UNIVERZITA PALACKÉHO V OLOMOUCI
PŘÍRODOVĚDECKÁ FAKULTA

BAKALÁŘSKÁ PRÁCE

Soustavy lineárních rovnic s obdélníkovými maticemi a jejich řešení

Katedra matematické analýzy a aplikací matematiky
Vedoucí bakalářské práce: **RNDr. Jitka Machalová, Ph.D.**
Vypracoval: **Radim Kunhart**
Studijní program: B1103 Aplikovaná matematika
Studijní obor: Matematika–ekonomie se zaměřením na bankovnictví/pojišťovnictví
Forma studia: prezenční
Rok odevzdání: 2017

BIBLIOGRAFICKÁ IDENTIFIKACE

Autor: Radim Kunhart

Název práce: Soustavy lineárních rovnic s obdélníkovými maticemi a jejich řešení

Typ práce: Bakalářská práce

Pracoviště: Katedra matematické analýzy a aplikací matematiky

Vedoucí práce: RNDr. Jitka Machalová, Ph.D.

Rok obhajoby práce: 2017

Abstrakt: Tato práce je zaměřena na řešení soustav lineárních rovnic s obdélníkovými maticemi, a to za pomoci pseudoinverzních matic. Ukážeme si zde devět různých metod pro výpočet pseudoinverzních matic, které budeme ilustrovat na vhodně zvolených příkladech. Součástí práce jsou také kódy vytvořené ve výpočetním softwaru R. Ty řeší všech devět metod pro výpočet pseudoinverzních matic, ale také, s jejich využitím, řeší soustavy lineárních rovnic jako takové.

Klíčová slova: soustava lineárních rovnic, pseudoinverze matice, software R

Počet stran: 59

Počet příloh: 1

Jazyk: český

BIBLIOGRAPHICAL IDENTIFICATION

Author: Radim Kunhart

Title: Solving a system of linear equations with a non square matrix

Type of thesis: Bachelor's

Department:
Department of Mathematical Analysis and Application of Mathematics

Supervisor: RNDr. Jitka Machalová, Ph.D.

The year of presentation: 2017

Abstract: This thesis is focused on solving the systems of linear equations with rectangular matrices using pseudoinverse matrices. Nine different methods for calculating pseudoinverse matrices are explained and illustrated by appropriately selected examples. Another part of the thesis are the codes created in the computer software R. They solve all nine methods for the calculation of pseudoinverse matrices, but also, with their use, solve the systems of linear equations as such.

Key words: system of linear equations, pseudoinverse of matrix, software R

Number of pages: 59

Number of appendices: 1

Language: Czech

Prohlášení

Prohlašuji, že jsem bakalářskou práci zpracoval samostatně pod vedením paní RNDr. Jitky Machalové, Ph.D. a všechny použité zdroje jsem uvedl v seznamu literatury.

V Olomouci dne 5. května 2017 ....

podpis

Obsah

Poděkování

Děkuji paní RNDr. Jitce Machalové, Ph.D. za spolupráci a za čas, který mi věnovala při konzultacích. Zároveň děkuji i celé své rodině za podporu po celou dobu studia.

Úvod

Ruční výpočet soustav lineárních rovnic je zdlouhavý, a proto si pomáháme výpočetními softwary. Doposud v žádném dostupném matematickém softwaru nejsou implicitně nastaveny příkazy, potažmo kódy, pro různé metody výpočtů inverzních a pseudoinverzních matic. Tyto matice jsou zapotřebí při výpočtech soustav lineárních rovnic. Vhledem k tomu jsme se rozhodli, že takové kódy sami vytvoříme.

Veškeré kódy v této práci jsou zpracovány ve výpočetním softwaru R. Někteří lidé vnímají R jen jako 18. písmeno v abecedě. Pro nás to však také je jeden z nejznámějších volně šířitelných matematicko-statistických softwarů, který poprvé spatřil světlo světa roku 1996. Jelikož je tento software volně šířitelný, tudíž bezplatný, nachází si stále větší oblibu u uživatelů, a v poslední době jejich počtem předčil i takové komerční programy jako jsou SAS, nebo MATLAB.

První kapitola pojednává o základních pojmech. A to jak teoretických, zejména z teorie matic, tak také praktických, zabývající se výpočetním softwarem R.

Ve druhé kapitole, která je zároveň jádrem této práce, ukážeme devět různých postupů, kterými lze spočítat pseudoinverzi matice. Zároveň jsme ke každému vytvořili kód, kterým můžeme pseudoinverzi spočítat.

Ve třetí kapitole ukážeme řešení soustav rovnic ve všech možných případech. Navážeme tím tak na druhou kapitolu, kde jsme si předpřipravili matematický aparát a kódy pro výpočet pseudoinverzí matic.

Rádi bychom také zde v úvodu předestřeli, že ve výpočetním softwaru R není definovaná nula. Při počítání proto musíme nastavit přesnost jako číslo *eps* rovnající se téměř nule. Tuto problematiku si představíme v poslední kapitole této práce zabývající se stabilitou a rychlostí našich kódů pro výpočet pseudoinverzí matic.

„Pouze dvě věci jsou nekonečné. Vesmír a lidská hloupost. U té první si tím však nejsem tak jist.“ A. Einstein

1. Přípravná kapitola

V první podkapitole si ukážeme základní pojmy teoretické. Některé základní pojmy, a to například z teorie matic, jsme do této práce záměrně nezahrnovali. Tyto základní pojmy můžete nalézt například v [3], nebo jiných knihách zabývajících se lineální algebru jako takovou. Děláme tak z toho důvodu, jelikož předpokládáme alespoň základní čtenářovu znalost v oblasti lineární algebry.

Ve druhé podkapitole si vysvětlíme základy pro práci se softwarem R. Jestliže by nám tyto základy nestačily, je možné se rovněž podívat do podrobnější literatury, například do [7].

1.1. Přípravná kapitola teoretická

V této kapitole si připomeneme základní věty a definice související s řešením soustav lineárních rovnic. Například si nadefinujeme, co je soustava lineárních rovnic, jakým způsobem ji můžeme převést na maticový tvar, anebo jak vypadá Moore–Penroseova pseudoinverze.

Definice 1.1 Soubor prvků

$$\mathbf{A} = (a_{ij}) = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{pmatrix} \tag{1}$$

nazýváme maticí typu m x n. Vektor $(a_{i1}, a_{i2}, \dots, a_{in})$ nazýváme i-tým řádkem matice $\mathbf{A}$ a vektor $(a_{1j}, a_{2j}, \dots, a_{mj})^T$ nazýváme j-tým sloupcem matice $\mathbf{A}$.

Definice 1.2 Soustavou m lineárních rovnic o n neznámých rozumíme soustavu rovnic ve tvaru

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + \ldots + a_{1n}x_n &= b_1 \\ a_{21}x_1 + a_{22}x_2 + \ldots + a_{2n}x_n &= b_2 \\ &\vdots \\ a_{m1}x_1 + a_{m2}x_2 + \ldots + a_{mn}x_n &= b_m \ , \end{aligned} \tag{2}$$

kde reálná čísla a_{ij}, $i \in \{1, 2, \ldots, m\}$, $j \in \{1, 2, \ldots, n\}$, nazýváme koeficienty levé strany a reálná čísla b_i, $i \in \{1, 2, \ldots, m\}$, nazýváme koeficienty pravé strany soustavy rovnic. Proměnné x_1, x_2, ..., x_n nazýváme neznámé. Řešením soustavy rovnic rozumíme takovou uspořádanou n-tici reálných čísel $\mathbf{x} = (x_1, x_2, \ldots, x_n)^T$, že po jejím dosazení do soustavy jsou všechny rovnice splněny.

Definice 1.3 Matice $\mathbf{A} = (a_{ij})$ typu m x n se nazývá matice soustavy (2), matici

$$\mathbf{A_r} = \begin{pmatrix} a_{11} & a_{12} & \ldots & a_{1n} & b_1 \\ a_{21} & a_{22} & \ldots & a_{2n} & b_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{m1} & a_{m2} & \ldots & a_{mn} & b_m \end{pmatrix} \tag{3}$$

nazýváme rozšířenou maticí soustavy (2).

Poznámka 1.1 Soustavu (2) lze ekvivalentně přepsat do maticového tvaru pomocí maticového součinu

$$\begin{pmatrix} a_{11} & a_{12} & \ldots & a_{1n} \\ a_{21} & a_{22} & \ldots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \ldots & a_{mn} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix}, \tag{4}$$

nebo zkráceně

$$\mathbf{Ax} = \mathbf{b}, \tag{5}$$

kde $\mathbf{A}$ je maticí soustavy, $\mathbf{x} = (x_1, x_2, \ldots, x_n)^T$ je vektor řešení soustavy a $\mathbf{b} = (b_1, b_2, \ldots, b_n)^T$ je vektor pravé strany soustavy rovnic.

Definice 1.4 Nechť $\mathbf{A}$ je matice typu m x n. Hodnost matice $\mathbf{A}$ je rovna maximálnímu počtu jejích lineárně nezávislých řádků. Hodnost matice značíme $r(\mathbf{A})$.

Věta 1.1 *(Frobeniova věta) Soustava (2) je řešitelná právě tehdy, když její matice soustavy (1) a rozšířená matice soustavy (3) mají stejnou hodnost, tj.* $r(\boldsymbol{A}) = r(\boldsymbol{A}_r)$.

Důkaz: viz [3].

Definice 1.5 Nechť je matice $\mathbf{A}$ čtvercová a regulární (nemá lineárně závislé řádky), pak k matici $\mathbf{A}$ existuje matice inverzní $\mathbf{A}^{-1}$, taková, že platí následující rovnice:

$$\mathbf{A}\mathbf{A}^{-1} = \mathbf{A}^{-1}\mathbf{A} = \mathbf{I}, \tag{6}$$

kde $\mathbf{I}$ značí jednotkovou matici.

Definice 1.6 Vektorová norma na $\mathbb{R}^n$ je funkce $\|.\| : \mathbb{R}^n \to \mathbb{R}$ s následujícími vlastnostmi:

1. $\|\mathbf{x}\| \geq 0 \qquad \forall\, \mathbf{x} \in \mathbb{R}^n$
2. $\|c\mathbf{x}\| = |c|\,\|\mathbf{x}\| \qquad \forall\, \mathbf{x} \in \mathbb{R}^n, \forall\, c \in \mathbb{R}$
3. $\|\mathbf{x}\| = 0 \Leftrightarrow \mathbf{x} = \mathbf{o}$, kde $\mathbf{o}$ je nulový vektor
4. $\|\mathbf{x}+\mathbf{y}\| \leq \|\mathbf{x}\| + \|\mathbf{y}\| \qquad \forall\, \mathbf{x} \in \mathbb{R}^n, \forall\, \mathbf{y} \in \mathbb{R}^n$

Poznámka 1.3 Při výpočtu Moore-Penroseovy pseudoinverze v této práci používáme tzv. eukleidovskou normu, kterou definujeme následovně:

$$\|\mathbf{x}\|_2 = \sqrt{\sum_{i=1}^{n} |x_i|^2} \tag{7}$$

Tato norma udává vzdálenost bodu $\mathbf{x}$ od počátku soustavy souřadnic, což je důsledek Pythagorovy věty.

Definice 1.7 Nechť $\mathbf{A}$ je daná matice. Matice $\mathbf{A}^+$ se nazývá Moore–Penroseova pseudoinverze matice k matici $\mathbf{A}$, jestliže platí:

$$\mathbf{A}\mathbf{A}^+\mathbf{A} = \mathbf{A} \tag{8}$$

$$\mathbf{A}^+\mathbf{A}\mathbf{A}^+ = \mathbf{A}^+ \tag{9}$$

$$(\mathbf{A}^+\mathbf{A})^T = \mathbf{A}^+\mathbf{A} \tag{10}$$

$$(\mathbf{A}\mathbf{A}^+)^T = \mathbf{A}\mathbf{A}^+ \tag{11}$$

Poznámka 1.2 Z definice 1.5 vyplývá, že jestliže $\mathbf{A}$ je čtvercová regulární matice, pak existuje inverzní matice $\mathbf{A}^{-1}$, která rovněž splňuje výše uvedené čtyři axiomy z definice 1.7, a tedy $\mathbf{A}^{-1}$ je také pseudoinverzní maticí k matici $\mathbf{A}$.

Definice 1.8 Matice $\mathbf{A}$ typu m x n se nazývá matice plné řádkové hodnosti, jestliže hodnost matice $\mathbf{A}$ je rovna počtu řádků, tedy $r(\mathbf{A}) = m$.

Matice $\mathbf{A}$ typu m x n se nazývá matice plné sloupcové hodnosti, jestliže hodnost matice $\mathbf{A}$ je rovna počtu sloupců, tedy $r(\mathbf{A}) = n$.

Poznámka 1.3 Pro některé speciální typy matic vycházejí pseudoinverze velice jednoduše:

1. Pro čtvercovou regulární matici $\mathbf{A}$ je

$$\mathbf{A}^+ = \mathbf{A}^{-1}. \tag{12}$$

2. Je-li matice $\mathbf{A}$ skalár, tedy matice typu 1 x 1, pak

$$\mathbf{A}^+ = \begin{cases} 1/\mathbf{A} & \text{pokud} \quad \mathbf{A} \neq 0, \\ 0 & \text{pokud} \quad \mathbf{A} = 0. \end{cases} \tag{13}$$

3. Je-li matice $\mathbf{A}$ diagonální, tedy matice typu $\mathbf{A} = diag(a_1, ..., a_n)$, pak

$$\mathbf{A}^+ = diag(a_1^+, ..., a_n^+), \tag{14}$$

kde

$$a_i^+ = \begin{cases} 1/a_i & \text{pokud} \quad a_i \neq 0, \\ 0 & \text{pokud} \quad a_i = 0. \end{cases}$$

4. Je-li čtvercová matice $\mathbf{A}$ symetrická, pak existuje ortogonální matice $\mathbf{T}$ a diagonální matice $\mathbf{D}$ takové, že $\mathbf{A} = \mathbf{TDT}^T$. Potom

$$\mathbf{A}^+ = \mathbf{TD}^{-1}\mathbf{T}^T. \tag{15}$$

5. Pro matici $\mathbf{A}$ plné řádkové hodnosti, tj. matici typu r x n a hodnosti r je

$$\mathbf{A}^+ = \mathbf{A}^T(\mathbf{AA}^T)^{-1}. \tag{16}$$

6. Pro matici $\mathbf{A}$ plné sloupcové hodnosti, tj. matici typu m x r a hodnosti r je

$$\mathbf{A}^+ = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T. \tag{17}$$

Definice 1.9 Skalární součin dvou vektorů $\mathbf{u} = (u_1, u_2), \mathbf{v} = (v_1, v_2)$ v rovině je číslo $u_1v_1 + u_2v_2$. Skalární součin vektorů $\mathbf{u}$ a $\mathbf{v}$ zapisujeme jako $(\mathbf{u}, \mathbf{v})$.

