UNIVERZITA KOMENSKÉHO V BRATISLAVE
FAKULTA MATEMATIKY, FYZIKY A INFORMATIKY

METÓDY SELEKCIE PREMENNÝCH V LINEÁRNYCH REGRESNÝCH MODELOCH

DIPLOMOVÁ PRÁCA

2018 Mgr. Michal DANIŠKA, PhD.

UNIVERZITA KOMENSKÉHO V BRATISLAVE
FAKULTA MATEMATIKY, FYZIKY A INFORMATIKY

METÓDY SELEKCIE PREMENNÝCH V LINEÁRNYCH REGRESNÝCH MODELOCH

DIPLOMOVÁ PRÁCA

Študijný program:	Ekonomicko-finančná matematika a modelovanie
Študijný odbor:	9.1.9. Aplikovaná matematika
Školiace pracovisko:	Katedra aplikovanej matematiky a štatistiky
Vedúci práce:	doc. Mgr. Radoslav Harman, PhD.

Bratislava 2018 **Mgr. Michal DANIŠKA, PhD.**

52336681

Univerzita Komenského v Bratislave
Fakulta matematiky, fyziky a informatiky

ZADANIE ZÁVEREČNEJ PRÁCE

Meno a priezvisko študenta: Mgr. Michal Daniška, PhD.
Študijný program: ekonomicko-finančná matematika a modelovanie (Jednoodborové štúdium, magisterský II. st., denná forma)
Študijný odbor: aplikovaná matematika
Typ záverečnej práce: diplomová
Jazyk záverečnej práce: slovenský
Sekundárny jazyk: anglický

Názov: Metódy selekcie premenných v lineárnych regresných modeloch
Algorithms for variable selection in linear regression models

Anotácia: Selekcia premenných je proces výberu podmnožiny dostupných vstupných premenných použitej pri konštrukcii štatistického modelu. Hlavným cieľom selekcie premenných je potlačiť vplyv šumu na hodnotu odhadov parametrov určujúcich výsledný štatistický model, prípadne zlepšiť interpretovateľnosť modelu a numerickú stabilitu výpočtu. V prípade lineárneho regresného modelu sa pri veľkom počte premenných štandardne používa len niekoľko heuristických selekčných algoritmov, napríklad dopredná selekcia či LASSO. Tieto sú dostatočne rýchle pre praktickú aplikáciu, avšak ich výsledok je z hľadiska chyby predikcie modelu vo všeobecnosti suboptimálny.

Cieľ: Cieľom diplomovej práce je porovnať štandardné metódy selekcie premenných v lineárnych regresných modeloch a pokúsiť sa vyhodnotiť odhad predikčnej chyby submodelov zvolených analyzovanými selekčnými algoritmami oproti globálnemu optimu.

Vedúci: doc. Mgr. Radoslav Harman, PhD.
Katedra: FMFI.KAMŠ - Katedra aplikovanej matematiky a štatistiky
Vedúci katedry: prof. RNDr. Daniel Ševčovič, DrSc.

Dátum zadania: 25.01.2017

Dátum schválenia: 27.01.2017 prof. RNDr. Daniel Ševčovič, DrSc.
garant študijného programu

.....
študent

.....
vedúci práce

Regression is for prediction, not explanation.

WERNER STÜTZLE

Poďakovanie

Touto cestou by som sa rád poďakoval vedúcemu tejto diplomovej práce, doc. Mgr. Radoslavovi Harmanovi, PhD., za jeho vzácny čas, trpezlivosť a ochotu vždy pomôcť. Ďakujem za cenné návrhy, rady a pripomienky, ktoré mi pomohli nielen pri tvorbe tejto práce. Ďakujem tiež Dr. Eng. Jánovi Dolinskému a Mgr. Róbertovi Tóthovi z firmy Tangent works za predstavenie problému, ktorý sa stal základom tejto práce; za cenné diskusie v procese riešenia zadanej úlohy a za poskytnutie dát potrebných pre numerickú analýzu. Ďakujem Centru pre výskum kvantovej informácie Fyzikálneho ústavu SAV za poskytnutie výpočtového času na výkonnom počítači, čo pomohlo výrazne zvýšiť vedeckú hodnotu numerických výsledkov prezentovaných v tejto práci. Ďakujem nemenovaným vyučujúcim z FMFI UK za to, že mi pomohli objaviť mnoho netušených krás, ktoré ukrýva matematika, ako aj za nemálo podnetných myšlienok, ktoré ma výrazne posunuli vpred. Ďakujem Matejovi, Jakubovi, Danielovi a Petrovi za podnetné nápady a záujem o túto prácu, modernú štatistiku, optimalizačné problémy a umelú inteligenciu. A najviac ďakujem Bohu.

Abstrakt

DANIŠKA, Michal: Metódy selekcie premenných v lineárnych regresných modeloch [Diplomová práca], Univerzita Komenského v Bratislave, Fakulta matematiky, fyziky a informatiky, Katedra aplikovanej matematiky a štatistiky; školiteľ: doc. Mgr. Radoslav Harman, PhD., Bratislava, 2018, 97 s.

V tejto práci analyzujeme vlastnosti rôznych metód selekcie premenných v lineárnom regresnom modeli. Hlavným porovnávacím kritériom je schopnosť algoritmu zvoliť model s čo najnižším odhadom predikčnej chyby na danej sade reálnych dát. Štandardné selekčné metódy dopĺňame niekoľkými vlastnými návrhmi selekčných algoritmov. V súlade s očakávaniami boli celkovo najlepšie výsledky získané pomocou relaxovaného LASSO. V kategórii prehľadávacích algoritmov bol najnižší odhad predikčnej chyby dosiahnutý pomocou validačného VS.KL algoritmu, ktorý je vlastným návrhom. Z hľadiska predikcie nebol pozorovaný významný rozdiel medzi best subset metódami a doprednou selekciou. Najlepšie výsledky pri riešení best subset úlohy dosiahol algoritmus VS.KL, ktorý je tiež vlastným návrhom. Detailne tiež popisujeme formalizmus sweepovania, ktorý je základom praktickej realizácie niekoľkých selekčných metód. Táto časť práce je sformulovaná výrazne odlišne v porovnaní s dostupnou literatúrou a prezentuje aj niekoľko dosiaľ nepublikovaných výsledkov.

Kľúčové slová: Selekcia premenných, Lineárna regresia, Bias-variance tradeoff, Best subset selekcia, Dopredná a spätná selekcia, Zmiešané celočíselné programovanie, Sweepovanie, Principle Pivot Transform, Výmenný KL algoritmus, LASSO, Relaxované LASSO.

Abstract

DANIŠKA, Michal: Algorithms for variable selection in linear regression models. [Diploma Thesis], Comenius University in Bratislava, Faculty of Mathematics, Physics and Informatics, Department of Applied Mathematics and Statistics; Supervisor: doc. Mgr. Radoslav Harman, PhD., Bratislava, 2018, 97 p.

In this thesis we analyse properties of various methods for variable selection in linear regression models. The main criterion in the comparative analysis is the ability of the algorithm to choose the model with the expected prediction error being as low as possible. In addition to the well-known selection algorithms, we propose three new ones. As expected, the best overall results were obtained via relaxed LASSO. If restricted to the non-shrinkage methods, the minimal expected prediction error was achieved in case of the newly proposed validation VS.KL algorithm. We did not observe any significant difference between the best subset methods and the forward selection regarding quality of the predictions. However, if solving the best subset problem, the best results were obtained by the VS.KL algorithm. We also provide detailed description of the sweeping formalism, which is crucial for practical implementation of several selection methods. This part of the thesis is formulated in a considerably different way if compared to other relevant sources and, in addition, presents several results, which have not been published yet.

Keywords: Variable Selection, Linear Regression, Bias-Variance Tradeoff, Best Subset Selection, Forward and Backward Selection, Mixed Integer Programming, Sweeping, Principle Pivot Transform, KL Exchange Algorithm, LASSO, Relaxed LASSO.

