

Bakalářská práce WIP

Benjamin de Silva

October 2023

Úvod

Motivace

1 Definice (hlavní zdroj - linalg skriptu)

V této kapitole si zavedeme základní definice potřebné pro práci s maticemi, operace na nich a různé typy matic, které budeme používat.

Definice 1.1 (Matice). *Matice* typu $m \times n$ nad tělesem \mathbf{T} rozumíme obdelníkové schéma prvků z tělesa \mathbf{T} s m řádky a n sloupci. Matice typu $m \times m$ se nazývá *čtvercová matice řádu m* . Matice typu $m \times 1$ se nazývá *sloupcový aritmetický vektor* a matice typu $1 \times m$ se nazývá *řádkový aritmetický vektor*. Zápisem $A = (a_{ij})_{m \times n}$ rozumíme matici typu $m \times n$ s prvkem $a_{ij} \in T$ na pozici (i,j) .

V rámci této práce budeme uvažovat matice nad tělesem $T = \mathbb{C}$

Definice 1.2 (Typy matic). Nechť A je čtvercová matice řádu n . Pak řekneme, že

- A je *nulová*, pokud $a_{ij} = 0, \forall i, j \in \{1, \dots, n\}$,
- A je *jednotková*, pokud $a_{ij} = 0$ pro $i \neq j$ a $a_{ii} = 1$ jinak,
- A je *diagonální*, pokud $a_{ij} = 0$ pro $i \neq j$,
- A je *horní trojuhelníková*, pokud $a_{ij} = 0$ pro $i > j$,
- A je *dolní trojuhelníková*, pokud $a_{ij} = 0$ pro $i < j$,
- A je *permutační*, pokud má v každém řádku a sloupci právě jeden prvek 1 a zbytek jsou nuly,
- A je *symetrická*, pokud $a_{ij} = a_{ji}, \forall i, j \in \{1, \dots, n\}$,
- A je *řádká*, pokud většina prvků matice je nulových.

Značení. Budeme značit jednotkovou matici $\mathbf{I}_{m \times n}$ a nulovou matici $\mathbf{0}_{m \times n}$, jsou-li obě matice typu $m \times n$. Pokud jsou matice čtvercové řádu n , tak značíme $\mathbf{I}_n, \mathbf{0}_n$ a v případě, že je to z kontextu zřejmé, tak index může být vynechán.

Definice 1.3 (Sčítání matic). Nechť A, B jsou matice stejného typu $m \times n$, pak definujeme jejich součet jako matici

$$A + B = (a_{ij} + b_{ij})_{m \times n}$$

Nechť A, B, C jsou matice stejného typu, pak operace sčítání matic má následující vlastnosti

- $(A + B) + C = A + (B + C)$,
- $A + \mathbf{0} = A$,
- $A + (-A) = \mathbf{0}$,
- $A + B = B + A$.

Definice 1.4 (Násobení matice skalárem). Nechť A je matice typu $m \times n$ a $t \in \mathbb{R}$. Pak definujeme t -násobek matice A jako matici

$$t \cdot A = tA = (t \cdot a_{ij})_{m \times n}$$

Nechť A, B jsou matice stejného typu a $s, t \in \mathbb{R}$. Pak platí následující vlastnosti

- $s(tA) = (st)A$,
- $1A = A$,
- $-A = (-1)A$,
- $(s + t)A = sA + tA$,
- $s(A + B) = sA + sB$.

Definice 1.5 (Transponovaná matice). Nechť A je matice typu $m \times n$, pak transponovanou maticí k matici A rozumíme matici

$$A^T = (b_{ij})_{n \times m},$$

kde $b_{ij} = a_{ji}$ pro $i \in \{1, \dots, n\}, j \in \{1, \dots, m\}$.

Z definice symetrické matice a transponované matice plyne, že pro A symetrickou platí $A^T = A$.