Definice 1.10 Stopou čtvercové matice řádu n nazýváme součet jejich prvků na hlavní diagonále. Značíme ji trace($\mathbf{A}$). Spočítáme ji jako trace($\mathbf{A}$) $= a_{11} + a_{22} + ... + a_{nn}$.

.

1.2. Přípravná kapitola pro práci se softwarem R

Samotná instalace softwaru R je pro uživatele nenáročná a jeho instalační soubor je volně k dispozici na stránkách distributora `https://cran.r-project.org/`. Na této stránce lze rovněž nalézt v sekci ”Manuals” veškeré návody potřebné při práci se softwarem R.

Pro uživatelsky příjemnější práci se softwarem R doporučujeme doinstalovat program RStudio, jehož instalační soubor je volně k dispozici na stránkách distributora `https://www.rstudio.com/`.

V základní nabídce softwaru R nejsou kódy pro řešení pseudoinverzních matic. Avšak některé jsou již zpracovány v doplňkových knihovnách, označovaných jako balíčky (packages). Například v balíčku „MASS“ nalezneme příkaz „ginv“, který vypočítá pseudoinverzi matice za pomoci singulárního rozkladu.

V následující kapitole si představíme devět metod pro výpočet pseudoinverze matice. Pro každou z těchto metod jsme vytvořili kód, který nám pseudoinverzi podle dané metody vypočítá.

Pro usnadnění načtení těchto kódů do softwaru R, jsme vytvořili doplňkový balíček s názvem „pseudoinverze“, který naleznete na přiloženém CD. Než začnete zkoušet naše kódy pro výpočet pseudoinverzí, nainstalujte a načtěte si náš doplňkový balíček podle příkazů níže, které stačí zadat do konzole v softwaru R. Je však nutné si zadat cestu k souboru dle Vašeho počítače. Níže uvedený příkaz je proto pouze ilustrační.

```
install.packages("E:/pseudoinverze.zip",
repos = NULL, type = "win.binary")
library(pseudoinverze)
```

Pro práci s kódy v následujích kapitolách čtenáři postačí, pokud bude mít software R nainstalován, a to včetně výše zmíněného doplňkového balíčku „pseudoinverze“. Zároveň musí umět zvládnout zadat vstupní matice.

Ty lze zadat více způsoby, které můžeme, mimo jiné, nalézt v návodech zmíněných výše. Zde si však uvedeme pouze základní příkaz ”matrix”.

Příklad 1.1

Zadej do Rka matici $\mathbf{A} = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{pmatrix}$.

Řešení:

Abychom mohli zadat matici, musíme Rku říct, jak to má udělat. Prvním parametrem příkazu matrix je vektor čísel, který daná matice obsahuje. Druhým je počet řádků, třetím počet sloupců a čtvrtý nám udává, jestli se mají čísla sázet podle řádků, anebo podle sloupců. Níže uvedený příkaz je možné také zkrátit, o čemž se můžete přesvědčit v návodech k Rku.

```
> matrix(c(1, 4, 2, 5, 3, 6), nrow=2, ncol=3, byrow = FALSE)
     [,1] [,2] [,3]
[1,]    1    2    3
[2,]    4    5    6
```

Je zapotřebí si uvědomit, že danou matici si musíme uložit do tzv. „globální proměnné“. To uděláme tak, že před matici, kterou chceme uložit, dáme její název a rovnítko. Jestliže chceme matici následně zobrazit, její název vepíšeme do konzole a stiskneme tlačítko enter.

```
> A = matrix(c(1, 4, 2, 5, 3, 6), 2, 3)
> A
     [,1] [,2] [,3]
[1,]    1    2    3
[2,]    4    5    6
```

Analogicky si můžeme uložit i výstupní matici. Níže uvedeným příkazem si uložíme matici **B**, která je pseudoinverzí matice **A** počítanou za pomoci skeletního rozkladu. Kód „pinv_skeletni_rozklad“, který nám takovou pseudoinverzi spočítá je vysvětlen v podkapitole 2.1. Zde si proto pouze uvedeme, jak si matici uložíme.

```
> B = pinv_skeletni_rozklad(A)
```

2. Výpočty pseudoinverzních matic

Zde si ukážeme devět různých postupů, kterými lze spočítat pseudoinverzi matice. V každé podkapitole si nejprve ukážeme potřebný matematický aparát potřebný k jeho výpočtu. Následně si představíme výpočetní kód v softwaru R. Jako poslední vždy ukážeme příklad daného postupu. Spočítáme jej ručně a také si ukážeme, jak nám jej vypočítá Rko.

2.1. Skeletní rozklad

Prvním postupem, pro výpočet pseudoinverzních matic, který si zde ukážeme, bude výpočet za pomoci skeletního rozkladu. Jádrem této metody je rozklad matice $\mathbf{A}$ na uspořádanou dvojici matic ($\mathbf{B}$, $\mathbf{C}$), pro které platí, že matice $\mathbf{B}$ je plné sloupcové hodnosti a matice $\mathbf{C}$ plné řádkové hodnosti. Pro takové typy matic se pseudoinverze nalezne jednoduše.

Definice 2.1 Nechť $\mathbf{A}$ je nenulová matice typu m x n s hodností r. Skeletním rozkladem matice $\mathbf{A}$ nazýváme každou uspořádanou dvojici matic ($\mathbf{B}$, $\mathbf{C}$), pro kterou platí:

1. $\mathbf{B}$ je typu m x r,
2. $\mathbf{C}$ je typu r x n, kde $r = r(\mathbf{B}) = r(\mathbf{C})$ a
3. $\mathbf{A} = \mathbf{BC}$.

Poznámka 2.1 Z definice 2.1 vyplývá, že je-li matice $\mathbf{A}$ typu m x n plné řádkové, respektive plné sloupcové hodnosti, pak její skeletní rozklad může být tvořen maticí jednotkovou a maticí $\mathbf{A}$.

Věta 2.1 *Ke každé matici $\boldsymbol{A}$ typu m x n existuje alespoň jeden skeletní rozklad.*

Důkaz: Takovéto matice $\mathbf{B}$ a $\mathbf{C}$ vždy existují. Matice $\mathbf{A}$ má r lineárně nezávislých sloupců. Bez újmy na obecnosti lze předpokládat, že prvních r sloupců matice $\mathbf{A}$ je lineárně nezávislých. Pokud tomu tak není, jistě existuje permutační matice $\mathbf{P}$ taková, že prvních r sloupců matice $\mathbf{AP}$ je r lineárně nezávislých sloupců matice $\mathbf{A}$. Matici $\mathbf{AP}$ poté určíme následovně:

$\mathbf{AP} = \mathbf{B}\overline{\mathbf{C}}$, hodnost $r(\mathbf{B}) = r(\overline{\mathbf{C}}) = r$, pak $\mathbf{A} = \mathbf{BC}$, kde $\mathbf{C} = \overline{\mathbf{C}}\mathbf{P}^{-1}$ a $r = r(\mathbf{C}) = r(\overline{\mathbf{C}})$, protože $\mathbf{P}$ je regulární matice.

Tedy vezmeme-li za $\mathbf{B}$ matici typu m x r skládající se z prvních r sloupců matice $\mathbf{A}$, zbývajících $n-r$ sloupců jsou lineárními kombinacemi sloupců matice $\mathbf{B}$ tvaru $\mathbf{BQ}_j$ pro nějaký r x 1 vektor $\mathbf{Q}_j$ Položíme-li $\mathbf{Q} = (\mathbf{Q}_1, ..., \mathbf{Q}_{n-r})$, typu r x $(n-r)$. Tedy matice $\mathbf{A} = (\mathbf{B} \quad \mathbf{BQ})$. Položíme-li $\mathbf{C} = (\mathbf{I} \quad \mathbf{Q})$, kde $\mathbf{I}$ je jednotková matice řádu r, dostaneme $\mathbf{A} = \mathbf{B}(\mathbf{I} \quad \mathbf{Q}) = \mathbf{BC}$ a $r(\mathbf{B}) = r(\mathbf{C}) = r$.

Poznámka 2.2 V důkazu věty 2.1 je zároveň uveden i postup, kterým lze k libovolné matici $\mathbf{A}$ najít její skeletní rozklad. Dále z něj vyplývá, že skeletní rozklad matice vždy existuje, ale není určen jednoznačně.

Věta 2.2 *Jestliže $\boldsymbol{A} = \boldsymbol{BC}$ je skeletní rozklad matice $\boldsymbol{A}$, pak*

$$\boldsymbol{A}^+ = (\boldsymbol{BC})^+ = \boldsymbol{C}^+\boldsymbol{B}^+,$$

kde matice $\boldsymbol{B}^+$ a $\boldsymbol{C}^+$ vypočítáme pomocí výše uvedených vztahů (16) a (17).

Důkaz: Tvrzení lze snadno dokázat prověřením Penroseových axiomů.

R-file 2.1 (pinv_skeletni_rozklad.R)

Ve statistickém softwaru R jsme vytvořili kód, který najdete na přiloženém CD pod názvem „2_1_pinv_skeletni_rozklad.R“.

Tento kód obsahuje vnořenou funkci „skeletni_rozklad“, která, jak již název napovídá, vypočítá skeletní rozklad zadané matice $\mathbf{A}$. Spočítáme jej tak, že v první řadě ověříme, zda námi zadaná matice $\mathbf{A}$ není plné řádkové, případně plné sloupcové hodnosti. Jestliže tomu tak je, pak skeletní rozklad tvoří matice $\mathbf{A}$ s maticí jednotkovou. Pokud ani jedna z těchto možností nenastala, hledáme matici $\mathbf{B}$, která bude tvořena lineárně nezávislými sloupci matice $\mathbf{A}$. Jakmile ji nalezneme, matici $\mathbf{C}$ dopočteme podle důkazu věty 2.1.

Výstupem jsou matice $\mathbf{B}$ a $\mathbf{C}$, které spolu tvoří skeletní rozklad matice $\mathbf{A}$. Následně vypočítá pseudoinverzi matic $\mathbf{B}$ a $\mathbf{C}$ podle vztahů (16) a (17).

Výstupem je matice $\mathbf{A}^+$, která je pseudoinverzí matice $\mathbf{A}$.

Příklad 2.1

Vypočítejte za pomoci skeletního rozkladu matice pseudoinverzní matici $\mathbf{A}^+$ k námi zadané matici $\mathbf{A} = \begin{pmatrix} 1 & -3 & 1 & 5 & 7 \\ 1 & -1 & -1 & 1 & 1 \\ -1 & 1 & 1 & 1 & -1 \end{pmatrix}$, kde $r(\mathbf{A}) = 2$.

1. Ruční výpočet:

   V prvé řadě je zapotřebí nalézt skeletní rozklad matice $\mathbf{A}$. Uvažujme matici

$$\mathbf{B} = \begin{pmatrix} 1 & -3 \\ 1 & -1 \\ -1 & 1 \end{pmatrix}, \text{ pak } \mathbf{Q} = \begin{pmatrix} -2 & -1 & -2 \\ -1 & -2 & -3 \end{pmatrix}, \text{ a konečně tedy}$$

$$\mathbf{C} = \begin{pmatrix} 1 & 0 & -2 & -1 & -2 \\ 0 & 1 & -1 & -2 & -3 \end{pmatrix}.$$

   Matice $\mathbf{B}$ je plné sloupcové a matice $\mathbf{C}$ plné řádkové hodnosti a proto podle vztahů (16) a (17) je

$$\mathbf{B}^+ = (\mathbf{B}^T\mathbf{B})^{-1}\,\mathbf{B}^T,$$

$$\mathbf{C}^+ = \mathbf{C}^T(\mathbf{C}\mathbf{C}^T)^{-1}.$$

   Dále tedy

$$\mathbf{B}^T\mathbf{B} = \begin{pmatrix} 3 & -5 \\ -5 & 11 \end{pmatrix}, \; (\mathbf{B}^T\mathbf{B})^{-1} = \begin{pmatrix} 11/8 & 5/8 \\ 5/8 & 3/8 \end{pmatrix},$$

$$\mathbf{C}\mathbf{C}^T = \begin{pmatrix} 10 & 10 \\ 10 & 15 \end{pmatrix}, \; (\mathbf{C}\mathbf{C}^T)^{-1} = \begin{pmatrix} 3/10 & -2/10 \\ -2/10 & 2/10 \end{pmatrix},$$

$$\mathbf{A}^+ = \mathbf{C}^+\mathbf{B}^+ = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ -2 & -1 \\ -1 & -2 \\ -2 & -3 \end{pmatrix} \begin{pmatrix} 3/10 & -2/10 \\ -2/10 & 2/10 \end{pmatrix} \begin{pmatrix} 11/8 & 5/8 \\ 5/8 & 3/8 \end{pmatrix} \begin{pmatrix} 1 & 1 & -1 \\ -3 & -1 & 1 \end{pmatrix},$$

$$\mathbf{A}^{+} = \begin{pmatrix} -1/20 & 7/40 & 7/40 \\ 0 & -1/10 & 1/10 \\ 1/10 & -1/4 & 1/4 \\ 1/20 & 1/40 & -1/40 \\ 1/10 & -1/20 & 1/20 \end{pmatrix}.$$

2. Výpočet pomocí softwaru R:

```
> A=matrix(c(1,1,-1,-3,-1,1,1,-1,1,5,1,-1,7,1,-1),3,5)
> skeletni_rozklad(A)
              [,1]   [,2]   [,3]
[1,] -5.000000e-02  0.175 -0.175
[2,] -1.387779e-17 -0.100  0.100
[3,]  1.000000e-01 -0.250  0.250
[4,]  5.000000e-02  0.025 -0.025
[5,]  1.000000e-01 -0.050  0.050
```

2.2. Grevillův algoritmus

Rekurzivní[1] metodu vedoucí k nalezení pseudoinverzní matice dokázal již v roce 1960 významný matematik Thomas N. E. Grevill. Jedná se o metodu, která je založena na postupném přidávání sloupců k prvnímu sloupci naší matice $\mathbf{A}$, který označíme $\mathbf{a}_1$. Tuto metodu oceníme hlavně v případě, že máme spočítanou pseudoinverzi $\mathbf{A}^+$ k matici $\mathbf{A}$, a potřebujeme spočítat pseudoinverzi k matici $(\mathbf{A} \quad \mathbf{a}_i)$, tj. k matici $\mathbf{A}$, ke které jsme přidali další sloupec, a nebo více sloupců. Můžeme tedy plynule navázat v algoritmu a nemusíme tak celou pseudoinverzi počítat znovu.