Obsah

Úvod

Lineárne regresné modely (LRM) sú jedným zo základných prostriedkov používaných pri analýze dát a tvorbe predikčných modelov, či už v tradičných aplikáciách štatistiky alebo v oblasti strojového učenia. Hlavným dôvodom tejto obľúbenosti je ich štruktúrna jednoduchosť, ktorá umožňuje odvodiť analytické vzorce pre odhady regresných koeficientov, intervaly (alebo elipsoidy) spoľahlivosti, či predikčné intervaly. Proces trénovania LRM (t. j. výpočtu odhadov regresných koeficientov) je preto výrazne rýchlejší a jednoduchší než pri iných prístupoch, kedy zvyčajne treba aplikovať rôzne časovo náročné numerické iteračné schémy.

Daňou za jednoduchosť lineárnej závislosti medzi prediktormi (vstupnými, vysvetľujúcimi premennými) $x_1, \ldots, x_n$ a vysvetľovanou (výstupnou) premennou y je relatívne nízka zložitosť množiny hypotéz[1] generovanej LRM. Dôsledkom je nedostatočná schopnosť popísať dáta, t. j. vysoká výchylka (bias). Riešením je rozšíriť pôvodnú množinu premenných $\{x_1, \ldots, x_n\}$ pridaním nových premenných $\{u_1, \ldots, u_m\}$ vytvorených rôznymi transformáciami pôvodných. Často ide o mocniny rôznych stupňov, diferencie, logaritmy, exponenty, rôzne kombinačné premenné (napr. $x_i x_j^3$), či nejaké dvojhodnotové $(0, 1)$, tzv. „dummy“, premenné. Tento prístup je intuitívne odôvodniteľný na základe analógie s Taylorovým rozvojom funkcie $y(x_1, \ldots, x_n)$, kedy pôvodnú, vo všeobecnosti nelineárnu, závislosť od n premenných $x_1, \ldots, x_n$ transformujeme na lineárnu závislosť, ale od v princípe nekonečného počtu premenných tvaru $\prod_{j=1}^{n} x_j^{k_j}$, $k_j \in \mathbb{N}_0$.

[1] Pojem *hypotéza* tu chápeme v zmysle terminológie zaužívanej v oblasti strojového učenia. Označuje jeden konkrétny tvar štatistického modelu pre závislosť výstupnej premennej y od vstupných premenných $x_1, \ldots, x_n$. Príkladom môže byť LRM $y \sim \beta_0 + \beta_1 x$ pre nejaké fixované koeficienty β_0, β_1. Množina hypotéz generovaná LRM označuje všetky lineárne regresné modely vytvorené z danej množiny vstupných premenných. Pre vyššie uvedený príklad sú to $y \sim \beta_0 + \beta_1 x$ pre $\beta_0, \beta_1 \in \mathbb{R}$. Zložitosť množiny hypotéz súvisí s jej dimenziou, resp. počtom nezávislých parametrov modelu.

Za predpokladu, že pôvodná množina premenných $\{x_1, \ldots, x_n\}$ obsahovala všetky (pre model) relevantné premenné, doplnením transformovaných premenných $\{u_1, \ldots, u_m\}$ získame dostatočne zložitú triedu hypotéz. Vďaka tomu sme schopní natrénovať LRM vynikajúco popisujúci trénovaciu sadu dát, teda s veľmi malou trénovacou chybou. Pri malom počte trénovacích dát[2] však príliš zložitý model má tendenciu prehnane sa prispôsobiť konkrétnej realizácii trénovacej množiny. Odhadnutý model je potom výrazne ovplyvnený náhodným šumom na úkor deterministickej časti zdrojového modelu generujúceho dáta. Tento efekt, nazývaný tiež preučenie (overfitting), vedie k vysokej predikčnej (testovacej) chybe v dôsledku vysokého rozptylu (variance) odhadnutého modelu v závislosti od konkrétnej realizácie trénovacej množiny.

Vo väčšine aplikácií je cieľom minimalizovať predikčnú chybu. Použitý model teda nemôže byť príliš jednoduchý, inak by nedokázal dostatočne dobre popísať trénovacie dáta, a následne ani kvalitne predikovať. Súčasne model nemôže byť príliš zložitý, inak by predikčná chyba vzrástla kvôli preučeniu. Toto je hlavná myšlienka konceptu tzv. *bias-variance tradeoff*-u.

Existuje viacero metód ako optimalizovať zložitosť množiny hypotéz generovanej LRM. Táto práca sa zameriava na porovnanie algoritmov, ktoré regulujú zložitosť redukciou množiny premenných $\mathcal{X} = \{x_1, \ldots, x_n\} \cup \{u_1, \ldots, u_m\}$. Cieľom je odstrániť (podstatnú) časť premenných z $\mathcal{X}$ tak, aby výsledný model nebol príliš zložitý a súčasne dostatočne dobre popisoval dáta.

Jednou z výhod selekčných algoritmov v porovnaní s inými metódami minimalizácie predikčnej chyby je, že spolu s nedôležitými premennými sa odstráni aj potreba ich merania a s tým spojené náklady. Navyše, rozdelenie premenných na množinu významnú a nevýznamnú pre regresiu, doplnené aj o približné usporiadanie jednotlivých premenných z hľadiska významnosti, je vítaným pomocníkom pri interpretácii výsledného regresného modelu.

Vždy však treba mať na pamäti, že z matematického hľadiska je optimalizovanou účelovou funkciou predikčná chyba LRM. Akékoľvek analýzy výsledku smerujúce k interpretácii či popisu štruktúry (trénovacích) dát preto treba brať s istou rezervou, pretože toto nebolo cieľom optimalizácie. Takéto analýzy môžu byť len akýmsi druhoradým

[2]V praxi je skoro vždy nedostatok dát.

pomocným výstupom. Podobné upozornenie možno nájsť v rôznych formách v mnohých monografiách o štatistickom modelovaní. Krátka a výstižná verzia „Regression is for prediction, not explanation“, pochádzajúca od Wernera Stützleho[3] [8], sa mi páčila natoľko, že som si ju zvolil ako motto celej tejto práce. Analogický, avšak obšírnejší citát možno nájsť aj v knihe [14]: „The objective of statistical modeling or data analysis is to obtain information about data that may arise in the future, rather than the observed data used in the model construction itself“.

Predstavená úloha selekcie premenných je silne analogická problému optimálneho návrhu štatistického experimentu ([19, 1]). Cieľom je výberom vhodných bodov návrhu maximalizovať množstvo informácie získanej z experimentu, čo reprezentujeme určitým kritériom odvodeným od Fisherovej informačnej matice návrhu. Pri použití vhodného matematického formalizmu sa tak otvárajú možnosti pre využitie modifikovaných algoritmov z oblasti optimálneho návrhu experimentu pre selekciu premenných v LRM.

Predkladaná práca je členená nasledovne. Prvá kapitola matematicky presne formuluje úlohu selekcie premenných v LRM a následne sumarizuje teoretické poznatky potrebné pre lepšie pochopenie dôvodov selekcie premenných, ako aj detailov praktickej realizácie selekčných algoritmov. Detailne je predstavený formalizmus tzv. sweepovania, ktorý umožňuje efektívne realizovať numerické výpočty základných selekčných algoritmov.

Detailný popis analyzovaných selekčných algoritmov možno nájsť v druhej kapitole. Okrem štandardných metód je predstavených aj niekoľko vlastných návrhov algoritmov pre selekciu premenných v LRM. Tretia kapitola porovnáva uvedené selekčné algoritmy na reálnych dátach z hľadiska výpočtovej náročnosti a schopnosti zvoliť model vedúci k čo najnižšiemu odhadu predikčnej chyby.

[3]W. Stützle bol PhD. školiteľom T. Hastieho, spoluautora známej prehľadovej monografie *The elements of statistical learning* ([10]), z ktorej v mnohom čerpá aj táto práca. Citát je prevzatý z prednáškových materiálov [8] Ch. Geyera, ďalšieho zo žiakov W. Stützleho.

Kapitola 1

Teoretické aspekty selekcie premenných

1.1 Základné pojmy a značenie

Uvažujeme štandardnú úlohu najmenších štvorcov (LS) pre lineárnu regresiu. To znamená, že máme k dispozícii N párov hodnôt 1-rozmernej vysvetľovanej premennej y_i a p-tice vysvetľujúcich[1] premenných $\{x_{i1}, \ldots, x_{ip}\}$ pre $1 \leq i \leq N$. Závislosť medzi vysvetľujúcimi premennými a vysvetľovanou premennou sa snažíme modelovať pomocou lineárneho vzťahu

$$y_i \sim \beta_0 + \sum_{j=1}^{p} x_{ij}\beta_j + \epsilon_i, \quad 1 \leq i \leq N, \tag{1.1}$$

kde β_0 je intercept, β_j, $1 \leq j \leq p$, sú regresné koeficienty pre každú z vysvetľujúcich premenných a ϵ_i sú realizácie nejakej náhodnej veličiny. Pre jednoduchosť budeme v prípade potreby predpokladať, že ϵ_i sú nezávislé realizácie z rozdelenia $N(0, \sigma^2)$.