Nechť A, B jsou matice stejného typu $m \times n$ a $s \in \mathbb{C}$, pak platí

- $(A^T)^T = A$
- $(A^T + B^T) = A^T + B^T$
- $(sA)^T = sA^T$

Značení. Někdy je vhodné nahlížet na matici jako na posloupnost sloupcových vektorů, tedy matici $A = (a_{ij})_{m \times n}$ můžeme zapsat jako $A = (\mathbf{a}_1 | \mathbf{a}_2 | \dots | \mathbf{a}_n)$, kde $\mathbf{a}_i = (a_{1i}, a_{2i}, \dots, a_{mi})^T, i \in \{1, \dots, n\}$ je i -tý m -složkový sloupcový aritmetický vektor matice A ,

Definice 1.6 (Součin matice a vektoru). Nechť $A = (\mathbf{a}_1 | \mathbf{a}_2 | \dots | \mathbf{a}_n)$ je matice typu $m \times n$ a $b = (b_1, b_2, \dots, b_n)^T$ je aritmetický sloupcový vektor. Pak definujeme součin matice A s vektorem b jako

$$A \cdot b = b_1 \mathbf{a}_1 + b_2 \mathbf{a}_2 + \dots + b_n \mathbf{a}_n,$$

což je m -složkový sloupcový vektor.

Definice 1.7 (Součin dvou matic). Nechť A je matice typu $m \times n$ a $B = (\mathbf{b}_1, \dots, \mathbf{b}_k)$ je matice typu $n \times k$. Pak součin matic A a B je

$$AB = (A\mathbf{b}_1, A\mathbf{b}_2, \dots, A\mathbf{b}_k),$$

což je matice typu $m \times k$.

Ze zavedených definic už můžeme začít řešit úlohu soustav lineárních rovnic. Mějme zadanou matici A typu $m \times n$ a $b = (\mathbf{b}_1, \dots, \mathbf{b}_m)^T$ m -složkový aritmetický vektor. Pak hledáme $x = (\mathbf{x}_1, \dots, \mathbf{x}_n)^T$ sloupcový vektor takový, že platí rovnost $A\mathbf{x} = \mathbf{b}$. Každý takový vektor \mathbf{x} nazveme řešením soustavy lineárních rovnic matice A s pravou stranou \mathbf{b} .

Definice 1.8 (Inverzní matice). Nechť A je matice typu $m \times n$. Pak

- matici X nazveme inverzní k A zprava, pokud platí $AX = I_m$
- matici X nazveme inverzní k A zleva, pokud platí $AX = I_n$
- je-li A čtvercová řádu n a matice X taková, že $AX = XA = I_n$, pak X nazveme inverzní k matici A , matici X značíme A^{-1} a matici A nazveme invertibilní

Definice 1.9 (Regulární matice). Matici A řádu n nazveme regulární, pokud je zobrazení $f_A : \mathbb{C}^n \rightarrow \mathbb{C}^n, x \mapsto Ax$ určené maticí A bijektivní. Matice A , která není regulární, se nazývá singulární.

Pro čtvercovou matici A platí následující ekvivalence

- Matice A je regulární,
- Zobrazení f_A je na
- Zobrazení f_A je prosté
- Soustava $A\mathbf{x} = 0$ má právě jedno řešení $\mathbf{x} = 0$
- Matice A je invertovatelná
- Existuje matice X tak, že $AX = I_n$
- Existuje matice Y tak, že $YA = I_n$
- $\text{rank}(A) = n$

Definice 1.10 (Obor hodnot, jádro, hodnost). Pro matici A typu $m \times n$ definujeme

$$\mathbf{R}(A) = \{A\mathbf{x} : \mathbf{x} \in \mathbb{R}^n\}, \quad \mathbf{N}(A) = \{\mathbf{x} \in \mathbb{C}^n\}, \quad \text{rank}(A) \equiv \dim(\mathbf{C}(A)),$$

kde $\mathbf{R}(A)$ nazveme *oborem hodnot* matice A , $\mathbf{N}(A)$ nazveme *jádrem* matice A a $\text{rank}(A)$ *hodností* matice A .

2 Metody pro hledání nulového prostoru (zdroj-kniha se stromem)

V této kapitole popíšeme různé metody pro hledání nulového prostoru matic. Mějme tedy nějakou matici A typu $m \times n$ řádu m , pak hledáme nějakou matici B typu $n \times (n - m)$ řádu $n - m$ tak, že platí $AB = 0$, tedy sloupce matice B tvoří bázi nulového prostoru matice A .