Věta 2.3 *Nechť $\boldsymbol{A}$ je matice typu m x n. Pro $k = 1, ..., n$ označíme $\boldsymbol{a}_k$ k-tý sloupec matice $\boldsymbol{A}$ a $\boldsymbol{A}_k$ submatici tvořenou prvními k-sloupci matice $\boldsymbol{A}$. Potom pro $k > 2$ je $\boldsymbol{A}_k = (\boldsymbol{A}_{k-1} \quad \boldsymbol{a}_k)$.*

Výpočet pseudoinverzní matice prováníme následovně:

1. *Je-li $\boldsymbol{a}_1 = \boldsymbol{o}$, položíme $\boldsymbol{A}_1^+ = \boldsymbol{o}$ (nulový vektor).*

   Je-li $\boldsymbol{a}_1 \neq \boldsymbol{o}$, vypočítáme $\boldsymbol{A}_1^+ = (\boldsymbol{a}_1^T \boldsymbol{a}_1)^{-1} \ \boldsymbol{a}_1^T$.

2. *Pro $k = 2, 3, ..., n$ vypočítáme postupně:*

   (a) $\boldsymbol{d}_k = \boldsymbol{A}_{k-1}^+ \boldsymbol{a}_k,$

   $\boldsymbol{c}_k = \boldsymbol{a}_k - \boldsymbol{A}_{k-1} \boldsymbol{d}_k.$

   (b) *Je-li $\boldsymbol{c}_k \neq \boldsymbol{o}$, položíme $\boldsymbol{b}_k = \boldsymbol{c}_k^+$, což je podle (5) rovno $(\boldsymbol{c}_k^T \boldsymbol{c}_k)^{-1} \ \boldsymbol{c}_k^T$,*

   je-li $\boldsymbol{c}_k = \boldsymbol{o}$, vypočítáme $\boldsymbol{b}_k = (1 + \boldsymbol{d}_k^T \boldsymbol{d}_k)^{-1} \boldsymbol{d}_k^T \boldsymbol{A}_{k-1}^+$.

   (c) *Vypočítáme* $\boldsymbol{A}_k^+ = \begin{pmatrix} \boldsymbol{A}_{k-1}^+ - \boldsymbol{d}_k \boldsymbol{b}_k \\ \boldsymbol{b}_k \end{pmatrix}.$

Pro $k = n$ je $\boldsymbol{A}_n^+ = \boldsymbol{A}^+$ hledaná pseudoinverzní matice k dané matici $\boldsymbol{A}$.

Důkaz: viz [2].

Poznámka 2.3 V případech, kdy má námi zadaná matice více sloupců, než řádků, je výhodnější provést výpočet pro matici transponovanou a výslednou

[1] Termín „rekurzivní“ je pravděpodobně odvozen z latinského slovesa recurrō (vrátit se).

pseudoinverzi taktéž transponovat. Děláme tak proto, aby byl rychlejší výpočet, což je patrné z povahy samotného algoritmu, a protože platí $(\mathbf{A}^T)^+ = (\mathbf{A}^+)^T$.

R-file 2.2 (grevilluv_algoritmus.R)

Ve statistickém softwaru R jsme vytvořili kód, který najdete na přiloženém CD pod názvem „2_2_grevilluv_algoritmus.R“. Tento kód vypočte pseudoinverzní matici $\mathbf{A}^+$ ze zadané matice $\mathbf{A}$ pomocí grevillova algoritmu.

Příklad 2.2

Vypočítejte za pomoci Grevillova algoritmu pseudoinverzní matici $\mathbf{A}^+$ k námi zadané matici $\mathbf{A} = \begin{pmatrix} 1 & 0 \\ 1 & 0 \\ 1 & 1 \\ 1 & 1 \end{pmatrix}$.

1. Ruční výpočet:

   (a) $\mathbf{a}_1 = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}$ a $\mathbf{A}_1^+ = (\mathbf{a}_1^T\ \mathbf{a}_1)^{-1}\ \mathbf{a}_1^T = \begin{pmatrix} 1/4, & 1/4, & 1/4, & 1/4 \end{pmatrix}$.

   (b) Pro $k = 2$ dostáváme

      i. $\mathbf{d}_2 = \mathbf{A}_1^+ \mathbf{a}_2 = 1/2$,

      $$\mathbf{c}_2 = \mathbf{a}_2 - \mathbf{A}_1 \mathbf{d}_2 = \begin{pmatrix} -1/2 \\ -1/2 \\ 1/2 \\ 1/2 \end{pmatrix}.$$

      ii. $\mathbf{b}_2 = \mathbf{c}_2^+ = (\mathbf{c}_2^T\ \mathbf{c}_2)^{-1}\ \mathbf{c}_2^T = \begin{pmatrix} -1/2, & -1/2, & 1/2, & 1/2 \end{pmatrix}$.

      iii. Vypočítáme $\mathbf{A}_2^+ = \begin{pmatrix} \mathbf{A}_1^+ - \mathbf{d}_2 \mathbf{b}_2 \\ \mathbf{b}_2 \end{pmatrix} = \begin{pmatrix} 1/2 & 1/2 & 0 & 0 \\ -1/2 & -1/2 & 1/2 & 1/2 \end{pmatrix}$.

   (c) Hledaná pseudoinverze matice $\mathbf{A}$ je $\mathbf{A}^+ = \begin{pmatrix} 1/2 & -1/2 \\ 1/2 & -1/2 \\ 0 & 1/2 \\ 0 & 1/2 \end{pmatrix}$.

2. Výpočet pomocí softwaru R:

```
> A=matrix(c(1,0,1,0,1,1,1,1),2,4)
> pinv_grevilluv_algoritmus(A)
     [,1] [,2]
[1,]  0.5 -0.5
[2,]  0.5 -0.5
[3,]  0.0  0.5
[4,]  0.0  0.5
```

2.3. Příkaz PINV

Ve výpočetním softwaru MATLAB nalezneme příkaz „PINV“, který dokáže pomocí singulárního rozkladu matice spočítat pseudoinverzi. V R-ku se nám taková možnost nenaskýtá. Proto jsme vytvořili příkaz, který počítá pseudoinverzi za pomocí singulárního rozkladu matice, analogicky jako MATLABovský příkaz „PINV“.

Věta 2.4 *Nechť $\boldsymbol{A}$ je matice typu m x n a hodnosti r. Pak existují unitární matice $\boldsymbol{U}$ a $\boldsymbol{V}$ řádů m a n, takové, že $\boldsymbol{A} = \boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^T$, kde $\boldsymbol{D} = \begin{pmatrix} \boldsymbol{D}_1 & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{pmatrix}$ je typu m x n a $\boldsymbol{D}_1 = diag\,(d_1, ..., d_r)$ je regulární matice řádu r.*

Důkaz: viz [4].

Věta 2.5 *Nechť $\boldsymbol{A} = \boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^T$, kde $\boldsymbol{U}$, $\boldsymbol{V}$ a $\boldsymbol{D}$ mají vlastnosti z výše uvedené věty, tj. tvoří singulární rozklad matice $\boldsymbol{A}$.*

Potom $\boldsymbol{A}^+ = \boldsymbol{V}\boldsymbol{D}^+\boldsymbol{U}^T$, přičemž $\boldsymbol{D}^+ = \begin{pmatrix} \boldsymbol{D}_1^{-1} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{pmatrix}$.

Důkaz: Výpočtem lze ukázat, že matice $\begin{pmatrix} \mathbf{D}_1^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix}$ typu n x m je $\mathbf{D}^+$. Zbytek důkazu spočívá v prověření Penroseových axiomů pro výpočet pseudoinverzní matice.

Samotný postup výpočtu singulárního rozkladu by neúměrně zvětšil rozsah této práce, a proto jej zde neuvádíme. Můžete jej však nalézt například v [6].

R-file 2.3 (analogie_prikazu_pinv.R)

Ve statistickém softwaru R jsme vytvořili kód, který najdete na přiloženém CD pod názvem „2_3_analogie_prikazu_pinv.R“. Tento kód vypočte pseudoinverzní matici $\mathbf{A}^+$ ze zadané matice $\mathbf{A}$ pomocí singulárního rozkladu matice.

Daný kód je převzat a poupraven z doplňkového balíčku „pracma“, ze stránek `https://cran.r-project.org/web/packages/pracma/pracma.pdf`.

Příklad 2.3

Vypočítejte za pomoci analogie příkazu „PINV“ pseudoinverzní matici $\mathbf{A}^+$ k námi zadané matici $\mathbf{A} = \begin{pmatrix} 1 & 1 & 1 & -1 \\ 0 & 0 & 1 & 1 \end{pmatrix}$.

1. Ruční výpočet:

   $\mathbf{A} = \mathbf{U}\mathbf{D}\mathbf{V}^T$, kde

$$\mathbf{U} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \mathbf{D} = \begin{pmatrix} 2 & 0 & 0 & 0 \\ 0 & \sqrt{2} & 0 & 0 \end{pmatrix},$$

$$\mathbf{V}^T = \begin{pmatrix} 1/2 & 1/2 & 1/2 & -1/2 \\ 0 & 0 & 1/\sqrt{2} & 1/\sqrt{2} \\ \sqrt{2/3} & 0 & -1/\sqrt{6} & 1/\sqrt{6} \\ -1/(2\sqrt{3}) & \sqrt{3}/2 & -1/(2\sqrt{3}) & 1/(2\sqrt{3}) \end{pmatrix}.$$

   Tedy

$$\mathbf{D}_1 = \begin{pmatrix} 2 & 0 \\ 0 & \sqrt{2} \end{pmatrix}, \text{ a proto je } \mathbf{D}_1^{-1} = \begin{pmatrix} 1/2 & 0 \\ 0 & 1/\sqrt{2} \end{pmatrix}.$$

   Potom je

$$\mathbf{A}^+ = \mathbf{V}\mathbf{D}^+\mathbf{U}^T = \begin{pmatrix} 1/4 & 0 \\ 1/4 & 0 \\ 1/4 & 1/2 \\ -1/4 & 1/2 \end{pmatrix}, \text{ kde } \mathbf{D}^+ = \begin{pmatrix} 1/2 & 0 \\ 0 & 1/\sqrt{2} \\ 0 & 0 \\ 0 & 0 \end{pmatrix}.$$

2. Výpočet pomocí softwaru R:

```
> A=matrix(c(1,0,1,0,1,1,-1,1),2,4)
> analogie_prikazu_pinv(A)
      [,1] [,2]
[1,]  0.25  0.0
[2,]  0.25  0.0
[3,]  0.25  0.5
[4,] -0.25  0.5
```

2.4. Zhukovskii - Liptserova metoda

Tato metoda vznikla zobecněním výpočtu inverzní matice. Je založena na řešení maticové rovnice $\mathbf{AX} = \mathbf{P}$, kde $\mathbf{P} = \mathbf{AA}^+$ se najde rekurzivně.

Nechť je dána matice $\mathbf{A}$. Doplníme ji nulami na čtvercovou matici $\mathbf{A}_0$. Označme $\mathbf{a}_k$ k-tý řádek matice $\mathbf{A}_0$ a $\mathbf{b}_k$ k-tý sloupec matice $\mathbf{A}_0$.

Výpočet provádíme podle následujícího algoritmu:

1. Pro $k = 0, 1, ..., n-1$ konstruujeme čtvercové matice $\mathbf{M}_k$ řádu n takto:

   (a) $\mathbf{M}_0 = \mathbf{I}$, kde $\mathbf{I}$ je jednotková matice řádu n,

   (b) $\mathbf{M}_{k+1} = \mathbf{M}_k - \mathbf{M}_k\mathbf{b}_{k+1}(\mathbf{b}_{k+1}^T\mathbf{M}_k\mathbf{b}_{k+1})^+ \ \mathbf{b}_{k+1}^T\mathbf{M}_k$.

2. Položíme

   (a) $\mathbf{C} = \mathbf{I} - \mathbf{M}_n$, kde $\mathbf{I}$ je jednotková matice řádu n a označíme $\mathbf{c}_k$ k-tý řádek matice $\mathbf{C}$.

3. Počítáme postupně matice $\mathbf{N}_k$ a $\mathbf{X}_k$ řádu n pro $k = 0, 1, ..., n-1$ takto:

   (a) $\mathbf{N}_0 = \mathbf{I}$,

   (b) $\mathbf{N}_{k+1} = \mathbf{N}_k - \mathbf{N}_k\mathbf{a}_{k+1}^T(\mathbf{a}_{k+1}\mathbf{N}_k\mathbf{a}_{k+1}^T)^+\mathbf{a}_{k+1}\mathbf{N}_k$,

   (c) $\mathbf{X}_0 = \mathbf{a}_1^T$,

   (d) vektor $\mathbf{X}_0$ doplníme nulami na čtvercovou matici řádu n, a další matice dopočítáme následovně:

   $\mathbf{X}_{k+1} = \mathbf{X}_k + \mathbf{N}_k\mathbf{a}_{k+1}^T(\mathbf{a}_{k+1}\mathbf{N}_k\mathbf{a}_{k+1}^T)^+(\mathbf{c}_{k+1} - \mathbf{a}_{k+1}\mathbf{X}_k)$.

Pro $k = n$ je $\mathbf{X}_n = \mathbf{A}^+$, ovšem s vynecháním přidaných sloupců a řádků.

R-file 2.4 (zhukovskii_liptserova_metoda.R)

Ve statistickém softwaru R jsme vytvořili kód, který najdete na přiloženém CD pod názvem „2_4_zhukovskii_liptserova_metoda.R“. Tento kód vypočte pseudoinverzní matici $\mathbf{A}^+$ ze zadané matice $\mathbf{A}$ pomocí Zhukovskii – Liptserovy metody.

Příklad 2.4

Vypočítejte za pomoci Zhukovskii – Liptserovy metody pseudoinverzní matici $\mathbf{A}^+$ k námi zadané matici $\mathbf{A} = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \end{pmatrix}$.

1. Ruční výpočet:

   Matici $\mathbf{A}$ doplníme nulami na čtvercovou matici $\mathbf{A}_0 = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$.

   (a) Položme $\mathbf{M}_0 = \mathbf{I}$ řádu 4.

   Pro $k = 0, 1, 2, 3$ vypočítáme matice $\mathbf{M}_{k+1}$.

   Připomeňme, že výraz $\mathbf{b}_{k+1}^T \mathbf{M}_k \mathbf{b}_{k+1}$ je skalár a proto

   $$(\mathbf{b}_{k+1}^T \mathbf{M}_k \mathbf{b}_{k+1})^+ = \begin{cases} 1/(\mathbf{b}_{k+1}^T \mathbf{M}_k \mathbf{b}_{k+1}) & \text{pokud } \mathbf{b}_{k+1}^T \mathbf{M}_k \mathbf{b}_{k+1} \neq 0, \\ 0 & \text{pokud } \mathbf{b}_{k+1}^T \mathbf{M}_k \mathbf{b}_{k+1} = 0. \end{cases}$$

   Tedy podle výše uvedených vzorců je $\mathbf{M}_1 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$.