Cieľom je nájsť také hodnoty (LS odhady) $\hat{\beta}_j$, $0 \leq j \leq p$, aby suma štvorcov odchýlok (RSS[2]) bola minimálna, t. j.

$$RSS = RSS(\hat{\beta}_0, \ldots, \hat{\beta}_p) = \min_{\beta_0,\ldots,\beta_p} RSS(\beta_0, \ldots, \beta_p) = \min_{\beta_0,\ldots,\beta_p} \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2. \tag{1.2}$$

[1] V ďalšom budeme ekvivalentne zamieňať označenie *vysvetľovaná premenná* za výraz *cieľová* či *výstupná premenná*. Podobne synonymom pre *vysvetľujúcu premennú* je výraz *prediktor* alebo *regresor*.

[2] RSS = Residual sum of squares

Táto úloha sa výrazne sprehľadní v maticovom zápise, kde hodnoty vysvetľovanej premennej y_i vložíme do stĺpcového $N \times 1$ vektora $\mathbf{y}$ a zodpovedajúce hodnoty vysvetľujúcich premenných x_{ij} do $N \times (p+1)$ matice plánu $\mathbf{X}$, ktorej nultý stĺpec vyplnený hodnotami 1 zodpovedá interceptu. Vektor LS odhadov $\hat{\boldsymbol{\beta}} = (\hat{\beta}_0, \ldots, \hat{\beta}_p)^\top$ je potom riešením sústavy $p+1$ normálnych rovníc

$$\mathbf{X}^\top \mathbf{X} \boldsymbol{\beta} = \mathbf{X}^\top \mathbf{y}, \tag{1.3}$$

t. j.

$$\hat{\boldsymbol{\beta}} = \left(\mathbf{X}^\top \mathbf{X}\right)^- \mathbf{X}^\top \mathbf{y}, \tag{1.4}$$

kde $\left(\mathbf{X}^\top \mathbf{X}\right)^-$ označuje pseudoinverziu matice uvedenej v zátvorke. Pre RSS zodpovedajúceho lineárneho regresného modelu potom dostávame

$$RSS \equiv RSS(\hat{\boldsymbol{\beta}}) = \hat{\boldsymbol{\epsilon}}^\top \hat{\boldsymbol{\epsilon}} = \left(\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}\right)^\top \left(\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}\right) = \mathbf{y}^\top \left[\mathbf{I}_{p+1} - \mathbf{X}\left(\mathbf{X}^\top \mathbf{X}\right)^- \mathbf{X}^\top\right] \mathbf{y}, \tag{1.5}$$

kde $\hat{\boldsymbol{\epsilon}} = \mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}$ je LS odhad reziduálneho vektora a $\mathbf{I}_{p+1}$ označuje maticu identity rozmeru $(p+1) \times (p+1)$. V ďalšom budeme dolný index pri matici identity kvôli sprehľadneniu vynechávať za predpokladu, že jej rozmery sú zrejmé. Matica $\mathbf{P}_\mathbf{X} = \mathbf{X}\left(\mathbf{X}^\top \mathbf{X}\right)^- \mathbf{X}^\top$ je projektor na stĺpcový priestor $\mathcal{M}(\mathbf{X})$ matice $\mathbf{X}$ a $\mathbf{I} - \mathbf{P}_\mathbf{X}$ je projektor na jeho ortogonálny doplnok $\mathcal{M}(\mathbf{X})^\perp$. Platí teda $\hat{\boldsymbol{\epsilon}} = (\mathbf{I} - \mathbf{P}_\mathbf{X})\,\mathbf{y}$.

Platí, že pridanie interceptu do modelu je ekvivalentné vycentrovaniu zložiek vektora $\mathbf{y}$ a stĺpcov $\mathbf{x}_j, j \in \{1, \ldots, p\}$, matice plánu $\mathbf{X}$ tak, aby ich výberové aritmetické priemery $\overline{x}_j$ a $\overline{y}$ boli nulové[3]. Táto vlastnosť je daná tým, že po eliminácii prvého stĺpca v sústave normálnych rovníc (1.3) zodpovedajúceho interceptu, dostaneme $\hat{\beta}_0 = \overline{y}$ a sústavu p normálnych rovníc pre vycentrované premenné bez interceptu. V ďalších úvahách preto vždy predpokladáme, že všetky premenné boli na začiatku výpočtu vycentrované transformáciami $\mathbf{x}_j \mapsto \mathbf{x}_j - \overline{x}_j, j \in \{1, \ldots, p\}$ a $\mathbf{y} \mapsto \mathbf{y} - \overline{y}$. Vzhľadom na to uvažujeme ďalej len modely bez interceptu, čo má za následok, že v predchádzajúcich úvahách sa nahradí $p+1 \mapsto p$.

Vo všeobecnosti možno problém selekcie premenných slovne formulovať ako proces výberu podmnožiny sady dostupných vstupných premenných $\{x_1, \ldots, x_p\}$ použitej pri

[3]Centruje sa každý stĺpec jednotlivo. Intercept je z modelu odstránený, rovnako aj jemu zodpovedajúci stĺpec v matici $\mathbf{X}$.

konštrukcii štatistického modelu. Hlavným cieľom tohto úkonu je zvyčajne, a z hľadiska tejto práce výhradne, minimalizovať odhad predikčnej chyby z dát odhadnutého modelu. Ak sa totiž z modelu odstránia tie prediktory, ktoré sú pre modelovanie výstupnej premennej nevýznamné alebo (v prítomnosti iných prediktorov) nadbytočné, potlačí sa vplyv šumu (náhodnej zložky v dátach) na hodnotu odhadov parametrov určujúcich model. Tento jav detailnejšie rozoberieme v časti 1.2. Ďalším zdrojom motivácie pre selekciu premenných môže byť napr. aj snaha zlepšiť interpretovateľnosť modelu či numerickú stabilitu výpočtu.

Z matematického pohľadu je teda selekcia premenných optimalizačnou úlohou na diskrétnej množine všetkých 2^p podmnožín množiny dostupných premenných $\mathcal{X} \equiv \{x_1, \ldots, x_p\}$. Označme $\Omega \equiv \{1, \ldots, p\}$ množinu indexov všetkých vstupných premenných a symbolom $\langle p \rangle$ množinu všetkých jej 2^p podmnožín. Pre skrátenie matematického zápisu a vzhľadom na bijektívnosť zobrazenia $j \mapsto x_j$, budeme ďalej množiny premenných ekvivalentne označovať takmer výhradne pomocou zodpovedajúcich podmnožín indexovej množiny Ω, teda prvkov množiny $\langle p \rangle$. LRM na množine premenných $\mathcal{X}$, ktorého regresné koeficienty sú dané LS odhadmi $\hat{\boldsymbol{\beta}}$ (1.4), budeme označovať ako úplný (FULL) model.

Zvoľme ľubovoľnú množinu indexov $S \subseteq \Omega$ a označme $\mathbf{X}_S$ maticu tvorenú stĺpcami $\mathbf{x}_j$, $j \in S$, matice plánu $\mathbf{X}$. Indexová množina S, reprezentujúca podľa predchádzajúcej dohody množinu premenných $\mathcal{X}_S \equiv \{x_j, j \in S\} \subseteq \mathcal{X}$, určuje submodel úplného modelu pri ohraničeniach $\beta_j = 0$, $j \notin S$. Ide o LRM na množine premenných $\mathcal{X}$, ktorého nenulové regresné koeficienty β_j, $j \in S$ sú ako vektor $\hat{\boldsymbol{\beta}}_S$ dané LS odhadom (1.4) pri nahradení $\mathbf{X}$ za $\mathbf{X}_S$. Pod pojmom submodel budeme ďalej rozumieť submodely tohto špeciálneho tvaru. Symbolom S budeme označovať jednak množinu (indexov) premenných v submodeli, jednak samotný submodel, pretože vzhľadom na uvedené je množinou S jednoznačne určený, ak $\mathbf{X}_S$ má plnú hodnosť. Matica $\mathbf{X}_S$ reprezentuje maticu plánu submodelu S. Špeciálnymi prípadmi submodelov sú uplný model $S_{\text{FULL}} = \Omega$ a prázdny submodel $S_{\text{NULL}} = \emptyset$. Množinu všetkých 2^p submodelov (indexových množín) S budeme označovať $\mathcal{S}$. Platí teda $\mathcal{S} = \langle p \rangle$. Označenie $\langle p \rangle$ budeme používať, ak budeme chcieť vyzdvihnúť indexový charakter množín S, prípadne explicitne zdôrazniť závislosť na parametri p.