Pro každou metodu bude výhodné nejprve rozložit matici A a poté matici B spočítáme za použití získaného rozkladu. Abychom zachovali numerickou stabilitu a řidkost, tak je někdy nutné prohodit některé řádky a sloupce za pomoci permutačních matic, tedy ve skutečnosti hledáme rozklad matice P_1EP_2 .

2.1 Singulární rozklad

Singulární rozklad (zkráceně SVD - singular value decomposition) nám umožňuje určit hodnotu či normu matice, ortogonální bázi oboru hodnot a nulového prostoru matice, nicméně má vysoké výpočetní nároky. Před zavedením singulárního rozkladu si nejprve zavedeme spektrální rozklad.

2.1.1 Spektrální rozklad

Věta 2.1. *Nechť $A \in \mathbb{C}^{m \times n}$, potom platí*

$$N(A) \oplus R(A^*) = \mathbb{C}^n, \quad N(A) \perp R(A^*),$$

$$N(A^*) \oplus R(A) = \mathbb{C}^m, \quad N(A^*) \perp R(A).$$

Věta 2.2 (Spektrální rozklad pro normální matice). *Matice A je normální právě tehdy, když existuje unitární matice U a diagonální matice D tak, že*

$$U^*AU = D, \quad \text{tj.} \quad A = UDU^*$$

Předpokládejme, že A je hermitovská pozitivně semidefinitní řádu n s hodnotami r s nezápornými vlastními čísly $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_r > 0$, $\lambda_{r+1} = \dots = \lambda_n = 0$. Podle věty 2.1 existuje rozklad matice A ve tvaru

$$A = Q\Lambda Q^*, \quad Q^*Q = I, \quad \Lambda = \text{diag}(\lambda_1, \dots, \lambda_n).$$

Sloupce unitární matice $Q = (q_1, \dots, q_n)$ jsou vlastní vektory matice A a platí

$$Aq_j = \lambda_j q_j, \quad j = 1, \dots, n$$

Vlastní vektory q_1, q_2, \dots, q_n tvoří ortonormální bázi prostoru \mathbb{C}^n .

Ze spektrálního rozkladu matice A vidíme bázi nulového prostoru a oboru hodnot matice A ,

$$R(A) = \text{span}\{q_1, \dots, q_r\}, \quad N(A) = \text{span}\{q_{r+1}, \dots, q_n\},$$

dále nám dává dyadický rozvoj matice A

$$A = \sum_{j=1}^n \lambda_j q_j q_j^* = \sum_{j=1}^m A_j, \quad A_j = \lambda_j q_j q_j^*,$$

kde matice A_j jsou uspořádány podle spektrální (či Frobeniovy) normy matice a platí $\|A_j\| = \|A_j\|_F = \lambda_j$, tedy můžeme matici A aproximovat maticí nižší hodnosti

$$A^{(k)} = \sum_{j=1}^k A_j.$$

Pro odvození singulárního rozkladu budou důležité spektrální rozklady matic A^*A a AA^* . Matice A^*A a AA^* jsou čtvercové hermitovské pozitivně semidefinitní a můžeme uvažovat jejich spektrální rozklad.

Lemma 2.3. Pro komplexní matici $A \in \mathbb{C}^{m \times n}$ platí

$$N(A^*A) = N(A), \quad R(A^*A) = R(A^*),$$

$$N(AA^*) = N(A^*), \quad R(AA^*) = R(A).$$

Důsledek **Věty 2.1** a **Lemma 2.3** je, že

$$\dim(R(AA^*)) = \dim(R(A)) = r = \dim(R(A^*)) = \dim(R(A^*A)),$$

pro A hodnosti r .