   Dále $\mathbf{M}_2 = \mathbf{M}_1$, neboť $\mathbf{b}_2^T \mathbf{M}_1 \mathbf{b}_2 = 0$ a $\mathbf{M}_3 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$.

   A konečně $\mathbf{M}_4 = \mathbf{M}_3$, neboť $\mathbf{b}_4^T \mathbf{M}_3 \mathbf{b}_4 = 0$.

   (b) Položíme $\mathbf{C} = \mathbf{I} - \mathbf{M}_4 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$.

   (c) Položíme $\mathbf{N}_0 = \mathbf{I}$ a $\mathbf{X}_0 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix}$.

Pro $k = 0,1,2,3$ vypočítáme podle výše uvedených vzorců matice $\mathbf{N}_{k+1}$ a $\mathbf{X}_{k+1}$. Máme tedy

$$\mathbf{N}_1 = \begin{pmatrix} 3/4 & -1/4 & -1/4 & -1/4 \\ -1/4 & 3/4 & -1/4 & -1/4 \\ -1/4 & -1/4 & 3/4 & -1/4 \\ -1/4 & -1/4 & -1/4 & 3/4 \end{pmatrix} \text{ a } \mathbf{X}_0 = \begin{pmatrix} 1/4 & 0 & 0 & 0 \\ 1/4 & 0 & 0 & 0 \\ 1/4 & 0 & 0 & 0 \\ 1/4 & 0 & 0 & 0 \end{pmatrix}.$$

Dále

$$\mathbf{N}_2 = \begin{pmatrix} 1/2 & -1/2 & 0 & 0 \\ -1/2 & 1/2 & 0 & 0 \\ 0 & 0 & 1/2 & -1/2 \\ 0 & 0 & -1/2 & 1/2 \end{pmatrix} \text{ a } \mathbf{X}_2 = \begin{pmatrix} 1/2 & -1/2 & 0 & 0 \\ 1/2 & -1/2 & 0 & 0 \\ 0 & 1/2 & 0 & 0 \\ 0 & 1/2 & 0 & 0 \end{pmatrix}.$$

Matice $\mathbf{N}_3 = \mathbf{N}_2$, neboť $\mathbf{a}_3\mathbf{N}_2\mathbf{a}_3^T = 0$. Podobně $\mathbf{X}_3 = \mathbf{X}_2$, $\mathbf{N}_4 = \mathbf{N}_3$. A konečně $\mathbf{X}_4 = \mathbf{X}_3, \mathbf{A}_0^+ = \mathbf{X}_4$.

$$\mathbf{A}^+ = \begin{pmatrix} 1/2 & -1/2 \\ 1/2 & -1/2 \\ 0 & 1/2 \\ 0 & 1/2 \end{pmatrix}.$$

2. Výpočet pomocí softwaru R:

```
> A=matrix(c(1,0,1,0,1,1,1,1),2,4)
> zhukovskii_liptserova_metoda(A)
     [,1] [,2]
[1,]  0.5 -0.5
[2,]  0.5 -0.5
[3,]  0.0  0.5
[4,]  0.0  0.5
```

2.5. Biortogonalizační metoda

Biortogonalizační metoda je založena na následujícím principu:

Jsou dány dvě uspořádané n-tice lineárně nezávislých vektorů $\mathbf{u}_i$ a $\mathbf{v}_i$, kde $i = 1, ..., n$. Pak n-tici $\mathbf{v}_1, ..., \mathbf{v}_n$ transformujeme postupně v n krocích tak, aby výsledná n-tice vektorů $\mathbf{v}_1^{(n)}, ..., \mathbf{v}_n^{(n)}$ tvořila s vektory $\mathbf{u}_1, ..., \mathbf{u}_n$ biortogonální systém.

Nechť je dána matice $\mathbf{A}$ typu m x n. K matici $\mathbf{A}$ připojíme p řádků, které jsou ortogonální ke všem řádkům matice $\mathbf{A}$ a jejichž připojením vznikne matice $\mathbf{U}$ s hodností n. Tuto matici budeme upravovat tak, abychom dostali matici $\mathbf{V}^{(n)}$, jejíž řádky tvoří se sloupci matice $\mathbf{U}$ biortogonální systém. Prvních m sloupců matice $\mathbf{V}^{(n)}$ tvoří pak hledanou pseudoinverzi matice $\mathbf{A}$.

Posloupnost matic, které přitom vytvoříme, budeme značit $\mathbf{V}^{(k)}$ pro $k = 0, 1, ..., n$. Označme $\mathbf{u}_j$ j-tý sloupec matice $\mathbf{U}$ a $\mathbf{v}_i^k$ i-tý řádek matice $\mathbf{V}^{(k)}$.

Algoritmus výpočtu:

1. Z matice $\mathbf{A}$ utvoříme matici $\mathbf{U}$.
2. Položíme $\mathbf{V}^{(0)} = \mathbf{U}^T$.
3. Pro $k = 1, ..., n$ provádíme kroky:
   (a) Vytvoříme matici $\mathbf{C}^{(k)}$ s prvky $\mathbf{c}_{ij}^{(k)}$ takto:

   $$\mathbf{c}_{kk}^{(k)} = (\mathbf{v}_k^{(k-1)}, \mathbf{u}_k)^{-1},$$

   $$\mathbf{c}_{ik}^{(k)} = -(\mathbf{v}_i^{(k-1)}, \mathbf{u}_k)\mathbf{c}_{kk}^{(k)} \qquad \text{pro } i = 1, ..., n \quad \text{a} \quad i \neq k,$$

   $$\mathbf{c}_{ij}^{(k)} = \delta_{ij}, \qquad \text{pro } i = 1, ..., n \quad \text{a} \quad j = 1, ..., n \quad \text{a} \quad j \neq k.$$

   (b) $\mathbf{V}^{(k)} = \mathbf{C}^{(k)}\mathbf{V}^{(k-1)}$.

Pro $k = n$ pak platí $\mathbf{V}^{(n)}\mathbf{U} = \mathbf{I}$.

Matice $\mathbf{A}^+$ je tvořena prvními m sloupci matice $\mathbf{V}^{(n)}$.

Poznámka 2.4 V případech, kdy má námi zadaná matice více sloupců, než řádků, je výhodnější provést výpočet pro matici transponovanou a výslednou pseudoinverzi taktéž transponovat. Děláme tak proto, aby byl rychlejší výpočet, což je patrné z povahy samotného algoritmu, a protože platí $(\mathbf{A}^T)^+ = (\mathbf{A}^+)^T$.

R-file 2.5 (biortogonalizacni_metoda.R)

Ve statistickém softwaru R jsme vytvořili kód, který najdete na přiloženém CD pod názvem „2_5_biortogonalizacni_metoda.R“. Tento kód vypočte pseudoinverzní matici $\mathbf{A}^+$ ze zadané matice $\mathbf{A}$ pomocí biortogonalizační metody.

Příkaz samotný obsahuje dvě vnořené funkce. První s názvem „ortodoplnek“ nám vytvoří z matice $\mathbf{A}$ matici $\mathbf{U}$ s hodností n, což je biortogonální systém. Daný kód je převzat a poupraven z doplňkového balíku „MASS“ze stránek `https://cran.r-project.org/web/packages/MASS/MASS.pdf`.

Druhá, s názvem „biortogonalizacni“, obsahuje samotný algoritmus biortogonalizační metody. Příkaz samotný zastřešuje obě zmíněné funkce a jeho úlohou je zjistit u vstupní matice $\mathbf{A}$ její rozměry a hodnost a následně s využitím obou výše zmíněných funkcí vypočítat její pseudoinverzi $\mathbf{A}^+$.

Příklad 2.5

Vypočítejte za pomoci Biortogonalizační metody pseudoinverzní matici $\mathbf{A}^+$ k námi zadané matici $\mathbf{A} = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \end{pmatrix}$, $r(\mathbf{A}) = 2$.

1. Ruční výpočet:

   Dle poznámky 2.4 bude výhodnější provést výpočet pro matici transponovanou. Položme tedy $\mathbf{B} = \mathbf{A}^T = \begin{pmatrix} 1 & 0 \\ 1 & 0 \\ 1 & 1 \\ 1 & 1 \end{pmatrix}$.

   Dále pokračujeme podle algoritmu:

   (a) Protože $r(\mathbf{B}) = r(\mathbf{A}) = 2$ je $\mathbf{U} = \mathbf{B}$.

   (b) Položme $\mathbf{V}^{(0)} = \mathbf{U}^T = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \end{pmatrix}$.

Pro $k=1$ je $\mathbf{c}_{11}^{(1)} = 1/4$ a $\mathbf{c}_{21}^{(1)} = -1/2$.

A tedy matice

$$\mathbf{C}^{(1)} = \begin{pmatrix} 1/4 & 0 \\ -1/2 & 1 \end{pmatrix},$$

$$\mathbf{V}^{(1)} = \mathbf{C}^{(1)}\mathbf{V}^{(0)} = \begin{pmatrix} 1/4 & 1/4 & 1/4 & 1/4 \\ -1/2 & -1/2 & 1/2 & 1/2 \end{pmatrix}.$$

Pro $k=2$ je $\mathbf{c}_{22}^{(2)} = 1$ a $\mathbf{c}_{12}^{(2)} = -1/2$.

A tedy matice

$$\mathbf{C}^{(2)} = \begin{pmatrix} 1 & -1/2 \\ 0 & 1 \end{pmatrix},$$

$$\mathbf{V}^{(2)} = \mathbf{C}^{(2)}\mathbf{V}^{(1)} = \begin{pmatrix} 1/2 & 1/2 & 0 & 0 \\ -1/2 & -1/2 & 1/2 & 1/2 \end{pmatrix}.$$

Protože jsme k matici $\mathbf{B}$ nepřipojili žádný řádek je $\mathbf{B}^+ = \mathbf{V}^{(2)}$, a tedy

$$\mathbf{A}^+ = (\mathbf{B}^T)^+ = (\mathbf{B}^+)^T = \begin{pmatrix} 1/2 & -1/2 \\ 1/2 & -1/2 \\ 0 & 1/2 \\ 0 & 1/2 \end{pmatrix}.$$

2. Výpočet pomocí softwaru R:

```
> A=matrix(c(1,0,1,0,1,1,1,1),2,4)
> biortogonalizacni_metoda(A)
     [,1] [,2]
[1,]  0.5 -0.5
[2,]  0.5 -0.5
[3,]  0.0  0.5
[4,]  0.0  0.5
```

2.6. Metoda gradientních projekcí

Další postup, který si představíme, je metoda gradientních projekcí. Celá tato metoda je založena na následující větě:

Věta 2.6 *Nechť matice* $\boldsymbol{A}$ *je typu* m x n. *Označme* $\boldsymbol{a}_1, ..., \boldsymbol{a}_m$ *sloupce matice* $\boldsymbol{A}^T$. *Tedy* $\boldsymbol{A}^T = (\boldsymbol{a}_1, ..., \boldsymbol{a}_m)$. *Dále nechť* $\boldsymbol{b} = (\beta_1, ..., \beta_m)^T \in \Re(\boldsymbol{A})$, *kde* $\Re(\boldsymbol{A})$ *značí množinu obrazů matice* $\boldsymbol{A}$, *tedy množinu vektorů*

$$\Re(\boldsymbol{A}) = \{\boldsymbol{z} : \boldsymbol{z} = \boldsymbol{A}\boldsymbol{x} \textit{ pro nějaké } \boldsymbol{x} \in \mathbb{R}^n\}.$$

Pak soustava rovnic $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$ *má řešení* $\boldsymbol{x}_m = \boldsymbol{A}^+\boldsymbol{b}$ *takové, že* $\boldsymbol{x}_m \in \Re(\boldsymbol{A})$, *kde*

$$\boldsymbol{x}_0 = \boldsymbol{o} \textit{ (nulový vektor)}$$
$$\boldsymbol{x}_k = \boldsymbol{x}_{k-1} + \alpha_k \boldsymbol{h}_k \textit{ pro } k = 1, ..., m,$$

kde

$$\alpha_k = \begin{cases} 0 & \textit{pokud} \quad \boldsymbol{h}_k = \boldsymbol{o}, \\ (\beta_k - \boldsymbol{a}_k^T \boldsymbol{x}_{k-1}) \ / \ (\boldsymbol{h}_k^T \boldsymbol{a}_k) & \textit{pokud} \quad \boldsymbol{h}_k \neq \boldsymbol{o}, \end{cases}$$

a $\{\boldsymbol{h}_1, ..., \boldsymbol{h}_m\}$ *jsou vektory získané ortogonalizací vektorů* $\{\boldsymbol{a}_1, ..., \boldsymbol{a}_m\}$.

Pokud ale $\boldsymbol{b} \notin \Re(\boldsymbol{A})$, *položíme* $\boldsymbol{d} = \boldsymbol{A}^T\boldsymbol{b}$, $\boldsymbol{C} = \boldsymbol{A}^T\boldsymbol{A}$. *Potom* $\boldsymbol{d} \in \Re(\boldsymbol{A}) = \Re(\boldsymbol{C})$. *Výše uvedený algoritmus nyní aplikujeme na výpočet* $\boldsymbol{C}^+\boldsymbol{d}$.

Důkaz: viz [1].

Tuto větu nyní použijeme pro výpočet $\mathbf{A}^+$:

Užitím ortonormalizace sloupců matice $\mathbf{A}$ získáme vektory $\mathbf{d}_1, ..., \mathbf{d}_r$. Jelikož sloupce matice $\mathbf{A}$ a vektory $\mathbf{d}_1, ..., \mathbf{d}_r$ generují tentýž podprostor, platí, že:

$$\mathbf{A}\mathbf{A}^+ = \textstyle\sum_{j=1}^{r} \mathbf{d}_j \mathbf{d}_j^T.$$

Označme nyní $\mathbf{b}_1, ..., \mathbf{b}_m$ sloupce matice $\mathbf{A}\mathbf{A}^+$. Pro každé $j = 1, ..., m$ je $\mathbf{b}_j \in \Re(\mathbf{A})$. Nyní lze použít výše uvedený m-krokový algoritmus pro výpočet $\mathbf{A}^+\mathbf{b}_j$ pro každé $j = 1, ..., m$. Pak platí, že:

$$(\mathbf{A}^+\mathbf{b}_1, ..., \mathbf{A}^+\mathbf{b}_m) = \mathbf{A}^+(\mathbf{b}_1, ..., \mathbf{b}_m) = \mathbf{A}^+(\mathbf{A}\mathbf{A}^+) = \mathbf{A}^+.$$

R-file 2.6 (metoda_gradientnich_projekci.R)

Ve statistickém softwaru R jsme vytvořili kód, který najdeme na přiloženém CD pod názvem „2_6_metoda_gradientnich_projekci“. Tento kód vypočte pseudoinverzní matici $\mathbf{A}^+$ ze zadané matice $\mathbf{A}$ za pomoci metody gradientních projekcí.

