end{pmatrix}, \quad \vec{v}_2 = \begin{pmatrix} 99 \\ 99 \\ 99 \end{pmatrix}$$

a

$$\vec{v}_1 = \begin{pmatrix} 10 \\ 10 \\ 10 \end{pmatrix}, \quad \vec{v}_2 = \begin{pmatrix} 9 \\ 9 \\ 9 \end{pmatrix}$$

mají různou absolutní chybu ale stejnou relativní (tedy přenost). V prvním případě by bylo zapotřebí více iterací než u druhého (i když chceme stejně dobrý výsledek).

Zastavení - Rozdíl dvou řešení podělený

Tento neduh oproti pouhému rozdílu dvou řešení napravuje algoritmus, kde iterujeme dokud platí

$$\frac{||\vec{x}_k - \vec{x}_{k-1}||}{||\vec{x}_k||} > \varepsilon\,.$$

Díky tomu už nezáleží na velikosti (v normě) vektorů.

Zastavení - Pomocí vzorce

Pokud použijeme podmínku

$$\frac{||\mathbf{A}^{-1}||\left|\left|\mathbf{A}\vec{x}_k - \vec{b}\right|\right|}{||\vec{x}_k||} > \varepsilon\,,$$

tak dostaneme nejlepší možný odhad přesnosti.

Jedná se také o dobrý odhad chyby, ale jinak je to až moc náročné na výpočet.

Málo kdy se tedy používá (jelikož musíte výpočítat nebo odhadnout norma inverzní matice, což je ekvivalentní, jako úlohu vyřešit).

Zastavení - Rozdíl oproti skutečnému řešení

Označíme si

$$\vec{r} = \vec{b} - \mathbf{A}\vec{x}_k \,.$$

Existuje pak několik způsobů, kdy zastavit

1. $||\vec{r}|| < \varepsilon$,
2. $\frac{||\vec{r}||}{||\vec{b}||} < \varepsilon$,
3. $\frac{||\vec{r}||}{||\mathbf{A}||\cdot||\vec{x}_k||+||\vec{b}||} < \varepsilon$.

První podmínka ale trpí opět nepřizpůsobení škálování (tedy jako *Rozdíl dvou řešení*), další dva body jsou velice dobré na zastavování a jsou často používány.

SOR

SOR - Successive over-relaxation

Označíme v Gaussově-Seidelově metodě rozdíl po sobě jdoucích řešení

$$\Delta x_k = \vec{x}_k - \vec{x}_{k-1},$$

díky tomu, můžeme napsat

$$\vec{x}_k = \vec{x}_{k-1} + \Delta x_k \,.$$

U metody SOR *Successive over-relaxation* (tedy superrelaxační metoda) se toto delta násobí parametrem ω.

SOR - Postup

Napočítáme tedy $\vec{x}_k^{\mathrm{GS}}$ pomocí Gaussovy-Seidelovy metody a následně provedeme opravu a získáme tak nové řešení

$$\vec{x}_k = \vec{x}_{k-1} + \omega \left(\vec{x}_k^{\mathrm{GS}} - \vec{x}_{k-1} \right).$$

Parametr ω může nabývat hodnot $(0, 2\rangle$, jelikož by ale $\omega < 1$ zpomalovalo konvergenci, volí se $\omega \in \langle 1, 2\rangle$.

SOR - Optimální omega

Optimální ω pak odpovídá

$$\omega_{\mathrm{OPT}} = \frac{2}{1 + \sqrt{1 - \rho^2(\mathbf{M}^{-1}\mathbf{N})}},$$

kde $\rho(\mathbf{M}^{-1}\mathbf{N})$ je spektrální poloměr matice $\mathbf{M}^{-1}\mathbf{N}$ a matice $\mathbf{M}$ a $\mathbf{N}$ pochází z Gaussovy-Seidelovy metody.

Zjistit spektrální poloměr je ale po většinou stejně obtížné, jako vyřešit danou soustavu. Často se tedy řešení s neoptimálním ω pomocí odhadu spektrálního poloměru.

Programy

Programy - Základní programy

Spouštěcí program zde

1. Metoda prostých iteracích zde
2. Jacobiho metoda zde
3. Gaussova-Seidelova metoda zde
4. Superrelaxační metoda zde

+ zpětný běh pro dolní trojúhelníkovou matici zde.

Programy - Porovnání počtu iterací

Spouštěcí program zde a k němu lehce upravené funkce metod

1. Metoda prostých iteracích zde
2. Jacobiho metoda zde
3. Gaussova-Seidelova metoda zde
4. Superrelaxační metoda zde

+ zpětný běh pro dolní trojúhelníkovou matici zde.

Programy - Porovnání různých způsobů zastavování

Spouštěcí program zde a k němu upravená Gaussova-Seidelova metoda se zastavováním

1. podle počtu iterací zde
2. rozdílu posobě jdoucích řešení zde
3. rozdílu posobě jdoucích řešení podělné posledním řešením zde
4. rozdíl pravých stran zde
5. rozdíl pravých stran podělený pravou stranou zde

+ zpětný běh pro dolní trojúhelníkovou matici zde.