Buď λ_j vlastní čísla a v_j ortonormální vlastní vektory matice A^*A , $j = 1, \dots, n$. Uvažujme dále bez újmy na obecnosti

$$\lambda_1 \geq \dots \geq \lambda_r > 0, \quad \lambda_{r+1} = \lambda_n = 0,$$

pak pro $V = (v_1, \dots, v_n)$ platí

$$V^* A^* A V = \text{diag}(\lambda_1, \dots, \lambda_r, 0, \dots, 0)$$

Dále z **Lemma 2.3** platí, že v_1, \dots, v_r resp. v_{r+1}, \dots, v_n tvoří ortonormální bázi

$$R(A^*A) = R(A^*) = \text{span}\{v_1, \dots, v_r\},$$

resp.

$$N(A) = N(A^*A) = \text{span}\{v_{r+1}, \dots, v_n\}.$$

Následující lemma nám dává vztah mezi ortonormálními vektory matic A^*A a AA^* .

Lemma 2.4. Uvažujme spektrální rozklad matice A^*A s vlastními čísly λ_j a v_j , $j = 1, \dots, r$. Potom jsou vektory

$$u_j \equiv A v_j / \sqrt{\lambda_j}, \quad j = 1, \dots, r$$

ortonormální vlastní vektory AA^* a platí

$$AA^*v_j = \lambda_j u_j, \quad \|u_j\| = 1, \quad j = 1, \dots, r.$$

Z **Lemma 2.4** plyne, že vektory u_j , $j = 1, \dots, r$ jsou ortonormální vlastní vektory matice AA^* , které odpovídají právě m nenulovým vlastním číslům λ_j a platí

$$Av_j = \sqrt{\lambda_j} u_j, \quad j = 1, \dots, r.$$

Analogie těchto vztahů platí i pro matici AA^* , tj. pokud u_1, \dots, u_r tvoří ortonormální bázi oboru hodnot matice a u_{r+1}, \dots, u_m tvoří ortonormální bázi doplňku oboru hodnot, pak platí

$$\text{span}\{u_1, \dots, u_r\} = R(AA^*) = R(A),$$

$$\text{span}\{u_{r+1}, \dots, u_m\} = N(AA^*) = N(A^*).$$

Z konstrukce plyne, že nenulová vlastní čísla matic A^*A a AA^* se rovnají včetně násobnosti.

Nyní můžeme popsat singulární rozklad.

Definice 2.5 (Singulární čísla). Nechť $A \in \mathbb{C}^{m \times n}$. Odmocniny vlastních čísel matice A^*A nazveme singulárními čísly matice A ,

$$\sigma_j = \sqrt{\lambda_j}, \quad j = 1, \dots, r, \quad r = \text{rank}(A).$$

Jelikož vlastní čísla λ_j jsou uspořádána od největšího po nejmenší, tak platí

$$\sigma_1 \geq \dots \geq \sigma_r > 0.$$

Tedy podle předchozích poznatků platí $Av_j = \sigma_j u_j$, $j = 1, \dots, r$ a můžeme psát

$$AV = U\Sigma,$$

kde

$$\Sigma = \begin{bmatrix} \Sigma_r & 0 \\ 0 & 0 \end{bmatrix} \in \mathbb{R}^{m \times n}, \quad \Sigma_r = \text{diag}(\sigma_1, \dots, \sigma_r) \in \mathbb{R}^{r \times r},$$

$$V = (v_1, \dots, v_n), U = (u_1, \dots, u_m).$$

Ekvivalentně

$$A = U\Sigma V^*, \quad A^* = V\Sigma^T U^*.$$

Vidíme tedy, že pokud bychom byli schopni spočítat singulární rozklad matice A , tak nulový prostor podle **Lemma 2.3** je roven lineárnímu obalu vektorů v_{r+1}, \dots, v_n .

Věta 2.6. Mějme matici $A \in \mathbb{R}^{m \times n}$ řádu m a její singulární čísla $\sigma_1 \geq \dots \geq \sigma_m > 0$. Potom platí $\|A\| = \sigma_1$, $\|A\|_F = (\sigma_1^2 + \sigma_2^2 + \dots + \sigma_m^2)^{1/2}$. Dále

$$\|A\| \leq \|A\|_F \leq \sqrt{m} \|A\|.$$

Jelikož platí, že

$$\|A\| = \sigma_1, \text{ a } \|A^{-1}\| = \sigma_m^{-1},$$

tak dostáváme

$$\kappa(A) = \sigma_1 / \sigma_m.$$

2.1.2 Výpočet nulového prostoru matice pomocí SVD

Nyní pomocí SVD můžeme vyřešit úlohu hledání matice B splňující $AB = 0$ následovně

$$P_1 A P_2 = U \Sigma V,$$

$$[V_1, V_2] = V^*,$$

$$B = P_2 V_2.$$

SVD můžeme vypočítat za pomoci spektrálních rozkladů matic AA^* a A^*A , ovšem podmíněnost obou matic je rovna $\kappa(A)^2$, tedy pokud by byla matice A špatně podmíněna, tak by podle **Věty 2.6.** mohl být výpočet singulárních čísel nepřesný.