Příkaz samotný obsahuje vnořenou funkci s názvem „gramschmidt“, která nám, pomocí Gramovy – Schmidtovy ortogonalizace, vytvoří matici ortogonální, či ortonormální. Samotný výpočet je popsán již na předcházející straně.

Příklad 2.6

Vypočítejte za pomoci metody gradientních projekcí pseudoinverzní matici $\mathbf{A}^+$ k námi zadané matici $\mathbf{A} = \begin{pmatrix} 1 & 1 & 1 & 1 \\ -1 & 0 & -1 & 0 \\ 0 & 1 & 0 & 1 \end{pmatrix}$, $r(\mathbf{A}) = 2$.

1. Ruční výpočet:

   Ortonormalizací sloupců matice $\mathbf{A}$ získáme sloupce:

$$\mathbf{d}_1 = \begin{pmatrix} 1/\sqrt{2} \\ -1/\sqrt{2} \\ 0 \end{pmatrix}, \mathbf{d}_2 = \begin{pmatrix} 1/\sqrt{6} \\ 1/\sqrt{6} \\ 2/\sqrt{6} \end{pmatrix}, \mathbf{d}_3 = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}, \mathbf{d}_4 = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}.$$

   Potom $\mathbf{A}\mathbf{A}^+ = \mathbf{d}_1\mathbf{d}_1^T + \mathbf{d}_2\mathbf{d}_2^T = \begin{pmatrix} 2/3 & -1/3 & 1/3 \\ -1/3 & 2/3 & 1/3 \\ 1/3 & 1/3 & 2/3 \end{pmatrix}$.

   Ortogonalizací sloupců matice $\mathbf{A}^T$ získáme vektory:

$$\mathbf{h}_1 = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}, \mathbf{h}_2 = \begin{pmatrix} -1/2 \\ 1/2 \\ -1/2 \\ 1/2 \end{pmatrix}, \mathbf{h}_3 = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}.$$

Výpočet $\mathbf{A}^+\mathbf{b}_1$:

$$\alpha_1 = 1/6,\ \mathbf{x}_1 = \begin{pmatrix} 1/6 \\ 1/6 \\ 1/6 \\ 1/6 \end{pmatrix},\ \alpha_2 = 0,\ \mathbf{x}_2 = \begin{pmatrix} 1/6 \\ 1/6 \\ 1/6 \\ 1/6 \end{pmatrix},\ \text{kde } \mathbf{x}_2 = \mathbf{A}^+\mathbf{b}_1.$$

Výpočet $\mathbf{A}^+\mathbf{b}_2$:

$$\alpha_1 = -1/12,\ \mathbf{x}_1 = \begin{pmatrix} -1/12 \\ -1/12 \\ -1/12 \\ -1/12 \end{pmatrix},\ \alpha_2 = 1/2,\ \mathbf{x}_2 = \begin{pmatrix} -1/3 \\ 1/6 \\ -1/3 \\ 1/6 \end{pmatrix},\ \text{kde } \mathbf{x}_2 = \mathbf{A}^+\mathbf{b}_1.$$

Výpočet $\mathbf{A}^+\mathbf{b}_3$:

$$\alpha_1 = 1/12,\ \mathbf{x}_1 = \begin{pmatrix} 1/12 \\ 1/12 \\ 1/12 \\ 1/12 \end{pmatrix},\ \alpha_2 = 1/2,\ \mathbf{x}_2 = \begin{pmatrix} -1/6 \\ 1/3 \\ -1/6 \\ 1/3 \end{pmatrix},\ \text{kde } \mathbf{x}_2 = \mathbf{A}^+\mathbf{b}_3.$$

Celkem tedy je $\mathbf{A}^+ = (\mathbf{A}^+\mathbf{b}_1, \mathbf{A}^+\mathbf{b}_2, \mathbf{A}^+\mathbf{b}_3)$, kde

$$\mathbf{A}^+ = \begin{pmatrix} 1/6 & -1/3 & -1/6 \\ 1/6 & 1/6 & 1/3 \\ 1/6 & -1/3 & -1/6 \\ 1/6 & 1/6 & 1/3 \end{pmatrix}.$$

2. Výpočet pomocí softwaru R:

```
> A=matrix(c(1,-1,0,1,0,1,1,-1,0,1,0,1),3,4)
> metoda_gradientnich_projekci(A)
          [,1]       [,2]       [,3]
[1,] 0.1666667 -0.3333333 -0.1666667
[2,] 0.1666667  0.1666667  0.3333333
[3,] 0.1666667 -0.3333333 -0.1666667
[4,] 0.1666667  0.1666667  0.3333333
```

2.7. Gramm – Schmidtova ortogonalizační metoda

Je dána matice $\mathbf{A}$ typu m x n a hodnosti r. Bez újmy na obecnosti lze předpokládat, že matice $\mathbf{A}$ má prvních r sloupců lineárně nezávislých. Pokud tomu tak není, jistě existuje permutační matice $\mathbf{P}$, taková, že

$$\mathbf{AP} = (\mathbf{R} \quad \mathbf{S}),$$

kde $\mathbf{R}$ je matice typu m x r hodnosti r a sloupce matice $\mathbf{S}$ jsou lineárními kombinacemi sloupců matice $\mathbf{R}$, tj. $\mathbf{S} = \mathbf{RU}$ pro nějakou matici $\mathbf{U}$. Vzhledem k tomu, že permutační matice $\mathbf{P}$ je ortogonální, je

$$\mathbf{APP}^T = \mathbf{A} = (\mathbf{R} \quad \mathbf{S})\mathbf{P}^T = (\mathbf{R} \quad \mathbf{RU})\mathbf{P}^T$$

a

$$\mathbf{A}^+ = \mathbf{P}[\mathbf{R}(\mathbf{I} \quad \mathbf{U})]^+.$$

Dále $[\mathbf{R}(\mathbf{I} \quad \mathbf{U})]^+ = (\mathbf{I} \quad \mathbf{U})^+\mathbf{R}^+$, neboť matice $\mathbf{R}$ a $(\mathbf{I} \quad \mathbf{U})$ tvoří skeletní rozklad matice $\mathbf{AP}$. Matice $(\mathbf{I} \quad \mathbf{U})$ je plné řádkové hodnosti a podle (16) je:

$$(\mathbf{I} \quad \mathbf{U})^+ = (\mathbf{I} \quad \mathbf{U})^T(\mathbf{I} + \mathbf{UU}^T)^{-1},$$

tedy

$$\mathbf{A}^+ = \mathbf{P}(\mathbf{I} \quad \mathbf{U})^T(\mathbf{I} + \mathbf{UU}^T)^{-1}\mathbf{R}^+.$$

Výpočet:

1. Provedeme ortogonalizaci sloupců matice $\mathbf{A} = (\mathbf{a}_1, ..., \mathbf{a}_n)$ a najdeme permutační matici $\mathbf{P}$. Obdržíme tedy vektory $\mathbf{c}_1, ..., \mathbf{c}_n$ takové, že

$$\|\mathbf{c}_j\| = \begin{cases} > 0 & \text{pro } j = 1, ..., r, \\ = 0 & \text{pro } j = r+1, ..., n. \end{cases}$$

2. Nenulové vektory $\mathbf{c}_i$ znormalizujeme a položíme

$$\mathbf{Q} = \left(\frac{\mathbf{c}_1}{\|\mathbf{c}_1\|}, ..., \frac{\mathbf{c}_r}{\|\mathbf{c}_r\|} \right).$$

Existuje tedy matice $\mathbf{B}$ taková, že $\mathbf{RB} = \mathbf{Q}$. Vynásobíme-li tuto rovnost zleva maticí $\mathbf{R}^T$ obdržíme

$$\mathbf{R}^T\mathbf{RB} = \mathbf{R}^T\mathbf{Q}.$$

Pak

$$\mathbf{B} = (\mathbf{R}^T\mathbf{R})^{-1}\mathbf{R}^T\mathbf{Q}.$$

Matice $\mathbf{Q}$ je ortonormální, tj. $\mathbf{Q}^T\mathbf{Q} = \mathbf{I}$, $\mathbf{B}$ je regulární a proto

$$\mathbf{I} = \mathbf{Q}^T\mathbf{Q} = \mathbf{Q}^T\mathbf{RB} \Rightarrow \mathbf{R} = \mathbf{QB}^{-1}.$$

Nyní použijeme následujícího tvrzení (Důkaz: viz [1].):

Rovnost $(\mathbf{AB})^+ = \mathbf{B}^+\mathbf{A}^+$ platí, jestliže $\mathbf{A}^T\mathbf{A} = \mathbf{I}$.

Odtud a z (15) tedy dostáváme, že

$$\mathbf{R}^+ = \mathbf{BQ}^T = \mathbf{B}(\mathbf{Q}^T\mathbf{Q})^{-1}\mathbf{Q}^T = \mathbf{BQ}^T.$$

3. Označme $\mathbf{R} = (\mathbf{r}_1, ..., \mathbf{r}_r)$ a $\mathbf{S} = (\mathbf{s}_1, ..., \mathbf{s}_{n-r})$. Vektory $\mathbf{c}_1, ..., \mathbf{c}_n$ jsou ortogonální, získané ortogonalizací vektorů $\mathbf{r}_1, ..., \mathbf{r}_r, \mathbf{s}_1, ..., \mathbf{s}_{n-r}$, tj.:

$$\begin{aligned} \mathbf{c}_1 &= \mathbf{r}_1, \\ \mathbf{c}_j &= \mathbf{r}_j - \textstyle\sum_{i=1}^{j-1} \frac{\mathbf{r}_i^T\mathbf{c}_j}{\|\mathbf{c}_i\|^2}\mathbf{c}_i \quad \text{pro } j = 2, ..., r, \\ 0 = \mathbf{c}_{r+j} &= \mathbf{s}_j - \textstyle\sum_{i=1}^{r} \frac{\mathbf{s}_j^T\mathbf{c}_i}{\|\mathbf{c}_i\|^2}\mathbf{c}_i \quad \text{pro } j = 1, ..., m-r. \end{aligned}$$

Odtud dostáváme, že

$$\mathbf{c}_j = \textstyle\sum_{i=1}^{j} \gamma_{ij}\mathbf{r}_i \quad \text{pro } j = 1, ..., r,$$

kde

$$\gamma_{ij} = \begin{cases} 0 & \text{pokud} \quad i > j, \\ 1 & \text{pokud} \quad i = j, \\ -\sum_{\alpha=i}^{j-1} \frac{\mathbf{r}_j^T \mathbf{c}_\alpha}{\|\mathbf{c}_\alpha\|^2} \gamma_{i\alpha} & \text{pokud} \quad i < j, \end{cases}$$

a

$$\mathbf{s}_j = \sum_{i=1}^{r} \omega_{ij} \mathbf{r}_i,$$

kde ω_{ij} určíme z toho, že

$$\mathbf{s}_j = \sum_{\alpha=1}^{r} \frac{\mathbf{s}_j^T \mathbf{c}_\alpha}{\|\mathbf{c}_\alpha\|^2} \left(\sum_{i=1}^{\alpha} \gamma_{i\alpha} \mathbf{r}_i\right) = \sum_{i=1}^{r} \left(\sum_{\alpha=i}^{r} \frac{\mathbf{s}_j^T \mathbf{c}_\alpha}{\|\mathbf{c}_\alpha\|^2} \gamma_{i\alpha}\right) \mathbf{r}_i,$$

$$\omega_{ij} = \sum_{\alpha=i}^{r} \frac{\mathbf{s}_j^T \mathbf{c}_\alpha}{\|\mathbf{c}_\alpha\|^2} \gamma_{i\alpha},$$

$$\text{pro } i = 1, ..., r \text{ a } j = 1, ..., n - r.$$

Prvky matice $\mathbf{B}$ jsou tvaru

$$\beta_{ij} = \frac{\gamma_{ij}}{\|\mathbf{c}_j\|},$$

a matice $\mathbf{U}$ má prvky ω_{ij}.

4. Výpočet $(\mathbf{I} + \mathbf{U}\mathbf{U}^T)^{-1}$ se provede podle následující věty:

Věta 2.7 *Nechť $\boldsymbol{U}$ je matice typu k x r, a nechť matice tvaru $\begin{pmatrix} \boldsymbol{V}_1 \\ \boldsymbol{V}_2 \end{pmatrix}$ je matice získaná Gramm – Schmidtovou ortonormalizací sloupců matice $\begin{pmatrix} \boldsymbol{U} \\ \boldsymbol{I} \end{pmatrix}$, kde $\boldsymbol{I}$ je jednotková matice řádu r. Pak platí:*

$$\boldsymbol{V}_2 \boldsymbol{V}_2^T = (\boldsymbol{I} + \boldsymbol{U}^T \boldsymbol{U})^{-1},$$

$$\boldsymbol{I} - \boldsymbol{V}_1 \boldsymbol{V}_1^T = (\boldsymbol{I} + \boldsymbol{U}\boldsymbol{U}^T)^{-1}.$$

Důkaz: viz [1].

R-file 2.7 (gramm_schmidtova_ortogonalizacni_metoda.R)

Ve statistickém softwaru R jsme vytvořili kód, který najdeme na přiloženém CD pod názvem „2_7_gramm_schmidtova_ortogonalizacni_metoda". Tento kód vypočte pseudoinverzní matici $\mathbf{A}^+$ ze zadané matice $\mathbf{A}$ za pomoci Gramm – Schmidtovy ortogonalizační metody.

Příkaz samotný obsahuje vnořenou funkci s názvem „gramschmidt", která nám, pomocí Gramovy – Schmidtovy ortogonalizace, vytvoří matici ortogonální, či ortonormální.

Dále obsahuje vnořenou funkci s názvem „muj_vypocet_vektoru_q", která seřadí sloupce matice $\mathbf{A}$ do takového tvaru, že jako první jsou sloupce nezávislé a až za nimi jsou všechny závislé sloupce (myšleno ve směru zleva doprava).

S pomocí těchto dvou vnořených funkcí již jsme schopni spočítat pseudoinverzi přesně podle postupu této metody.