Matice označujeme veľkými tučnými písmenami (napr. $\mathbf{A}$), ich podmatice, určené množinami riadkových indexov α_1 a stĺpcových indexov α_2, pridaním týchto symbolov vpravo dole (napr. $\mathbf{A}_{\alpha_1,\alpha_2}$). Všetky riadky/stĺpce matice sa vyznačia symbolom $\bullet$ (napr. $\mathbf{A}_{\bullet,j}$ - j-ty stĺpec matice $\mathbf{A}$); doplnková množina $\overline{\alpha}$ označuje všetky indexy riadkov/stĺpcov, ktoré nie sú v α; i,j-ty prvok matice značíme buď malými písmenami s indexami vpravo dole (napr. a_{ij}) alebo ako 1×1 podmaticu (napr. $\mathbf{A}_{i,j}$). Analogicky značíme vektory malými tučnými písmenami (napr. $\mathbf{v}$); indexovou množinou α vybranú časť vektora $\mathbf{v}$ ako $\mathbf{v}_\alpha$ a jeho j-tu zložku ako v_j. Symbolmi $\mathbf{0}_{m\times n}$ a $\mathbf{I}_m$ označujeme nulovú maticu rozmerov $m \times n$ resp. maticu identity rozmeru $m \times m$. Ak sú rozmery týchto matíc zrejmé, zvyčajne ich explicitne nepíšeme. Veľkosť množiny (počet jej prvkov) vyjadrujeme znakom $|\ |$ (napr. $|\alpha|$), ℓ_p normu vektora symbolom $||\ ||_p$. Symbol $\square$ vymedzuje koniec dôkazu vety alebo lemy.

1.2 Bias-variance trade-off

Cieľom štatistického modelovania je skonštruovať model, ktorý bude schopný pre nové pozorovania generovať kvalitné odhady (predikcie) $\hat{y}$ skutočných výstupov y. V tejto časti ukážeme, ako možno vyhodnocovať kvalitu predikcie a popíšeme hlavné faktory, ktoré ju ovplyvňujú. Viac informácií k tejto téme možno nájsť napr. v [10] (kapitola 7), [7] (časť 12.7) a [8].

Nech neznámy model $\mathcal{M}$, ktorým sa riadi závislosť medzi vektorom vstupov $\mathbf{x}$ a výstupom y, ako realizácie náhodnej premennej Y, má tvar

$$Y = \mathrm{f}(\mathbf{x}) + \epsilon, \tag{1.6}$$

kde ϵ je náhodná premenná nezávislá od $\mathbf{x}$, $\mathsf{E}(\epsilon) = 0$, $\mathsf{Var}(\epsilon) = \sigma^2$ a $\mathrm{f}(\mathbf{x})$ je nejaká nenáhodná (ale nám neznáma) funkcia. Pozorované dáta, generované z modelu $\mathcal{M}$, sa budeme snažiť opísať pomocou závislosti

$$y = \mathrm{h}(\mathbf{x}), \tag{1.7}$$

kde $\mathrm{h}(\mathbf{x})$ sú funkcie z nami definovanej množiny $\mathcal{H}$. Príkladom môže byť množina lineárnych funkcií $\mathcal{H} = \left\{\mathbf{x}^\top\boldsymbol{\beta},\ \boldsymbol{\beta} \in \mathbb{R}^p\right\}$ pri jednoduchej lineárnej regresii. Funkcie $\mathrm{h}(\mathbf{x})$ sa zvyknú označovať ako hypotézy. Na základe množiny N pozorovaní $\mathcal{T} = \{\mathbf{x}_i, y_i\}_{i=1}^N$

vyberieme jednu funkciu z množiny hypotéz $\mathcal{H}$, ktorú použijeme ako odhad funkcie $\mathrm{f}(\mathbf{x})$ a označíme ju $\hat{\mathrm{f}}(\mathbf{x})$. Napríklad, pri lineárnej regresii s $\mathcal{H} = \left\{\mathbf{x}^\top \boldsymbol{\beta},\, \boldsymbol{\beta} \in \mathbb{R}^p\right\}$ môžeme zvoliť $\hat{\mathrm{f}}(\mathbf{x}) = \mathbf{x}^\top \hat{\boldsymbol{\beta}}$, kde $\hat{\boldsymbol{\beta}}$ je LS odhad regresných koeficientov vypočítaný na základe množiny pozorovaní $\mathcal{T}$. Počet stupňov voľnosti pri výbere odhadu $\hat{\mathrm{f}}(\mathbf{x}) \in \mathcal{H}$ definuje zložitosť množiny hypotéz $\mathcal{H}$.

Uvažujme, pre všeobecnosť, že pozorované vstupy $\mathbf{x}$ sú nezávislými realizáciami náhodného vektora $\mathbb{X}$. Potom množina $\mathcal{T}$ je realizáciou náhodného výberu $\mathbb{T}$ veľkosti N pre náhodný vektor $(\mathbb{X}, Y)^\top$. Označme

$$\mathrm{h}^* = \underset{\mathrm{h}\in\mathcal{H}}{\operatorname{argmin}}\, \mathsf{E}_{\mathbb{X}} \left[(\mathrm{h}(\mathbb{X}) - \mathrm{f}(\mathbb{X}))^2\right] \tag{1.8}$$

tú funkciu z množiny hypotéz $\mathcal{H}$, ktorá teoreticky najlepšie aproximuje funkciu f z modelu $\mathcal{M}$ generujúceho dáta. Očakávaná predikčná chyba EPE (expected prediction error) pre nové pozorovanie (mimo $\mathcal{T}$) je definovaná (porov. [10], kapitola 7)

$$\mathrm{EPE} \equiv \mathsf{E}_{\mathbb{T}} \mathsf{E}_{\mathbb{X},Y} \left[\left(Y - \hat{\mathrm{f}}_{\mathcal{T}}(\mathbb{X})\right)^2\right], \tag{1.9}$$

kde $(\mathbb{X}, Y)^\top \notin \mathbb{T}$ zodpovedá novému pozorovaniu a indexom $\mathcal{T}$ sme zdôraznili, že náš výber odhadu $\hat{\mathrm{f}}_{\mathcal{T}}$ závisí od konkrétnej realizácie $\mathcal{T}$ náhodného výberu $\mathbb{T}$. Veličina EPE udáva strednú hodnotu chyby $(y - \hat{y})^2$ našich predikcií $\hat{y}$ pre nové dáta (nezahrnuté v $\mathcal{T}$) v porovnaní so skutočne pozorovanými hodnotami výstupu y ešte predtým, než máme k dispozícii realizáciu množiny $\mathcal{T}$. Ak model $\mathcal{M}$ generujúci dáta je daný, EPE závisí len od pravidla (algoritmu) pre výber odhadu $\hat{\mathrm{f}}_{\mathcal{T}}$. Platí