Abychom se vyhnuli problémům se špatnou podmíněností matice, tak využíváme následující standardní metody pro výpočet SVD a následovně i nulový prostor matice A .

Algoritmus 1 (Výpočet jádra matice pomocí SVD). Mějme matici A typu $m \times n$ hodnoti m .

- Transformujeme matici A na bidiagonální tvar užitím unitárních transformací nebo Golub-Kahanovy iterační bidiagonalizace. Transformovaná matice A je ve tvaru

$$B = PAQ,$$

kde P, Q jsou unitární a B je bidiagonální.

- Nalezení singulární rozkladu bidiagonální matice B pomocí iteračního procesu, který nuluje její nadbidiagonální prvky, například pomocí implicitního QR algoritmu.
- Máme rozklad $\Sigma = U^* A V$ a nulový prostor matice A tedy odpovídá sloupcovým vektorům v_{m+1}, \dots, v_n matice V , tedy

$$B = [v_{m+1}, \dots, v_n].$$

Algoritmus výše je proveden bez permutací, ovšem v obecném případě, kdy permutujeme matici A na matici P_1AP_2 , tak můžeme opět aplikovat algoritmus výše a dostaneme rozklad

$$\Sigma = U^*P_1AP_2V,$$

a potom platí, že hledaná matice nulového prostoru je

$$B = P_2V_2.$$

Složitost výpočtu matice B závisí na iteračním procesu, kterým počítáme singulární rozklad bidiagonální matice. Kdybychom předpokládali, že A je čtvercová matice řádu m, iterační proces rychle konverguje a nezvyšuje počet operací, pak dolní odhad na počet operací je $\frac{16}{3}m^3$.

2.2 LU rozklad

Jednou z nejdůležitějších metod pro řešení soustav lineárních rovnic je LU rozklad, který nám původní matici A rozloží na součin dolní trojúhelníkové matice a horní trojúhelníkové matice. LU rozklad je založen na Gaussově eliminaci následovně

$$A \longrightarrow A^{(1)} \equiv \begin{bmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,n} \\ 0 & a_{2,2}^{(1)} & \cdots & a_{2,n}^{(1)} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & a_{m,2}^{(1)} & \cdots & a_{m,n}^{(1)} \end{bmatrix} = M^{-1}A,$$

kde

$$M_1^{-1} = I - m_1 e_1^T = \begin{bmatrix} 1 & & & & \\ -\frac{a_{2,1}}{a_{1,1}} & 1 & & & \\ -\frac{a_{3,1}}{a_{1,1}} & & 1 & & \\ \vdots & & & \ddots & \\ -\frac{a_{m,1}}{a_{1,1}} & & & & 1 \end{bmatrix}, \quad m_1 = \begin{bmatrix} \frac{a_{2,1}}{a_{1,1}} \\ \frac{a_{3,1}}{a_{1,1}} \\ \vdots \\ \frac{a_{m,1}}{a_{1,1}} \end{bmatrix}.$$

Jelikož A má hodnost m, tak permutováním zaručíme, že nikdy nedělíme 0, tedy LU rozklad lze provést. V i-tém kroku bude

$$A^{(i)} \equiv \begin{bmatrix} a_{1,1} & a_{1,2} & \cdots & & a_{1,n} \\ 0 & a_{2,2}^{(1)} & \cdots & & a_{2,n}^{(1)} \\ \vdots & 0 & \ddots & & \vdots \\ & \vdots & & a_{i,i}^{(i-1)} & \cdots & a_{i,n}^{(i-1)} \\ 0 & 0 & & a_{m,i}^{(i-1)} & \cdots & a_{m,n}^{(i-1)} \end{bmatrix}, \quad m_i = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ \frac{a_{i+1,i}}{a_{i,i}} \\ \frac{a_{i+2,i}}{a_{i,i}} \\ \vdots \\ \frac{a_{m,i}}{a_{i,i}} \end{bmatrix}.$$