Příklad 2.7

Vypočítejte za pomoci Gramm – Schmidtovy ortogonalizační metody pseudoinverzní matici $\mathbf{A}^+$ k námi zadané matici $\mathbf{A} = \begin{pmatrix} 1 & 1 & 1 & 1 \\ -1 & 0 & -1 & 0 \\ 0 & 1 & 0 & 1 \end{pmatrix}$, $r(\mathbf{A}) = 2$.

1. Ruční výpočet:

   (a) Ortogonalizací sloupců matice $\mathbf{A}$ dostaneme vektory:

   $$\mathbf{c}_1 = \begin{pmatrix} 1 \\ -1 \\ 0 \end{pmatrix}, \mathbf{c}_2 = \begin{pmatrix} 1/2 \\ 1/2 \\ 1 \end{pmatrix}, \mathbf{c}_3 = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}, \mathbf{c}_4 = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}.$$

   Permutační matice $\mathbf{P} = \mathbf{I}$.

   (b) Položme

   $$\mathbf{R} = \begin{pmatrix} 1 & 1 \\ -1 & 0 \\ 0 & 1 \end{pmatrix}, \mathbf{S} = \begin{pmatrix} 1 & 1 \\ -1 & 0 \\ 0 & 1 \end{pmatrix}.$$

   Nyní počítáme prvky matic $\mathbf{B}$ a $\mathbf{U}$ podle výše uvedených vzorců:

$$\begin{array}{llll}
\gamma_{11} = 1, & \gamma_{12} = -1/2, & \gamma_{21} = 0, & \gamma_{22} = 1, \\
\omega_{11} = 1, & \omega_{12} = 0, & \omega_{21} = 0, & \omega_{22} = 1, \\
\beta_{11} = 1/\sqrt{2}, & \beta_{12} = -1/\sqrt{6}, & \beta_{21} = 0, & \beta_{22} = \sqrt{6}/3.
\end{array}$$

Tedy

$$\mathbf{B} = \begin{pmatrix} 1/\sqrt{2} & -1/\sqrt{6} \\ 0 & \sqrt{6}/3 \end{pmatrix}, \ \mathbf{Q} = \begin{pmatrix} \sqrt{2}/2 & 1/\sqrt{6} \\ -\sqrt{2}/2 & 1/\sqrt{6} \\ 0 & 2/\sqrt{6} \end{pmatrix}.$$

(c) Ortonormalizací sloupců matice

$$\begin{pmatrix} \mathbf{U} \\ \mathbf{I} \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 0 \\ 0 & 1 \end{pmatrix}$$

získáme matici

$$\mathbf{V} = \begin{pmatrix} 1/\sqrt{2} & 0 \\ 0 & 1/\sqrt{2} \\ 1/\sqrt{2} & 0 \\ 0 & 1/\sqrt{2} \end{pmatrix} = \begin{pmatrix} \mathbf{V}_1 \\ \mathbf{V}_2 \end{pmatrix}.$$

Odtud poté dostáváme:

$$\mathbf{I} - \mathbf{V}_1\mathbf{V}_1^T = \begin{pmatrix} 1/2 & 0 \\ 0 & 1/2 \end{pmatrix} = (\mathbf{I} + \mathbf{U}\mathbf{U}^T)^{-1},$$

$$(\mathbf{I} + \mathbf{U}\mathbf{U}^T)^{-1}\mathbf{B}\mathbf{Q}^T = \begin{pmatrix} 1/6 & -1/3 & -1/6 \\ 1/6 & 1/6 & 1/3 \end{pmatrix}.$$

Celkově tedy máme:

$$\mathbf{A}^+ = \begin{pmatrix} \mathbf{I} \\ \mathbf{U}^T \end{pmatrix} (\mathbf{I} + \mathbf{U}\mathbf{U}^T)^{-1}\mathbf{B}\mathbf{Q}^T = \begin{pmatrix} 1/6 & -1/3 & -1/6 \\ 1/6 & 1/6 & 1/3 \\ 1/6 & -1/3 & -1/6 \\ 1/6 & 1/6 & 1/3 \end{pmatrix}.$$

2. Výpočet pomocí softwaru R:

```
> A=matrix(c(1,-1,0,1,0,1,1,-1,0,1,0,1),3,4)
> gramm_schmidtova_ortogonalizacni_metoda(A)
          [,1]       [,2]       [,3]
[1,] 0.1666667 -0.3333333 -0.1666667
[2,] 0.1666667  0.1666667  0.3333333
[3,] 0.1666667 -0.3333333 -0.1666667
[4,] 0.1666667  0.1666667  0.3333333
```

2.8. Cayley – Hamiltonova metoda

V následujících dvou větách uvádíme návod na praktický výpočet pseudoinverze pomocí Cayley – Hamiltonovy metody.

Věta 2.8 *Nechť* $\boldsymbol{H}$ *je symetrická matice řádu* n*. Označme* $\boldsymbol{H}_1 = \boldsymbol{H}$ *a pro* $k = 1, ..., n-1$ *označme*

$$\gamma_k = (trace(\boldsymbol{H}_k))/k,$$
$$\boldsymbol{B}_k = \boldsymbol{H}_k - \gamma_k \boldsymbol{I},$$
$$\boldsymbol{H}_{k+1} = \boldsymbol{H}\boldsymbol{B}_k.$$

Označme dále

$$\boldsymbol{M} = \begin{cases} \textit{první} \quad k \leq n \quad \textit{takové, že} \quad \boldsymbol{H}\boldsymbol{B}_k = \boldsymbol{0} \quad \textit{pokud takové} \quad k \quad \textit{existuje,} \\ n \quad \textit{jinak,} \end{cases}$$

a nechť r *je největší číslo* $k \leq \boldsymbol{M}$*, takové, že* $\gamma_k \neq 0$.
Potom

$$p(\lambda) = det\ (\boldsymbol{H} - \lambda \boldsymbol{I}) = (-1)^n\ \gamma_r\ \lambda^{n-r}\ (\boldsymbol{I} - \varphi(\lambda)),$$

kde

$$\varphi(\lambda) = (\gamma_r)^{-1}\ \left[\lambda^{r-1}\ -\ \textstyle\sum_{j=1}^{r-1} \gamma_j\ \lambda^{r-1-j}\right].$$

Důkaz: viz [1].

Věta 2.9 *Nechť* $\boldsymbol{H} = \boldsymbol{A}^T\boldsymbol{A}$*, kde* $\boldsymbol{A}$ *je libovolná matice typu* $m\ x\ n$*. Podle předchozí věty vypočteme* $\boldsymbol{H}_k, \boldsymbol{B}_k, \boldsymbol{M}$ *a* r*. Pak*

$$\boldsymbol{A}^+ = (\gamma_r)^{-1}\boldsymbol{B}_{r-1}\boldsymbol{A}^T,$$

kde r *je hodnost matice* $\boldsymbol{A}$*.*

Důkaz: viz [1].

Poznámka 2.5 V případě, že má matice $\mathbf{H}$ hodnost rovnu 1, matice $\mathbf{B} = \mathbf{I}$ řádu n.

R-file 2.8 (cayley_hamiltonova_metoda.R)

Ve statistickém softwaru R jsme vytvořili kód, který najdeme na přiloženém CD pod názvem „2_8_cayley_hamiltonova_metoda.R“. Tento kód vypočte pseudoinverzní matici $\mathbf{A}^+$ ze zadané matice $\mathbf{A}$ za pomoci Cayley – Hamiltonovy metody.

Příklad 2.8 Vypočítejte za pomoci Cayley – Hamiltonovy metody pseudoinverzní matici $\mathbf{A}^+$ k námi zadané matici $\mathbf{A} = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 0 & -1 \\ 2 & 1 & 0 \\ 1 & 1 & 1 \end{pmatrix}$.

1. Ruční výpočet:

   Označme

$$\mathbf{H}_1 = \mathbf{H} = \mathbf{A}^T\mathbf{A} = \begin{pmatrix} 7 & 4 & 1 \\ 4 & 3 & 2 \\ 1 & 2 & 3 \end{pmatrix}.$$

   Podle předchozí věty počítáme:

   $\gamma_1 = 13$,

$$\mathbf{B}_1 = \mathbf{H}_1 - 13\ \mathbf{I} \begin{pmatrix} -6 & 4 & 1 \\ 4 & -10 & 2 \\ 1 & 2 & -10 \end{pmatrix},$$

$$\mathbf{H}_2 = \mathbf{H}\mathbf{B}_1 \begin{pmatrix} -25 & -10 & 5 \\ -10 & -10 & -10 \\ 5 & -10 & -25 \end{pmatrix},$$

   $\gamma_2 = -30$,

$$\mathbf{B}_2 = \mathbf{H}_2 + 30\ \mathbf{I} \begin{pmatrix} 5 & -10 & 5 \\ -10 & 20 & -10 \\ 5 & -10 & 5 \end{pmatrix},$$

$$\mathbf{H}_3 = \mathbf{H}\mathbf{B}_2 \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}.$$

Tedy $\mathbf{M} = r = 2$, a

$$\mathbf{A}^+ = \frac{1}{\gamma_2}\mathbf{B}_1\mathbf{H}^T = \begin{pmatrix} 1/30 & 7/30 & 8/30 & 1/30 \\ 4/30 & -2/30 & 2/30 & 4/30 \\ 7/30 & -11/30 & -4/30 & 7/30 \end{pmatrix}.$$

2. Výpočet pomocí softwaru R:

```
> A=matrix(c(1,1,2,1,1,0,1,1,1,-1,0,1),4,3)
> cayley_hamiltonova_metoda(A)
           [,1]        [,2]        [,3]       [,4]
[1,] 0.03333333  0.23333333  0.26666667 0.03333333
[2,] 0.13333333 -0.06666667  0.06666667 0.13333333
[3,] 0.23333333 -0.36666667 -0.13333333 0.23333333
```

2.9. Gauss – Jordanova metoda

Nechť $\mathbf{A}$ je matice typu m x n a hodnosti r. Vždy existuje regulární matice $\mathbf{E}$ a ortogonální matice $\mathbf{P}$ takové, že

$$\mathbf{E}\mathbf{A}^T\mathbf{A}\mathbf{P} = \begin{pmatrix} \mathbf{I} & \mathbf{L} \\ \mathbf{0} & \mathbf{0} \end{pmatrix},$$

kde $\mathbf{L}$ je nějaká matice typu r x $(n-r)$ a $\mathbf{I}$ je jednotková matice řádu r. Reálně existuje mnoho takových matic $\mathbf{E}$ a $\mathbf{P}$.

Věta 2.10 *Nechť $\boldsymbol{E}$ je regulární a $\boldsymbol{P}$ ortogonální matice takové, že splňují předcházející rovnost. Pak*

$$\boldsymbol{A}^+ = \boldsymbol{P}(\boldsymbol{E}\boldsymbol{A}^T\boldsymbol{A}\boldsymbol{P})^+\boldsymbol{E}\boldsymbol{A}^T.$$

Důkaz: viz [1].

Důsledek 2.1 Nechť $\mathbf{E}$ je regulární a $\mathbf{P}$ ortogonální matice takové, že matici $\mathbf{E}\mathbf{A}^T\mathbf{A}\mathbf{P}$ lze psát ve tvaru

$$\begin{pmatrix} \mathbf{H} \\ \mathbf{0} \end{pmatrix}.$$

Potom

$$\mathbf{A}^+ = \mathbf{P}(\mathbf{H}^+\mathbf{0})\mathbf{E}\mathbf{A}^T.$$

Důkaz: viz [1].

Výpočet pseudoinverzní matice spočívá v tom, že provádíme Gaussovu eliminaci matice $\mathbf{A}^T\mathbf{A}$ a přitom tytéž operace, které provádíme s řádky této matice, provádíme také s maticí $\mathbf{A}^T$. Tím získáme matici tvaru

$$\begin{pmatrix} \mathbf{I} & \mathbf{L} & \vdots & \multirow{2}{*}{$\mathbf{E}\mathbf{A}^T$} \\ \mathbf{0} & \mathbf{0} & \vdots & \end{pmatrix}.$$

Matice

$$\mathbf{EA}^T\mathbf{AP} = \begin{pmatrix} \mathbf{I} & \mathbf{L} \\ \mathbf{0} & \mathbf{0} \end{pmatrix},$$

a tedy podle důsledku

$$\left(\mathbf{EA}^T\mathbf{AP}\right)^+ = \left(\mathbf{H}^+ \ \ \mathbf{0}\right) = ((\mathbf{I} \ \ \mathbf{L})^+ \ \ \mathbf{0}).$$

Matice $(\mathbf{I} \ \ \mathbf{L})$ je plné řádkové hodnosti a proto podle (16) je její pseudoinverze rovna

$$(\mathbf{I} \ \ \mathbf{L})^+ = (\mathbf{I} \ \ \mathbf{L})^T \left(\mathbf{I} + \mathbf{LL}^T\right)^{-1}.$$

Matici $\left(\mathbf{I} + \mathbf{LL}^T\right)^{-1}$ lze vypočítat podle již uvedené věty č. 2.7, kterou jsme využili v Gramm – Schmidtově ortogonalizační metodě.

R-file 2.9 (gauss_jordanova_metoda.R)

Ve statistickém softwaru R jsme vytvořili kód, který najdeme na přiloženém CD pod názvem „2_9_gauss_jordanova_metoda.R“. Tento kód vypočte pseudoinverzní matici $\mathbf{A}^+$ ze zadané matice $\mathbf{A}$ za pomoci Gauss – Jordanovy metody.

Příklad 2.9 Vypočítejte za pomoci Gauss – Jordanovy metody pseudoinverzní matici $\mathbf{A}^+$ k námi zadané matici $\mathbf{A} = \begin{pmatrix} 1 & -1 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & -1 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & -1 & 0 & 1 \\ 1 & 0 & 1 & 1 \end{pmatrix}$.