$$\begin{aligned}
\mathrm{EPE} =& \mathsf{E}_{\mathbb{T}} \mathsf{E}_{\mathbb{X},Y} \left[\left(Y - \mathrm{f}(\mathbb{X}) + \mathrm{f}(\mathbb{X}) - \mathrm{h}^*(\mathbb{X}) + \mathrm{h}^*(\mathbb{X}) - \mathsf{E}_{\mathbb{T}}\left(\hat{\mathrm{f}}_{\mathcal{T}}(\mathbb{X})\right) + \right.\right. \\
& \left.\left. \mathsf{E}_{\mathbb{T}}\left(\hat{\mathrm{f}}_{\mathcal{T}}(\mathbb{X})\right) - \hat{\mathrm{f}}_{\mathcal{T}}(\mathbb{X})\right)^2\right] \\
=& \mathsf{E}_{\mathbb{X},Y}\left[(Y - \mathrm{f}(\mathbb{X}))^2\right] + \mathsf{E}_{\mathbb{X}} \mathsf{E}_{\mathbb{T}}\left[\left(\hat{\mathrm{f}}_{\mathcal{T}}(\mathbb{X}) - \mathsf{E}_{\mathbb{T}}\left(\hat{\mathrm{f}}_{\mathcal{T}}(\mathbb{X})\right)\right)^2\right] + \\
& \mathsf{E}_{\mathbb{X}}\left[(\mathrm{f}(\mathbb{X}) - \mathrm{h}^*(\mathbb{X}))^2\right] + \mathsf{E}_{\mathbb{X}}\left[\left(\mathrm{h}^*(\mathbb{X}) - \mathsf{E}_{\mathbb{T}}\left(\hat{\mathrm{f}}_{\mathcal{T}}(\mathbb{X})\right)\right)^2\right] \\
=& \sigma^2 + \mathsf{E}_{\mathbb{X}}\left[\mathsf{Var}_{\mathbb{T}}\left(\hat{\mathrm{f}}_{\mathcal{T}}(\mathbb{X})\right)\right] + \mathsf{E}_{\mathbb{X}}\left[(\mathit{Model\ Bias})^2\right] + \\
& \mathsf{E}_{\mathbb{X}}\left[(\mathit{Estimation\ Bias})^2\right].
\end{aligned} \tag{1.10}$$

Prvý člen (σ^2) reprezentuje varianciu výstupnej premennej Y spôsobenú náhodným šumom ϵ. Túto zložku štatistickým modelovaním nedokážeme ovplyvniť, lebo je daná

priamo zdrojovým modelom $\mathcal{M}$. Voľbou množiny hypotéz $\mathcal{H}$ a algoritmu výberu funkcie $\hat{\mathrm{f}}$ však vieme meniť zvyšné tri príspevky. Prvý z nich, $\mathsf{E}_{\mathbb{X}}\left[\mathsf{Var}_{\mathbb{T}}\left(\hat{\mathrm{f}}_{\mathcal{T}}(\mathbb{X})\right)\right]$, je variancia predikcií $\hat{y} = \hat{\mathrm{f}}_{\mathcal{T}}(\mathbf{x})$ v závislosti od pozorovanej realizácie $\mathcal{T}$ náhodného výberu $\mathbb{T}$, na základe ktorej vyberáme $\hat{\mathrm{f}} \in \mathcal{H}$. Posledné dva výrazy súvisia s výchylkou (bias) $\mathrm{f}(\mathbf{x}) - \mathsf{E}_{\mathbb{T}}\left[\hat{\mathrm{f}}_{\mathcal{T}}(\mathbf{x})\right]$ priebehu strednej úrovne odhadnutej funkcie $\mathsf{E}_{\mathbb{T}}\left[\hat{\mathrm{f}}_{\mathcal{T}}(\mathbf{x})\right]$ oproti skutočnej závislosti $\mathrm{f}(\mathbf{x})$. Túto výchylku možno rozdeliť na dve zložky, označené ako *Model Bias* a *Estimation Bias*. Prvá zložka, $Model\ Bias = \mathrm{f}(\mathbf{x}) - \mathrm{h}^*(\mathbf{x})$, je príspevok k výchylke spôsobený tým, že hypotézy $\mathrm{h} \in \mathcal{H}$ nedokážu ideálne aproximovať zdrojovú funkciu $\mathrm{f}(\mathbf{x})$. Inak povedané, je to príspevok v dôsledku toho, že zložitosť množiny hypotéz $\mathcal{H}$ nie je dostatočná na opis závislosti $\mathrm{f}(\mathbf{x})$[4]. Ak $\mathrm{f} \in \mathcal{H}$, *Model Bias* je nulový. V realite však nielenže nepoznáme f, zvyčajne ani nevieme určiť, do akej triedy funkcií patrí. Nevieme preto garantovať splnenie podmienky $\mathrm{f} \in \mathcal{H}$, čoho výsledkom je $\mathsf{E}_{\mathbb{X}}\left[(Model\ Bias)^2\right] > 0$. Druhá zložka, $Estimation\ Bias = \mathrm{h}^*(\mathbb{X}) - \mathsf{E}_{\mathbb{T}}\left(\hat{\mathrm{f}}_{\mathcal{T}}(\mathbb{X})\right)$, závisí od spôsobu výberu odhadu $\hat{\mathrm{f}} \in \mathcal{H}$ a je nenulová, ak $\hat{\mathrm{f}}(\mathbf{x})$ je vychýleným odhadom funkcie $\mathrm{h}^*(\mathbf{x})$. Napríklad, ak $\mathrm{f}(\mathbf{x}) = \mathbf{x}^\top\boldsymbol{\beta}$ pre nejaké fixné $\boldsymbol{\beta} \in \mathbb{R}^p$, LS odhad $\hat{\mathrm{f}}(\mathbf{x}) = \mathbf{x}^\top\hat{\boldsymbol{\beta}}^{\mathrm{LS}}$ je nevychýlený ($Estimation\ Bias = 0$[5]), ale tzv. LASSO odhad $\hat{\mathrm{f}}(\mathbf{x}) = \mathbf{x}^\top\hat{\boldsymbol{\beta}}^{\mathrm{LASSO}}$ (viď. časť 2.4.2) je vychýlený ($Estimation\ Bias \neq 0$).

Pre jednoduchosť ďalej predpokladajme, že $Estimation\ Bias = 0$. Čím väčšia je zložitosť množiny hypotéz $\mathcal{H}$, tým lepšie riešenie $\mathrm{h}^*(\mathbf{x})$ v úlohe (1.8) vieme nájsť. Preto príspevok výchylky k očakávanej predikčnej chybe, $\mathsf{E}_{\mathbb{X}}\left[(Model\ Bias)^2\right]$, klesá. Avšak s rastom zložitosti množiny $\mathcal{H}$ sa stále lepšie výberom odhadu $\hat{\mathrm{f}}(\mathbf{x})$ vieme prispôsobiť danej množine realizovaných pozorovaní $\mathcal{T}$. Odhady $\hat{\mathrm{f}}(\mathbf{x})$ preto budú pri zmenách v množine $\mathcal{T}$ výraznejšie fluktuovať, čo sa prejaví v náraste variancie $\mathsf{E}_{\mathbb{X}}\left[\mathsf{Var}_{\mathbb{T}}\left(\hat{\mathrm{f}}_{\mathcal{T}}(\mathbb{X})\right)\right]$. Zmenou zložitosti množiny hypotéz $\mathcal{H}$ teda nemožno súčasne znižovať príspevok od bias-u aj variancie, tieto dva efekty sú vzájomne protichodné. Toto pozorovanie je známe ako *bias-variance trade-off*. Nízka zložitosť množiny $\mathcal{H}$ spôsobuje vysokú výchylku pri nízkej variancii (oblasť podučenia - *underfitting*); vysoká zložitosť, naopak, nízku výchylku, no vysokú varianciu (oblasť preučenia - *overfitting*). Očakávaná predikčná chyba, ako súčet oboch príspevkov, má preto minimum pre množinu hypotéz s netriviálnou zložitosťou. Množinu hypotéz $\mathcal{H}$ sa snažíme konštruovať tak, aby zod-

[4]Napríklad $\mathrm{f}(x) = x^2$ a $\mathcal{H} = \{\beta_0 + \beta_1 x; \beta_0, \beta_1 \in \mathbb{R}\}$.

[5]Myslí sa nulová funkcia.

povedala tomuto minimu. Vplyv variancie je tým silnejší, čím menší je počet dát N v množine $\mathcal{T}$, pre $N \to \infty$ variančná zložka chyby $\mathsf{E}_{\mathbb{X}}\left[\mathsf{Var}_{\mathbb{T}}\left(\hat{\mathrm{f}}_{\mathcal{T}}(\mathbb{X})\right)\right]$ zaniká. Príspevok k chybe od *Model Bias* je vzhľadom na N konštantný.