Po $m-1$ krocích dostaneme

$$U \equiv A^{(m-1)} = M_{m-1}^{-1} \dots M_2^{-1} M_1^{-1} A, \quad L \equiv M_1 M_2 \dots M_{m-1},$$

kde U je horní trojúhelníková typu $m \times n$ a L je dolní trojúhelníková řádu m s jednotkami na diagonále a upravením rovnosti dostáváme

$$A = LU = L[U_1 \ U_2], \quad U_1 = [u_1, \dots, u_m], \quad U_2 = [u_{m+1}, \dots, u_n].$$

2.2.1 Výpočet jádra matice pomocí LU rozkladu

Nyní můžeme LU rozklad použít k řešení naší úlohy $AB = 0$.

Algoritmus 2 (Výpočet jádra pomocí LU rozkladu). Mějme matici A typu $m \times n$ hodnoti m .

- Provedeme potřebné permutace matice A na matici $P_1 A P_2$, abychom měli rozklad matice v hledaném tvaru.
- Pomocí nějaké varianty vypočítáme LU rozklad, tedy $P_1 A P_2 = LU = L[U_1 \ U_2]$.
- Položíme $J = U_1^{-1} U_2$, pak hledané řešení úlohy je

$$B = P_2 \begin{bmatrix} -J \\ I_{n-m} \end{bmatrix}$$

2.3 Sloupcová eliminace

Metoda sloupcové eliminace je založena na stejné myšlence jako LU rozklad, nicméně místo poddiagonálních prvků budeme nulovat naddiagonální prvky. Tedy

$$M_1^{-1} = \begin{bmatrix} 1 & -\frac{a_{1,2}}{a_{1,1}} & -\frac{a_{1,3}}{a_{1,1}} & \dots & -\frac{a_{1,n}}{a_{1,1}} \\ 0 & 1 & 0 & \dots & 0 \\ \vdots & & \ddots & & \\ 0 & & & & 1 \end{bmatrix},$$

pak

$$A^{(1)} = A M_1^{-1} = \begin{bmatrix} a_{1,1} & 0 & 0 & \dots & 0 \\ a_{2,1} & a_{2,2}^{(1)} & a_{2,3}^{(1)} & \dots & a_{2,n}^{(1)} \\ \vdots & \vdots & \vdots & & \vdots \\ a_{m,1} & a_{m,2}^{(1)} & \dots & & a_{m,n}^{(1)} \end{bmatrix}.$$

V obecném i -tém kroku ($i < m$) budeme mít

$$M_i^{-1} = \begin{bmatrix} 1 & 0 & \dots & & 0 \\ 0 & 1 & & & 0 \\ \vdots & 0 & \ddots & & \\ & & & 1 & -\frac{a_{i,i+1}}{a_{i,i}} & \dots & -\frac{a_{m,n}}{a_{i,i}} \\ & & & & \ddots & & 0 \\ 0 & \dots & & & & 0 & 1 \end{bmatrix}$$

a

$$A^{(i)} = AM_1^{-1}M_2^{-1} \dots M_i^{-1} =$$

$$= \begin{bmatrix} a_{1,1} & 0 & \dots & & 0 \\ a_{2,1} & a_{2,2}^{(1)} & 0 & \dots & \vdots \\ \vdots & a_{3,2}^{(1)} & a_{3,3}^{(2)} & \ddots & \\ & & \ddots & \ddots & \\ & & & a_{i,i-1}^{(i-1)} & a_{i,i}^{(i)} & 0 & \dots \\ & & & & \ddots & \ddots & \\ a_{m,1} & a_{m,2}^{(1)} & \dots & & & a_{m,m}^{(i)} & \dots & a_{m,n}^{(i)} \end{bmatrix}.$$

Po $m-1$. kroku tedy dostáváme rozklad

$$P_1AP_2 = [L \ 0]U, U^{-1} = [N_1 \ N_2],$$

kde N_1 je typu $n \times m$ a N_2 je typu $n \times (n - m)$, L je dolní trojúhelníková řádu m a U je horní trojúhelníková řádu n .