1. Ruční výpočet:

   Nechť $\mathbf{A}$ je námi zadaná matice. Pak

$$\left(\mathbf{A}^T\mathbf{A} \quad \mathbf{A}^T\right) = \left(\begin{array}{rrrr|rrrrrr} 6 & -3 & 3 & 6 & 1 & 1 & 1 & 1 & 1 & 1 \\ -3 & 3 & 0 & -3 & -1 & 0 & -1 & 0 & -1 & 0 \\ 3 & 0 & 3 & 3 & 0 & 1 & 0 & 1 & 0 & 1 \\ 6 & -3 & 3 & 6 & 1 & 1 & 1 & 1 & 1 & 1 \end{array}\right).$$

Po provedení Gaussovy eliminace dostaneme:

$$\left(\begin{array}{cccc|cccccc} 1 & 0 & 1 & 1 & 0 & 1/3 & 0 & 1/3 & 0 & 1/3 \\ 0 & 1 & 1 & 0 & -1/3 & 1/3 & -1/3 & 1/3 & -1/3 & 1/3 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{array}\right) =$$

$$= \begin{pmatrix} \mathbf{E}\mathbf{A}^T\mathbf{A}\mathbf{P} & \mathbf{E}\mathbf{A}^T \end{pmatrix}.$$

Permutační matice $\mathbf{P}$ je tedy rovna jednotkové matici řádu 4. Podle důsledku je tedy matice

$$\mathbf{E}\mathbf{A}^T\mathbf{A}\mathbf{P} = \begin{pmatrix} \mathbf{H} \\ \mathbf{0} \end{pmatrix} = \left(\begin{array}{cc|cc} 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \\ \hline \multicolumn{4}{c}{\mathbf{0}} \end{array}\right).$$

Dále tedy

$$\mathbf{L} = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}, \quad \mathbf{I} + \mathbf{L}\mathbf{L}^T = \begin{pmatrix} 3 & 1 \\ 1 & 2 \end{pmatrix}.$$

Proto následně

$$(\mathbf{I} + \mathbf{L}\mathbf{L}^T)^{-1} = \begin{pmatrix} 2/5 & -1/5 \\ -1/5 & 3/5 \end{pmatrix}.$$

$$\mathbf{H}^+ = (\mathbf{I} \quad \mathbf{L})^T \, (\mathbf{I}+\mathbf{L}\mathbf{L}^T)^{-1} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 2/5 & -1/5 \\ -1/5 & 3/5 \end{pmatrix} = \begin{pmatrix} 2/5 & -1/5 \\ -1/5 & 3/5 \\ 1/5 & 2/5 \\ 2/5 & -1/5 \end{pmatrix}.$$

A konečně, podle důsledku, je $\mathbf{A}^+ = \mathbf{P}\ (\mathbf{H}^+\ \ \mathbf{0})\ \mathbf{E}\ \mathbf{A}^T$, tedy

$$\mathbf{A}^+ = \begin{pmatrix} 1/15 & 1/15 & 1/15 & 1/15 & 1/15 & 1/15 \\ -1/5 & 2/15 & -1/5 & 2/15 & -1/5 & 2/15 \\ -2/15 & 1/5 & -2/15 & 1/5 & -2/15 & 1/5 \\ 1/15 & 1/15 & 1/15 & 1/15 & 1/15 & 1/15 \end{pmatrix}.$$

2. Výpočet pomocí softwaru R:

```
> A=matrix(c(1,1,1,1,1,1,-1,0,-1,0,-1,0,0,1,0,1,0,1,1,1,1,
1,1,1),6,4)
> gauss_jordanova_metoda(A)
            [,1]       [,2]        [,3]       [,4]
[1,]  0.06666667 0.06666667  0.06666667 0.06666667
[2,] -0.20000000 0.13333333 -0.20000000 0.13333333
[3,] -0.13333333 0.20000000 -0.13333333 0.20000000
[4,]  0.06666667 0.06666667  0.06666667 0.06666667
            [,5]       [,6]
[1,]  0.06666667 0.06666667
[2,] -0.20000000 0.13333333
[3,] -0.13333333 0.20000000
[4,]  0.06666667 0.06666667
```

3. Řešení soustav lineárních rovnic

Samotné řešení soustav lineárních rovnic je jedním z nejstarších problémů matematiky. Již před asi 4000 lety lidé z Babylonu věděli, jak vyřešit jednoduchý systém dvou lineárních rovnic se dvěma neznámými. Kolem roku 200 př. n. l. Číňané v knize „Nine Chapters of the Mathematical Art“ ukázali řešení systémů tří lineárních rovnic se třemi neznámými. Na přelomu 19. století Gauss představil postup, který řešil soustavy lineárních rovnic větších rozměrů. V té době však neexistoval termín „matice“, i proto později vzniklo celé odvětví matematiky – teorie matic. Ve 20. století byl vznesen požadavek na to, aby počítače mohly rychle a přesně vyřešit velké systémy rovnic jakýchkoliv typů. Proto se s rozvojem technologií mnoho matematiků snažilo tento problém vyřešit. Technologie se neustále vylepšují, a proto dnes umíme řešit i pseudoinverze matic. Nicméně historie lineární algebry nám poskytuje pevný základ, který se nemění. A ačkoliv všichni aktualizují své učební materiály, základy zůstavají vždy stejné.

V této kapitole si ukážeme všechny tři možnosti, které mohou při řešení soustav lineárních rovnic nastat. Buď bude mít soustava jedno řešení, nekonečně mnoho řešení, anebo žádné řešení.

Ve všech případech však můžeme nalézt „řešení“ za pomoci pseudoinverzní matice. V případě, kdy má soustava jedno řešení, to bude právě toto řešení. Pokud má soustava nekonečně mnoho řešení, pseudoinverze nalezne takové řešení, které je v normě nejmenší. Jestliže nastane poslední možná situace, a to, že soustava nemá řešení, pseudoinverze nalezne nejlepší přibližné řešení.

Všechny tyto výše zmíněné případy si podrobně rozebereme v následujících podkapitolách.

Zde je zapotřebí připomenout, že počítáme soustavu m lineárních rovnic o n neznámých (2), kterou lze ekvivalentně přepsat do maticového tvaru (4), nebo zkráceně

$$\mathbf{Ax} = \mathbf{b},$$

kde $\mathbf{A}$ je maticí soustavy lineárních rovnic, $\mathbf{x} = (x_1, x_2, \ldots, x_n)^T$ je vektor řešení soustavy a $\mathbf{b} = b_1, b_2, \ldots, b_n)^T$ je vektor pravé strany soustavy rovnic.

3.1. Řešení soustav lineárních rovnic v případě, je-li množina všech řešení tohoto systému jednoprvková

Tato situace nastane v případě, kdy $r(\mathbf{A}) = r(\mathbf{A}_r) = n$, kde n značí počet neznámých. Řešení takové soustavy rovnic je jediné – máme tedy právě jedno řešení dané soustavy rovnic.

Jestliže navíc je matice $\mathbf{A}$ čtvercová, lze řešení nalézt výpočtem $\mathbf{x}^* = \mathbf{A}^{-1}\mathbf{b}$. V Rku lze totéž jednoduše vypočítat příkazem „solve“. Tento typ soustavy lineárních rovnic je nejjednodušší, jaký vůbec můžeme řešit.

Příklad 3.1

Vyřešte za pomoci výpočetního softwaru R danou soustavu lineárních rovnic:

$$\begin{aligned} x_1 + 2x_2 &= 3 \\ -x_1 - x_2 &= 1, \end{aligned}$$

$$\text{kde } r(\mathbf{A}) = 2 \text{ a } r(\mathbf{A}_r) = 2.$$

```
> A = matrix(c(1, -1, 2, -1), 2, 2)
> b = matrix(c(3, 1), 2, 1)
> x = solve(A, b)
> x
     [,1]
[1,]   -5
[2,]    4
```

Pokud ale matice $\mathbf{A}$ čtvercová není, je zapotřebí rozšířenou matici soustavy $\mathbf{A}_r$ řádkovými úpravami (například za pomoci Gaussovy eliminační metody) převést do tvaru, kdy se pod hlavní diagonálou nachází pouze nuly. Nulové řádky poté vyškrtnout a $\mathbf{A}_r$ opět rozdělit na matici $\mathbf{A}_2$ a vektor řešení $\mathbf{b}_2$. Tímto se matice $\mathbf{A}$ a vektor řešení $\mathbf{b}$ dostanou do tvaru, který jsme výše zmínili.

Takový postup je však zdlouhavý a pracný. Daleko efektivnější je použít, namísto klasické inverze, pseudoinverzi matice $\mathbf{A}$.

V případě, je-li množina všech řešení soustavy lineárních rovnic jednoprvková, nám výpočet $\mathbf{x}^* = \mathbf{A}^+\mathbf{b}$, kde $\mathbf{A}^+$ je pseudoinverzní matice matice $\mathbf{A}$, nalezne přesné řešení soustavy rovnic, stejně jako výpočet užitím klasické inverze, a to $\mathbf{x}^* = \mathbf{A}^{-1}\mathbf{b}$.

Příklad 3.2

Vyřešte za pomoci výpočetního softwaru R danou soustavu lineárních rovnic:

$$\begin{aligned} x_1 \; + 2x_2 &= 3 \\ -x_1 - \; x_2 &= 1 \\ 2x_1 + 4x_2 &= 6, \end{aligned}$$

kde $r(\mathbf{A}) = 2$ a $r(\mathbf{A}_r) = 2$.

```
> A = matrix(c(1, -1, 2, 2, -1, 4), 3, 2)
> b = matrix(c(3, 1, 6), 3, 1)
> Apinv = pinv_skeletni_rozklad(A)
> x = Apinv %*% b
> x
     [,1]
[1,]   -5
[2,]    4
```

V případě užití příkazu „solve“ v této situaci Rko vypíše chybovou hlášku, že vstupní matice $\mathbf{A}$ musí být čtvercová.

```
> A = matrix(c(1, -1, 2, 2, -1, 4), 3, 2)
> b = matrix(c(3, 1, 6), 3, 1)
> x = solve(A, b)
> Error in solve.default(A, b) : 'a' (3 x 2) must be square
```

3.2. Řešení soustav lineárních rovnic v případě, je-li množina všech řešení tohoto systému nekonečná

Tato situace nastane v případě, kdy $r(\mathbf{A}) = r(\mathbf{A}_r) = k < n$, kde n značí počet neznámých. V případě, kdy má námi zadaná soustava rovnic nekonečně mnoho řešení, hledáme takové řešení soustavy rovnic, které je v normě nejmenší. Jinými slovy hledáme takové $\mathbf{x}^* \in \mathbb{R}^n$, že pro něj platí:

$$||\mathbf{x}^*|| \leq ||\bar{\mathbf{x}}|| \quad \forall \bar{\mathbf{x}} \in \mathbb{R}^n : \mathbf{A}\bar{\mathbf{x}} = \mathbf{b}.$$

Kde $\mathbf{x}^*$ spočítáme jako:

$$\mathbf{x}^* = \mathbf{A}^+\mathbf{b}, \text{ kde } \mathbf{A}^+ \text{ je pseudoinverzní matice matice } \mathbf{A}.$$

Příklad 3.3

Vyřešte za pomoci výpočetního softwaru R danou soustavu lineárních rovnic:

$$\begin{array}{rcrcl} x_1 & + & x_2 & = & 4 \\ 2x_1 & + & 2x_2 & = & 8, \end{array}$$

$$\text{kde } h(\mathbf{A}) = 1 \text{ a } h(\mathbf{A}_r) = 1.$$

```
> A = matrix(c(1, 2, 1, 2), 2, 2)
> b = matrix(c(4, 8), 2, 1)
> Apinv = pinv_skeletni_rozklad(A)
> x = Apinv %*% b
> x
     [,1]
[1,]    2
[2,]    2
```

3.3. Řešení soustav lineárních rovnic v případě, je-li množina všech řešení tohoto systému prázdná

Tato situace nastane v případě, kdy $r(\mathbf{A}) \neq r(\mathbf{A}_r)$. Jestliže námi zadaná soustava rovnic nemá žádné řešení, hledáme nejlepší přibližné řešení $\mathbf{x}^*$. Tedy takové $\mathbf{x}^* \in \mathbb{R}^n$, že platí:

$$||\mathbf{A}\mathbf{x}^* - \mathbf{b}|| \leq ||\mathbf{A}\mathbf{x} - \mathbf{b}|| \quad \forall \mathbf{x} \in \mathbb{R}^n.$$

Kde $\mathbf{x}^*$ spočítáme jako:

$$\mathbf{x}^* = \mathbf{A}^+\mathbf{b}, \text{ kde } \mathbf{A}^+ \text{ je pseudoinverze matice } \mathbf{A}.$$

Příklad 3.4

Vyřešte za pomoci výpočetního softwaru R danou soustavu lineárních rovnic:

$$\begin{array}{rcrcl} x_1 & + & x_2 & = & 4 \\ 2x_1 & + & 2x_2 & = & 1, \end{array}$$

$$\text{kde } h(\mathbf{A}) = 1 \text{ a } h(\mathbf{A}_r) = 2.$$

```
> A = matrix(c(1, 2, 1, 2), 2, 2)
> b = matrix(c(4, 1), 2, 1)
> Apinv = pinv_skeletni_rozklad(A)
> x = Apinv %*% b
> x
     [,1]
[1,]  0.6
[2,]  0.6
```

3.4. Ucelený kód pro řešení soustav lineárních rovnic

Jako poslední kód v Rku jsme vytvořili tzv. „ucelený kód“, který obsahuje všechny kódy pro výpočet pseudoinverzních matic. Můžeme tedy vyřešit soustavu lineárních rovnic jakýmkoliv z devíti postupů, které jsme si uvedli ve druhé kapitole.

R-file 3.1 (uceleny_kod.R)

Ve statistickém softwaru R jsme vytvořili kód, který najdeme na přiloženém CD pod názvem „3_1_uceleny_kod.R“. Tento kód vyřeší soustavu lineárních rovnic jakýmkoliv z námi uvedených způsobů. V prvé řadě musíme zadat vstupní matici **A** a vektor řešení **b**. Poté vybereme metodu, kterou budeme chtít použít podle čísla, které následně použijeme dle příkladu níže:

1. Skeletní rozklad
2. Grevillův algoritmus
3. Příkaz PINV
4. Zhukovskii - Liptserova metoda
5. Biortogonalizační metoda
6. Metoda gradientních projekcí
7. Gramm - Schmidtova ortogonalizační metoda
8. Cayley - Hamiltonova metoda
9. Gauss - Jordanova metoda

Příklad 3.5

Vyřešte za pomoci výpočetního softwaru R soustavu lineárních rovnic z příkladu 3.4 za pomoci příkazu PINV. (*za pomoci skeletniho rozkladu: metoda=1, atd.)

```
> A = matrix(c(1, -1, 2, 2, -1, 4), 3, 2)
> b = matrix(c(3, 1, 6), 3, 1)
> x = uceleny_kod(A, b, metoda=3)
> x
     [,1]
[1,]   -5
[2,]    4
```

4. Stabilita a rychlost kódů pro výpočet pseudoinverzích matic

Na stabilitu a rychlost výpočtu se v praxi klade největší důraz. A proto čtvrtou kapitolu věnujeme právě stabilitě a rychlosti výpočtů pseudoinverzních matic.

Následující výpočty byly prováděny na notebooku HP ProBook 4540s s procesorem Intel(R) Core(TM) i5-3230M CPU @2.60GHz, 4.00GB RAM.