Pri reálnom výpočte $\mathrm{f}(\mathbf{x})$, a teda ani $\mathrm{h}^*(\mathbf{x})$, nepoznáme, preto očakávanú predikčnú chybu EPE nevieme vypočítať. Množiny hypotéz $\mathcal{H}$ s optimálnou zložitosťou preto hľadáme minimalizáciou nejakého odhadu očakávanej predikčnej chyby, ktorý označíme $\widehat{\mathrm{EPE}}$. Štandardný spôsob výpočtu $\widehat{\mathrm{EPE}}$ ja taký, že dostupnú množinu N pozorovaní $\mathcal{O} = \{\mathbf{x}_i, y_i\}_{i=1}^N$ náhodne rozdelíme na tri časti - trénovaciu $\mathcal{O}^{\mathrm{Train}}$, validačnú $\mathcal{O}^{\mathrm{Val}}$ a testovaciu $\mathcal{O}^{\mathrm{Test}}$ (zodpovedajúcich veľkostí N_{Train}, N_{Val} a N_{Test}, pričom $N_{\mathrm{Train}} + N_{\mathrm{Val}} + N_{\mathrm{Test}} = N$). Účelom trénovania je z danej množiny hypotéz $\mathcal{H}$ vybrať funkciu $\hat{\mathrm{f}}(\mathbf{x})$ v istom zmysle najlepšie opisujúcu (fitujúcu) dáta z trénovacej množiny $\mathcal{O}^{\mathrm{Train}}$. Toto zodpovedá výberu $\hat{\mathrm{f}}_{\mathcal{T}}(\mathbf{x})$ na základe množiny $\mathcal{T}$, ako bolo diskutované vyššie. Máme množinu q množín hypotéz $\{\mathcal{H}^{(j)}\}_{j=1}^q$[6], pre každú z nich pomocou trénovania nájdeme optimálny fit $\hat{\mathrm{f}}^{(j)}(\mathbf{x})$. Pri validácii sa z $\{\mathcal{H}^{(j)}\}_{j=1}^q$ vyberie tá množina hypotéz $\mathcal{H}^{(j^*)}$, ktorá pri fitovaní dát z validačnej množiny pomocou $\hat{\mathrm{f}}^{(j)}(\mathbf{x})$ dáva najnižší odhad očakávanej predikčnej chyby $\widehat{\mathrm{EPE}}$. Tento krok reprezentuje výber optimálnej množiny hypotéz na základe minimalizácie EPE (bias-variance krivky). Výsledkom je predikčný model daný funkciou $\hat{\mathrm{f}}^{(j^*)}(\mathbf{x})$. Odhad jeho očakávanej predikčnej chyby sa vypočíta fitovaním dát z testovacej množiny. Rozdelenie dát na tri nezávislé množiny je potrebné kvôli tomu, aby sa potlačilo vychýlenie trénovacích a validačných odhadov $\widehat{\mathrm{EPE}}$ smerom k nižším hodnotám. Toto vychýlenie je spôsobené tým, že odhad $\widehat{\mathrm{EPE}}$ je získaný pomocou funkcie $\hat{\mathrm{f}}(\mathbf{x})$, ktorá bola vybraná jeho minimalizáciou.

Odhady očakávanej predikčnej chyby konštruujeme v tvare sumy štvorcov rezíduí pripadajúcej na jedno pozorovanie

$$\widehat{\mathrm{EPE}} = \mathrm{MSE} \equiv \frac{RSS(\mathrm{h})}{M} = \frac{1}{M}\sum_{i=1}^{M}\left(y_i - \mathrm{h}(\mathbf{x}_i)\right)^2, \tag{1.11}$$

kde M je počet pozorovaní v danej množine. Pre LRM máme

$$\begin{aligned}
\mathrm{MSE}_{\mathrm{Train}}(\boldsymbol{\beta}) &= RSS^{\mathrm{Train}}(\boldsymbol{\beta})/N_{\mathrm{Train}} = ||\mathbf{y}^{\mathrm{Train}} - \mathbf{X}^{\mathrm{Train}}\boldsymbol{\beta}||_2^2/N_{\mathrm{Train}}, \\
\mathrm{MSE}_{\mathrm{Val}}(\boldsymbol{\beta}) &= RSS^{\mathrm{Val}}(\boldsymbol{\beta})/N_{\mathrm{Val}} = ||\mathbf{y}^{\mathrm{Val}} - \mathbf{X}^{\mathrm{Val}}\boldsymbol{\beta}||_2^2/N_{\mathrm{Val}}, \\
\mathrm{MSE}_{\mathrm{Test}}(\boldsymbol{\beta}) &= RSS^{\mathrm{Test}}(\boldsymbol{\beta})/N_{\mathrm{Test}} = ||\mathbf{y}^{\mathrm{Test}} - \mathbf{X}^{\mathrm{Test}}\boldsymbol{\beta}||_2^2/N_{\mathrm{Test}},
\end{aligned} \tag{1.12}$$

[6]Pri selekcii premenných v LRM (viď. časť 1.3) sú množiny hypotéz $\mathcal{H}^{(k)}$ dané množinami lineárnych funkcií $\{\mathbf{x}^\top\boldsymbol{\beta}; \boldsymbol{\beta} \in \mathbb{R}^p, ||\boldsymbol{\beta}||_0 \leq k\}$.

pričom platí konvencia v značení $RSS^{\text{Train}}(\boldsymbol{\beta}) \equiv RSS(\boldsymbol{\beta})$ a $\left\{\mathbf{X}^{\text{Train}}, \mathbf{y}^{\text{Train}}\right\}$, $\left\{\mathbf{X}^{\text{Val}}, \mathbf{y}^{\text{Val}}\right\}$, $\left\{\mathbf{X}^{\text{Test}}, \mathbf{y}^{\text{Test}}\right\}$ sú matice plánu a vektory výstupov pre trénovaciu, validačnú a testovaciu množinu.

Ak je dát nedostatok, možno sa obmedziť len na vytvorenie trénovacej a testovacej množiny. Výber optimálnej množiny hypotéz $\mathcal{H}^*$ sa vykoná len na základe pozorovaní z trénovacej množiny, a to napr. crossvalidáciou alebo pomocou niektorého z informačných kritérií (IC). Dve základné informačné kritériá - Akaikeho (AIC) a Bayesovo (BIC) sú predstavené nižšie.

1.2.1 Informačné kritériá

Ďalej budeme uvažovať len množiny hypotéz $\mathcal{H}$ zodpovedajúce LRM. Všeobecnejšie výsledky môže čitateľ nájsť napr. v [14], prípadne v [10]. Máme danú trénovaciu množinu $\mathcal{O}^{\text{Train}}$ s N_{Train} pozorovaniami, maticou plánu $\mathbf{X}^{\text{Train}}$ a výstupným vektorom $\mathbf{y}^{\text{Train}}$, pričom počet prediktorov je p. Pre skrátenie zápisu budeme ďalej doplnkové označenie „Train“ vynechávať. Len na základe trénovacej množiny sa snažíme určiť, ktorý z 2^p submodelov $S \in \mathcal{S}$ dokáže v istom zmysle „najlepšie“(nie nutne z pohľadu očakávanej predikčnej chyby) simulovať správanie zdrojového modelu, z ktorého sú generované dáta. Žiadame, aby zvolený model mal túto vlastnosť najmä pre nové, dosiaľ nepozorované dáta. Každému submodelu S je priradený LS odhad vektora regresných koeficientov $\hat{\boldsymbol{\beta}}_S$ a hodnota $RSS_S = RSS\left(\hat{\boldsymbol{\beta}}_S\right) = ||\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}_S||_2^2$.