2.3.1 Výpočet jádra matice pomocí sloupcové eliminace

Algoritmus 3 (Výpočet jádra pomocí sloupcové eliminace). Mějme matici A typu $m \times n$ hodnosti m .

- Permutováním převedeme A na P_1AP_2 .
- Provedeme analogii Gaussovy eliminace pro sloupcovou eliminaci a získáme rozklad $P_1AP_2 = [L \ 0]U$.
- Položíme $U^{-1} = [N_1 \ N_2]$, potom hledaná matice nulového prostoru je

$$B = P_2N_2.$$

2.4 Gauss-Jordanova eliminace

Tato metoda navazuje na LU rozklad, máme-li tedy rozklad

$$P_1 A P_2 = L[U_1 \ U_2],$$

tak v Gauss-Jordanově eliminaci ještě vynulujeme naddiagonální prvky matice $[U_1 \ U_2]$ s tím, že před každým krokem vynormujeme příslušný diagonální prvek tak, aby byl roven 1.

Postup v i -tém kroce je následující

$$M_i^{-1} = \begin{bmatrix} 1 & 0 & \dots & 0 & -a_{1,i} & 0 & \dots & 0 \\ 0 & 1 & \ddots & & \vdots & & & \vdots \\ \vdots & 0 & \ddots & & -a_{i-1,i} & & & \\ & & \vdots & & 1 & 0 & & \\ & & & & & \ddots & \ddots & \\ & & & & & & 1 & 0 \\ 0 & \dots & & & & & 0 & 1 \end{bmatrix}, \quad A^{(i)} = \frac{1}{a_{i,i}^{(i-1)}} M_i^{(-1)} A^{(i-1)}$$

Po $m-1$. kroku dostáváme

$$A^{(m-1)} = M_{m-1}^{-1} \dots M_2^{-1} M_1^{-1} L[U_1 \ U_2] = \begin{bmatrix} 1 & 0 & \dots & 0 & a_{1,m+1}^{(m-1)} & \dots & a_{1,n}^{(m-1)} \\ 0 & 1 & & \vdots & a_{2,m+1}^{(m-1)} & \dots & a_{2,n}^{(m-1)} \\ \vdots & 0 & \ddots & & \vdots & & \vdots \\ 0 & \dots & 0 & 1 & a_{m+1,m+1}^{(m-1)} & \vdots & a_{m,n}^{(m-1)} \end{bmatrix} = [I_m \ J]$$

kde

$$J = \begin{bmatrix} a_{1,m+1}^{(m-1)} & a_{1,m+2}^{(m-1)} & \dots & a_{1,n}^{(m-1)} \\ a_{2,m+1}^{(m-1)} & a_{2,m+2}^{(m-1)} & \dots & a_{2,n}^{(m-1)} \\ \vdots & & \ddots & \\ a_{m+1,m+1}^{(m-1)} & \dots & & a_{m,n}^{(m-1)} \end{bmatrix}$$

Získali jsme tedy následující rozklad

$$P_1 A P_2 = M[I_m \ J],$$

kde M je řádu m a J je typu $m \times (n - m)$.

2.4.1 Výpočet jádra pomocí Gauss-Jordanovy eliminace

Algoritmus 4 (Výpočet jádra pomocí Gauss Jordanovy eliminace). Mějme matici A typu $m \times n$ hodnosti m .

- Provedeme potřebné permutace a převedeme matici A na matici $P_1 A P_2$.

- Vypočítáme LU rozklad matice $P_1AP_2 = LU$.
- Postupným přenásobováním a sloupcovou eliminací získáme požadovaný rozklad $P_1AP_2 = M[I_m \ J]$
- Hledaný nulový prostor matice A je roven

$$B = P_2 \begin{bmatrix} -J \\ I_{n-m} \end{bmatrix}.$$

2.5 QR rozklad

Tento rozklad nám rozloží matici A na součin unitární matice Q typu $m \times m$ a horní trojúhelníkové matice R typu $m \times n$, tedy $A = QR$. Máme více algoritmů pro výpočet QR rozkladu matice A, konkrétně vybíráme mezi metodami založených na Givensových rotacích, Householderových reflexích a nebo Gram-Schmidtovým ortogonalizačním procesem. V našem případě budeme uvažovat Gram-Schmidtův ortogonalizační proces. Ten nám z matice A iteračně vygeneruje ortonormální bázi $Q = (q_1, \dots, q_m)$ a matici koeficientů R.