Označme nyní $\mathbf{A}_i$ pseudoinverzní matici matice $\mathbf{A}$ získanou i-tou metodou. Pořadí metod je stejné, jako pořadí podkapitol, tj.:

1. Skeletní rozklad
2. Grevillův algoritmus
3. Příkaz PINV
4. Zhukovskii - Liptserova metoda
5. Biortogonalizační metoda
6. Metoda gradientních projekcí
7. Gramm - Schmidtova ortogonalizační metoda
8. Cayley - Hamiltonova metoda
9. Gauss - Jordanova metoda

Při ověřování Moore – Penroseových axiomů pro pseudoinverzi dostáváme, že tato pseudoinverze je stanovena s určitou chybou. Nyní budeme zjišťovat, u které metody jsou tyto odchylky v normě nejmenší. Pro každou matici $\mathbf{A}_i$ vypočítáme vektor

$$\mathbf{dh}_i = (\mathbf{dh}_i^1, \mathbf{dh}_i^2, \mathbf{dh}_i^3, \mathbf{dh}_i^4)^T,$$

jehož složky jsou rezidua v Moore – Penroseových axiomech, tedy

i-tá metoda	vektor $\mathbf{dh}_i$
$\|\mathbf{A}\mathbf{A}_i\mathbf{A} - \mathbf{A}\|$	$\mathbf{dh}_i^1$
$\|\mathbf{A}_i\mathbf{A}\mathbf{A}_i - \mathbf{A}_i\|$	$\mathbf{dh}_i^2$
$\|\mathbf{A}_i^T\mathbf{A}^T - \mathbf{A}\mathbf{A}_i\|$	$\mathbf{dh}_i^3$
$\|\mathbf{A}^T\mathbf{A}_i^T - \mathbf{A}_i\mathbf{A}\|$	$\mathbf{dh}_i^4$

a normu $\|\mathbf{dh}_i\| = \sqrt{\mathbf{dh}_i \mathbf{dh}_i^T}$.

V tabulkách níže jsme uvažovali matice uvedených rozměrů a hodností. Prvky matic tvořili vždy čísla od 0 do 1. Pokud bychom zvolili jiná čísla, například z normálního rozdělení, doba výpočtů i odchylky se zvýšily, a to z důvodu podmíněnosti úloh. Tento fakt však není předmětem našeho zkoumání.

V tabulkách 1a, 2a jsou $\|\mathbf{dh}_i\|$ pro různé typy matic. Nejmenší z hodnot $\|\mathbf{dh}_i\|$ ukazuje metodu, při které je chyba nejmenší, naopak největší hodnota ukazuje na nejméně stabilní metodu. Z výsledků je zřejmé, že z hlediska přesnosti je nejlepší příkaz „PINV“, který využívá singulární rozklad matice.

Tabulky 1b, 2b nám shrnují potřebné doby výpočtů ve vteřinách. Jak můžeme vidět, nejrychlejší je opět příkaz „PINV“. U matic malých rozměrů je v dnešní době doba výpočtu zanedbatelná. Čím jsou rozměry matice větší, tím je výpočet náročnější a samozřejmě také déle trvá.

Výpočet pseudoinverze za pomoci Cayley–Hamiltonovy metody je velmi nepřesný. U matic větších rozměrů dokonce software R tyto odchylky ani nedokáže spočítat, jelikož přesahují jeho výpočetní možnosti. Proto nám Rko vrátí pouze hodnoty neznámé (NA). Tento výpočet je vhodný proto pouze pro matice malých rozměrů.

Přesnost je ve výpočetním softwaru R vyjádřena číslem $eps = 2.220446e-16$. Z důvodu šíření chyb v našich kódech jsme ji však cíleně u některých kódů snížili.

Níže uvedené tabulky je nutné brát s rezervou. Doby výpočtů a přesnost úzce souvisí s tím, jak jsou napsány výpočetní kódy. Vzhledem k rozsahu této práce jsme se snažili tyto kódy vytvořit tak, aby fungovaly, byť s menší přesností. Je proto pravděpodobné, že námi vytvořené kódy se dají zrychlit, a zároveň lze zvýšit jejich přesnost.

V praxi se při řešení soustav lineárních rovnic nebere v potaz, která ze tří možných situací nastane. Ukázali jsme si, jaká řešení v těchto třech případech dostaneme. Jestliže bychom však chtěli zjistit, která ze situací nastala, je to možné přes hodnost matice. Postupovali bychom analogicky jako ve třetí kapitole.

TABULKA 1a

		Metoda		
m x n	**r**	**1**	**2**	**3**
5 x 5	**5**	8.212153e-14	1.129851e-13	1.237583e-14
5 x 10	**5**	1.419663e-13	2.881142e-14	1.964872e-14
10 x 10	**5**	6.449862e-14	1.647483e-14	1.077076e-14
15 x 10	**5**	6.465422e-14	8.865648e-14	1.029248e-14
20 x 10	**5**	1.392065e-13	6.292297e-14	1.047911e-14
25 x 10	**5**	2.022504e-13	5.201062e-14	1.377924e-14
průměrné odch.		**1.157829e-13**	**6.031021e-14**	**1.289102e-14**

		Metoda		
m x n	**r**	**4**	**5**	**6**
5 x 5	**5**	3.591247e-14	5.453816e-15	1.028708e-13
5 x 10	**5**	2.634672e-14	6.671929e-15	8.136771e-14
10 x 10	**5**	2.158793e-14	7.329576e-15	6.005922e-14
15 x 10	**5**	7.879034e-14	8.103031e-15	6.282614e-14
20 x 10	**5**	9.019455e-14	6.370539e-15	7.529085e-14
25 x 10	**5**	2.679672e-13	6.216882e-15	1.184639e-13
průměrné odch.		**8.679986e-14**	**6.690962e-15**	**8.347976e-14**

		Metoda		
m x n	**r**	**7**	**8**	**9**
5 x 5	**5**	6.384362e-14	1.04946	5.120228e-14
5 x 10	**5**	3.086404e-14	668.78481	4.198224e-14
10 x 10	**5**	1.544044e-14	124.86332	3.643911e-14
15 x 10	**5**	2.433243e-14	58.98749	4.423919e-14
20 x 10	**5**	3.667496e-14	66.69287	1.064499e-13
25 x 10	**5**	6.995323e-14	69.39335	1.677721e-13
průměrné odch.		**4.018479e-14**	**164.96191**	**7.46808e-14**

Tabulka 1a: Průměrné $\|\mathbf{dh}_i\|$ pro různé typy matic

TABULKA 1b

		Metoda		
m x n	**r**	**1**	**2**	**3**
5 x 5	**5**	0.02	0.02	0.01
5 x 10	**5**	0.02	0.03	0.02
10 x 10	**5**	0.04	0.02	0.03
15 x 10	**5**	0.07	0.02	0.03
20 x 10	**5**	0.08	0.03	0.03
25 x 10	**5**	0.08	0.04	0.03
průměrné doby		**0.0517**	**0.0267**	**0.0250**

		Metoda		
m x n	**r**	**4**	**5**	**6**
5 x 5	**5**	0.02	0.01	0.03
5 x 10	**5**	0.02	0.01	0.03
10 x 10	**5**	0.02	0.02	0.03
15 x 10	**5**	0.02	0.03	0.02
20 x 10	**5**	0.03	0.05	0.05
25 x 10	**5**	0.03	0.06	0.11
průměrné doby		**0.0233**	**0.0300**	**0.0450**

		Metoda		
m x n	**r**	**7**	**8**	**9**
5 x 5	**5**	0.01	0.02	0.01
5 x 10	**5**	0.03	0.02	0.02
10 x 10	**5**	0.04	0.03	0.03
15 x 10	**5**	0.02	0.03	0.02
20 x 10	**5**	0.03	0.02	0.02
25 x 10	**5**	0.05	0.03	0.04
průměrné doby		**0.0300**	**0.0250**	**0.0233**

Tabulka 1b: Průměrné doby výpočtů ve vteřinách pro různé typy matic

TABULKA 2a

		Metoda		
m x n	**r**	**1**	**2**	**3**
50 x 50	**50**	2.299822e-10	6.381746e-10	7.875812e-13
50 x 100	**50**	6.620814e-10	2.548953e-10	6.414034e-13
75 x 100	**50**	7.819199e-10	3.095571e-10	6.724949e-13
100 x 100	**50**	4.087051e-10	1.479305e-10	5.615352e-13
125 x 100	**50**	6.087025e-10	5.898947e-10	4.940294e-13
150 x 100	**50**	7.746208e-10	2.252173e-10	6.268997e-13
175 x 100	**50**	6.723722e-10	1.939012e-09	7.947023e-13
200 x 100	**50**	8.182534e-10	1.938129e-10	5.981222e-13
250 x 100	**50**	7.869336e-10	4.151867e-10	6.981554e-13
průměrné odch.		**6.381746e-10**	**4.986542e-10**	**6.527693e-13**

		Metoda		
m x n	**r**	**4**	**5**	**6**
50 x 50	**50**	3.397931e-11	1.898712e-11	3.041169e-10
50 x 100	**50**	3.786256e-11	6.427621e-11	2.460951e-10
75 x 100	**50**	1.033612e-10	1.518707e-11	3.318881e-10
100 x 100	**50**	7.837038e-11	3.514121e-11	2.062702e-10
125 x 100	**50**	6.786811e-11	5.349159e-11	1.735763e-10
150 x 100	**50**	1.222738e-10	6.130897e-11	1.867163e-10
175 x 100	**50**	2.140342e-10	5.776159e-11	4.352949e-10
200 x 100	**50**	6.666576e-11	2.406394e-11	5.091226e-10
250 x 100	**50**	1.125697e-10	5.480025e-11	3.502216e-10
průměrné odch.		**9.299831e-11**	**4.277977e-11**	**3.048113e-10**

		Metoda		
m x n	**r**	**7**	**8**	**9**
50 x 50	**50**	2.187203e-10	4979.78852	7.032385e-10
50 x 100	**50**	1.232812e-10	5968.78771	4.99569e-10
75 x 100	**50**	1.902359e-10	9961.94614	2.696588e-10
100 x 100	**50**	9.267146e-11	NA	3.131771e-10
125 x 100	**50**	9.203614e-11	NA	3.611364e-10
150 x 100	**50**	1.063129e-10	NA	2.573381e-10
175 x 100	**50**	2.738702e-10	NA	4.291846e-10
200 x 100	**50**	3.174906e-10	NA	6.888951e-10
250 x 100	**50**	2.232844e-10	NA	6.538341e-10
průměrné odch.		**1.819892e-10**	**NA**	**4.640035e-10**

Tabulka 2a: Průměrné $\|\mathbf{dh}_i\|$ pro různé typy matic

TABULKA 2b

		Metoda		
m x n	**r**	**1**	**2**	**3**
50 x 50	**50**	0.08	0.11	0.11
50 x 100	**50**	0.16	0.19	0.17
75 x 100	**50**	1.18	0.28	0.25
100 x 100	**50**	1.36	0.34	0.33
125 x 100	**50**	1.38	0.36	0.34
150 x 100	**50**	1.45	0.39	0.35
175 x 100	**50**	1.52	0.36	0.33
200 x 100	**50**	1.60	0.42	0.38
250 x 100	**50**	1.69	0.45	0.35
průměrné doby		**1.16**	**0.32**	**0.29**

		Metoda		
m x n	**r**	**4**	**5**	**6**
50 x 50	**50**	0.11	0.41	0.20
50 x 100	**50**	0.31	0.51	0.27
75 x 100	**50**	0.41	1.36	0.45
100 x 100	**50**	0.50	2.75	0.64
125 x 100	**50**	0.69	2.83	0.78
150 x 100	**50**	1.00	2.81	0.90
175 x 100	**50**	1.48	2.90	1.11
200 x 100	**50**	2.08	2.84	1.37
250 x 100	**50**	4.11	2.92	1.93
průměrné doby		**1.19**	**2.15**	**0.85**

		Metoda		
m x n	**r**	**7**	**8**	**9**
50 x 50	**50**	0.69	0.09	0.10
50 x 100	**50**	3.09	0.23	0.22
75 x 100	**50**	3.36	0.32	0.33
100 x 100	**50**	3.53	0.96	0.36
125 x 100	**50**	3.71	1.26	0.39
150 x 100	**50**	3.76	1.53	0.41
175 x 100	**50**	4.01	1.83	0.43
200 x 100	**50**	4.19	2.10	0.41
250 x 100	**50**	4.46	2.46	0.49
průměrné doby		**3.42**	**1.20**	**0.35**

Tabulka 2b: Průměrné doby výpočtů ve vteřinách pro různé typy matic

Závěr

Cílem této bakalářské práce je poskytnutí ucelených informací týkajících se problematiky řešení soustav lineárních rovnic s obdélníkovými maticemi. K této problematice jsme museli přidat zejména řešení pseudoinverzních matic, kterým je věnována převážná část této práce.

Dále jsme vytvořili kódy v softwaru R, kterými lze vypočítat pseudoinverzní matice, či vyřešit soustavy lineárních rovnic. Díky tomu jsme se naučili pracovat se softwarem R, čemuž klademe největší přínos této práce. Práce s tímto softwarem byla pro nás, hlavně v začátcích, složitá, nicméně postupem času se stávala denní rutinou.

V neposlední řadě jsme se naučili ovládat program TeX, což je sázecí systém pro tvorbu knih. Obzvláště pak takových, které obsahují spoustu matematických symbolů a vzorců.

Závěrem pouze dodáváme, že celá tvorba této práce nás donutila nahlédnout do takových zákoutí matematiky, do kterých bychom se v běžných hodinách bakalářského studia nedostali.

Věřím, že tato práce bude přínosem nejen pro nás, ale i pro všechny další čtenáře, kteří se budou chtít něco o této problematice dozvědět.

Literatura

[1] ALBERT, A.: *Regression and the Moore-Penrose Pseudoinverse*. Academic Press, New York and London, 1972.

[2] BEN-ISRAEL, Adi. a T. N. E. GREVILLE: *Generalized inverses: theory and applications*. Springer, New York, 2003.

[3] BICAN, L.: *Lineární algebra a geometrie*. Academia, Praha, 2009.

[4] DAVIS, P. J.: *Circulant matrices*. John Wiley and Sons, New York, 1979.

[5] GANTMACHER, F. R.: *The theory of Matrices*. Chelsea publishing company, New York, 1959.

[6] GOLUB, H. a Charles F. VAN LOAN: *Matrix computations*. Johns Hopkins University Press, Baltimore, 1996.

[7] KOMÁREK, A.: *Hrátky s R*. [online]. 2009, [cit. 2017-04-04]. Dostupný z: `https://www.karlin.mff.cuni.cz/~kulich/vyuka/Rdoc/arnost_introR.pdf`

[8] MACHALOVÁ, J.: *Výpočty pseudoinverzních matic*. Preprint Department MAaAM, Olomouc, 1998.

[9] POKORNÁ, O.: *Pseudoinverzní matice*. Státní pedagogické nakladatelství, Praha, 1978.