Definujme tzv. Bayesovské informačné kritérium (BIC) a Akaikeho informačné kritérium (AIC) vzťahmi

$$\text{BIC}_S = \ln\left(\frac{RSS_S}{N}\right) + \ln(N)\frac{k}{N}, \qquad \text{AIC}_S = \ln\left(\frac{RSS_S}{N}\right) + 2\frac{k}{N}, \tag{1.13}$$

kde $k = |S|$ je počet prediktorov v submodeli S. Číslo k určuje zložitosť (počet stupňov voľnosti) množiny hypotéz $\mathcal{H}_S = \left\{\mathbf{x}^\top\boldsymbol{\beta}; \boldsymbol{\beta} \in \mathbb{R}^p, \beta_j = 0, j \notin S,\right\}$ zodpovedajúcej submodelu S. Zvolené IC generuje usporiadanie modelov $S \in \mathcal{S}$ podľa hodnôt IC_S, pričom platí, že čím nižšia hodnota IC_S, tým „kvalitnejší“ je model S. Optimálnym modelom je $S^* = \underset{S\in\mathcal{S}}{\operatorname{argmin}}\,\text{IC}_S$. V reálnom prípade ($N \gg 8$) BIC vždy vyberá modely S^* s menším počtom prediktorov $|S^*|$ než AIC, pretože penalizácia kvôli zložitosti $\mathcal{H}_S$ je v dôsledku $\ln(N) > 2$ pri BIC (výrazne) väčšia. Tiež vidno, že v rámci množiny

$\mathcal{S}_k = \{S \in \mathcal{S}; |S| = k\}$ modelov rovnakej veľkosti k sa usporiadanie podľa IC_S redukuje na usporiadanie podľa RSS_S.

Aj napriek tomu, že vzorce pre AIC a BIC sú veľmi podobné, majú konceptuálne značne rozdielny pôvod. BIC porovnáva modely S na základe aposteriórnych pravdepodobností $\mathsf{P}(S|\mathcal{O})$ týchto modelov po pozorovaní realizácie dát v podobe trénovacej množiny $\mathcal{O}$. Ak pred získaním dát bola naša vedomosť reprezentovaná rovnomerným rozdelením pravdepodobností $\mathsf{P}(S) = 1/|\mathcal{S}| = 2^{-p}$, pre ľubovoľné $S \in \mathcal{S}$, potom pri istých netriviálnych aproximáciách (viď napr. [10], časť 7.7 alebo [14]) možno ukázať, že aposteriórna pravdepodobnosť $\mathsf{P}(S|\mathcal{O})$ súvisí s BIC_S podľa vzťahu

$$\mathsf{P}(S|\mathcal{O}) = \frac{\mathrm{e}^{-\frac{N}{2}\mathrm{BIC}_S}}{\sum_{S' \in \mathcal{S}} \mathrm{e}^{-\frac{N}{2}\mathrm{BIC}_{S'}}}. \tag{1.14}$$

Minimalizáciou BIC_S sa teda vyberá model S^* s najvyššou aposteriórnou pravdepodobnosťou v rámci množiny $\mathcal{S}$.

Pre pochopenie konceptu AIC je potrebné uvažovať o štatistickom modeli ako o združenej pravdepodobnostnej distribúcii $\mathsf{P}(\mathbb{X}, Y)$ náhodného vektora vstupov $\mathbb{X}$ a výstupnej náhodnej premennej Y. Nech zdrojový model je daný rozdelením $\mathsf{P}_0(\mathbb{X}, Y)$. Minimalizáciou AIC sa snažíme vybrať ten model z množiny hypotéz, ktorému zodpovedá pravdepodobnostné rozdelenie $\hat{\mathsf{P}}(\mathbb{X}, Y)$, ktoré je v zmysle tzv. *Kullback–Leiblerovej informácie* (KLI)[7] najbližšie k $\mathsf{P}_0(\mathbb{X}, Y)$. Vhodne nanormované AIC je len odhadom KLI, lebo $\mathsf{P}_0(\mathbb{X}, Y)$ nepoznáme.

BIC a AIC majú nasledovné vlastnosti (porov. [8, 10]). Ak sa zdrojový model $\mathcal{M}$ nachádza v uvažovanej množine hypotéz, $N \to \infty$ a $p \ll N$, BIC zvolí $\mathcal{M}$ s pravdepodobnosťou 1. BIC sa teda zameriava na to, aby detailne odhadol štruktúru závislosti zdrojového modelu. V realite však sotva bude nami zvolená množina hypotéz pokrývať $\mathcal{M}$. Niektoré premenné, prítomné v $\mathcal{M}$, môžu chýbať; prítomné môžu byť naopak iné premenné, ktoré su vysoko korelované s výstupom Y, a pritom sa v $\mathcal{M}$ nenachádzajú. Je teda zrejme vhodnejšie zamerať sa na to, aby sme dokázali správanie $\mathcal{M}$ čo najlepšie reprodukovať, aj keby to malo znamenať použitie štruktúrne veľmi odlišného modelu[8]. Tomuto prístupu by mal zodpovedať odhad modelu pomocou AIC. Pre LRM

[7] Viď [14].

[8] Tu sa opäť dostávame k citátu W. Stützleho: „Regression is for prediction, not explanation“. Z hľadiska predikcie vedomosť o vnútornej štruktúre modelu nie je prioritná, pri interpretácii áno.

v limite $N \to \infty$ platí, že výber modelu pomocou AIC zodpovedá tzv. leave-one-out crossvalidácii.

1.3 Matematická formulácia úlohy selekcie premenných

Presnejšiu matematickú formuláciu úlohy selekcie premenných v LRM získame, keď si zvolíme konkrétny spôsob výpočtu odhadu predikčnej chyby. Pre tieto účely budeme pre zjednodušenie zápisu ďalej symbolmi bez indexu ($\mathbf{X}$, $\mathbf{y}$) označovať veličiny charakterizujúce trénovaciu množinu, dolným indexom V potom ich ekvivalenty na validačnej množine ($\mathbf{X}_V$, $\mathbf{y}_V$). Štandardne sa selekcia premenných formuluje ako úloha

$$\begin{aligned} &\min_{\boldsymbol{\beta}} ||\mathbf{y} - \mathbf{X}\boldsymbol{\beta}||_2^2 \\ &s.t. \quad ||\boldsymbol{\beta}||_0 \leq k, \end{aligned} \tag{1.15}$$

kde $||\boldsymbol{\beta}||_0$ je ℓ_0 norma vektora $\boldsymbol{\beta}$, t. j. počet jeho nenulových zložiek. Cieľom je teda nájsť submodel S s minimálnym (trénovacím) RSS v rámci triedy $\mathcal{S}_k = \{S \in \mathcal{S} : |S| = k\}$ submodelov zvolenej veľkosti $k \leq p$ [9].

Pre účely využita optimalizačných solverov môže byť výhodné preformulovanie (1.15) do tvaru MIO (*mixed integer optimization*) úlohy. Jednou z možností je variant navrhnutý v [2]

$$\begin{aligned} &\min_{\boldsymbol{\beta},\mathbf{z}} ||\mathbf{y} - \mathbf{X}\boldsymbol{\beta}||_2^2 \\ &s.t. \quad -M_U z_i \leq \beta_i \leq M_U z_i, \quad i = 1, \ldots, p, \\ &\qquad z_i \in \{0, 1\}, \quad i = 1, \ldots, p, \\ &\qquad \sum_{i=1}^{p} z_i \leq k, \end{aligned} \tag{1.16}$$

kde z_i je indikátor zahrnutia premennej i v modeli a M_U je dostatočne veľká konštanta. Aby úlohy (1.15) a (1.16) boli ekvivalentné, musí byť splnená podmienka $M_U > ||\hat{\boldsymbol{\beta}}||_\infty$,

Výsledky získané minmalizáciou predikčnej chyby môžu byť preto z pohľadu interpretácie štruktúry zdrojového modelu zavádzajúce.

[9] Podmienku $||\boldsymbol{\beta}||_0 \leq k$ v (1.15) vo všeobecnosti nemožno nahradiť $||\boldsymbol{\beta}||_0 = k$, pretože môže byť optimálne niektoré regresné koeficienty v submodeli zvoliť nulové.

kde $\hat{\boldsymbol{\beta}}$ je riešením (1.15). Postup, ako správne odhadnúť M_U bez znalosti $\hat{\boldsymbol{\beta}}$, ako aj prakticky implementovať výpočet MIO úlohy (1.16) je uvedený v časti 2.1.2.

Tento prístup zodpovedá použitiu niektorého z informačných kritérií. V rámci triedy $\mathcal{S}_k$ submodelov s rovnakým počtom premenných k sa porovnanie podľa hodnoty IC redukuje na usporiadanie podľa hodnoty RSS, kde najnižšia hodnota RSS zodpovedá „najkvalitnejšiemu“ modelu. Vyriešením (1.15) pre $k \in \{1, \ldots, p\}$ sa získa postupnosť modelov $\{S_k\}_{k=1}^{p}$ s minimálnym RSS v rámci tried $\mathcal{S}_k$. Z postupnosti submodelov $\{S_k\}_{k=1}^{p}$ sa nakoniec zvolí celkový víťaz porovnaním hodnôt ich informačných kritérií alebo (cross)validačných chýb.