Označme $A = [a_1, a_2, \dots, a_n]$, kde a_1, \dots, a_m jsou sloupcové vektory matice A, budeme navíc v každém kroku iterace chtít, aby platilo $\text{span}\{a_1, \dots, a_k\} = \text{span}\{q_1, \dots, q_k\}$. Postup je pak následující

- V prvním kroku normováním dostaneme $r_{1,1} = \|a_1\|$ a $q_1 = \frac{a_1}{r_{1,1}}$.
- V druhém kroku musí platit $\text{span}\{a_1, a_2\} = \text{span}\{q_1, q_2\}$ a $q_1 \perp q_2$, což splníme tím, že odečteme projekci vektoru a_2 do podprostoru $\text{span}q_1$ a vynormujeme. Tedy

$$z = (I - q_1 q_1^*) a_2 = a_2 - (q_1^* a_2) q_1, \quad r_{1,2} = q_1^* a_2, \quad r_{2,2} = \|z\|,$$

$$q_2 = \frac{z}{r_{2,2}}.$$

- V k-tém kroku stejným postupem odečteme od vektoru a_k jeho projekci do prostoru $\text{span}q_1, \dots, q_{k-1}$ a vynormujeme. Konkrétně

$$z = (I - Q_{k-1} Q_{k-1}^*) a_k = a_k - \sum_{i=1}^{k-1} (q_i^* a_k) q_i, \quad r_{i,k} = q_i^* a_k, \quad k = 1, \dots, k-1, \quad r_{k,k} = \|z\|,$$

$$q_k = \frac{z}{r_{k,k}}.$$

Ve výsledku dostáváme unitární matici $Q = [q_1, \dots, q_m]$ a horní trojúhelníkovou matici typu $m \times n$

$$R = \begin{bmatrix} r_{1,1} & r_{1,2} & \dots & r_{1,m} & r_{1,m+1} & \dots & r_{1,n} \\ 0 & r_{2,2} & & r_{2,n} & r_{2,m+1} & & r_{2,n} \\ \vdots & \ddots & \ddots & & & \ddots & \\ 0 & \vdots & 0 & r_{m,m} & r_{m,m+1} & & r_{m,n} \end{bmatrix}$$

a bude platit rovnost

$$A = QR.$$

Ve výpočtu výše jsme předpokládali, že prvních m sloupcových vektorů matice A jsou lineárně nezávislé, což můžeme, neboť matice A je hodnosti m . Permutováním tedy můžeme tyto předpoklady zaručit.

2.5.1 Výpočet jádra pomocí QR rozkladu

Mějme matici A typu $m \times n$ hodnosti m .

Algoritmus 5 (Výpočet jádra pomocí QR rozkladu). Mějme matici A typu $m \times n$ hodnosti m .

- Permutováním převedeme A na $P_1 A P_2$.
- Pomocí vhodné metody vypočítáme QR rozklad $P_1 A P_2 = QR = Q[R_1 \ R_2]$.
- Položíme $J = R_1^{-1} R_2$. Potom hledaná matice nulového prostoru je

$$B = P_2 \begin{bmatrix} -J \\ I_{n-m} \end{bmatrix}.$$

kde pomocí $R = [r_1, \dots, r_m, r_{m+1}, \dots, r_n]$ značíme $R_1 = [r_1, \dots, r_m]$, $R_2 = [r_{m+1}, \dots, r_n]$.

2.6 LQ rozklad

3 Teoretické porovnání numerických metod

V teorii: Numerická stabilita jednotlivých metod, přesnost/chyba, výpočetní cena (kolik operací?), čas výpočtu, různé metody pro různý druh matic? Celkově: co bychom od každé metody mohli teoreticky očekávat