Pokiaľ máme záujem vhodným výberom submodelu minimalizovať validačnú chybu ako odhad predikčnej chyby, postup z predchádzajúceho odstavca zvyčajne nie je optimálny. V rámci triedy $\mathcal{S}_k$ totiž existuje veľké množstvo ďalších submodelov, ktoré majú len nepatrne vyššie RSS než víťazný submodel S_k, avšak pritom môžu mať (výrazne) nižšiu validačnú chybu. Účelom validačnej množiny je pomôcť s výberom hyperparametrov modelu, ktorým je v našom prípade najmä množina premenných obsiahnutých v tomto modeli. Z teoretického hľadiska by teda selekcia premenných mala byť vykonávaná na základe validačnej množiny, zatiaľ čo trénovacia množina by mala poslúžiť len na výpočet LS odhadu vektora koeficientov $\hat{\boldsymbol{\beta}}$ pre zvolený submodel. Pokiaľ sa explicitne chceme obmedziť na triedu submodelov $\mathcal{S}_k$, matematicky dostávame úlohu

$$\begin{aligned} &\min_{\boldsymbol{\beta}} ||\mathbf{y}_V - \mathbf{X}_V\boldsymbol{\beta}||_2^2 \\ &s.t. \qquad \boldsymbol{\beta} \in \left\{\text{argmin}||\mathbf{y} - \mathbf{X}_S\boldsymbol{\beta}||_2^2 \;\; |S \in \mathcal{S}\right\} \\ &\qquad\quad ||\boldsymbol{\beta}||_0 \leq k. \end{aligned} \tag{1.17}$$

Pripomíname, že symbolmi $\mathbf{X}_V$ a $\mathbf{y}_V$ s indexom V označujeme maticu plánu resp. výstupný vektor pre pozorovania z validačnej množiny, kým symboly bez indexu ($\mathbf{X}$, $\mathbf{y}$) sú rezervované pre trénovaciu množinu. Spomenuté obmedzenie, reprezentované podmienkou $||\boldsymbol{\beta}||_0 \leq k$, však vo všeobecnosti nie je potrebné aplikovať. Validačná chyba totiž narozdiel od RSS môže pridaním premennej do submodelu narásť (preučenie). Z hľadiska selekcie premenných má teda dobrý zmysel aj formulácia

$$\begin{aligned} &\min_{\boldsymbol{\beta}} ||\mathbf{y}_V - \mathbf{X}_V\boldsymbol{\beta}||_2^2 \\ &s.t. \quad \boldsymbol{\beta} \in \left\{\text{argmin}||\mathbf{y} - \mathbf{X}_S\boldsymbol{\beta}||_2^2 \;\; |S \in \mathcal{S}\right\}, \end{aligned} \tag{1.18}$$

pretože vo všeobecnosti je globálne minimum validačnej chyby dosiahnuté pre submodel s veľkosťou $|S| < p$.

Ohraničenie pre vektor $\boldsymbol{\beta}$ v tvare argmin je v praktickom výpočte ťažko aplikovateľné, preto by bolo vhodné nahradiť ho normálnymi rovnicami (1.3). Tu však treba pamätať na to, že pre submodel S treba v (1.3) dosadiť za $\mathbf{X}$ maticu plánu $\mathbf{X}_S$ submodelu S. Pre každý submodel teda dostávame inú sadu $|S| \leq p$ normálnych rovníc

$$\mathbf{X}_S^\top \mathbf{X}_S \boldsymbol{\beta}_S = \mathbf{X}_S^\top \mathbf{y}, \tag{1.19}$$

kde $\boldsymbol{\beta}_S$ je vektor dĺžky $|S|$ obsahujúci regresné koeficienty β_j prislúchajúce premenným obsiahnutým v modeli S. Túto sadu však môžeme doplniť $p - |S|$ podmienkami

$$\beta_j = 0, \quad j \notin S, \tag{1.20}$$

pre koeficienty tých premenných, ktoré nie sú zahrnuté v S, čo je ekvivalentom podmienky $||\boldsymbol{\beta}||_0 \leq k$ pre fixovaný submodel S veľkosti $|S| = k$.

Podmienky (1.19) a (1.20) možno kompaktne formulovať v tvare

$$\mathbf{w} = \mathbf{X}^\top \mathbf{X}\boldsymbol{\beta} - \mathbf{X}^\top \mathbf{y}, \tag{1.21}$$

$$w_j \beta_j = 0, \qquad j = 1, \ldots, p, \tag{1.22}$$

$$\mathbf{w}, \boldsymbol{\beta} \in \mathbb{R}^p. \tag{1.23}$$

Podmienka komplementarity (1.22) zabezpečuje, že pre ľubovoľný (ale fixný) submodel S musí platiť $w_j = 0$ pre $j \in S$, pretože $\beta_j \in \mathbb{R}$, $j \in S$. Dostávame tak sústavu rovníc

$$\mathbf{0} = \mathbf{X}_S^\top \mathbf{X}\boldsymbol{\beta} - \mathbf{X}_S^\top \mathbf{y}, \tag{1.24}$$

ktorá vďaka rovnosti $\mathbf{X}\boldsymbol{\beta} = \mathbf{X}_S\boldsymbol{\beta}_S$ (v dôsledku (1.20)) prejde na normálne rovnice (1.19) pre submodel S. Súčasne, keďže pre $j \notin S$ je podmienka komplementarity (1.22) splnená vďaka $\beta_j = 0$, zodpovedajúce w_j môžu byť ľubovoľné. Splnenie normálnych rovníc ($w_j = 0$) je teda správne požadované len pre $j \in S$. Premenné w_j budeme ďalej označovať ako *slacky*. Ukazuje sa, že kombinácia podmienok (1.21) a (1.22) nie je len peknou formou zápisu, ale je aj základom formalizmu pre efektívny výpočet veličín $\hat{\boldsymbol{\beta}}_S, RSS_S$ v prípade postupnosti viacerých submodelov S. Tento formalizmus, tzv. *sweepovanie*, je základným kameňom viacerých algoritmov selekcie premenných v LRM použitých v tejto práci a je detailne popísaný v nasledúcej podkapitole.

Ekvivalentným prepísaním prípustnej množiny v (1.18) pomocou podmienok (1.21)-(1.23), dostávame

$$\begin{aligned} &\min_{\beta} \boldsymbol{\beta}^\top \mathbf{X}_V^\top \mathbf{X}_V \boldsymbol{\beta} - 2\mathbf{y}_V^\top \mathbf{X}_V \boldsymbol{\beta} + \mathbf{y}_V^\top \mathbf{y}_V \\ &s.t. \quad \mathbf{w} \quad = -\mathbf{X}^\top \mathbf{y} + \mathbf{X}^\top \mathbf{X} \boldsymbol{\beta}, \\ &\qquad w_j \beta_j \quad = 0, \qquad\qquad\qquad j = 1, \ldots, p, \\ &\qquad \mathbf{w}, \boldsymbol{\beta} \quad \in \mathbb{R}^p, \end{aligned} \tag{1.25}$$

kde sme súčasne rozpísali účelovú funkciu $||\mathbf{y}_V - \mathbf{X}_V \boldsymbol{\beta}||_2^2$ do tvaru kvadratickej funkcie, čím sme sa zbavili ℓ_2 normy. Navyše, $\mathbf{y}_V^\top \mathbf{y}_V$ je len aditívna konštanta, ktorá neovplyvňuje polohu minima, preto ju pri výpočte optimálneho vektora $\hat{\boldsymbol{\beta}}$ netreba uvažovať. Pokiaľ chceme kontrolovať aj veľkosť submodelu, t.j. chceme riešiť úlohu (1.17), stačí do (1.25) doplniť podmienku $||\boldsymbol{\beta}||_0 \leq k$.

1.4 Sweepovanie

Akákoľvek z formulácií úlohy selekcie premenných v LRM z predchádzajúcej časti má exponenciálnu zložitosť, pretože vo všeobecnosti vyžaduje výpočet RSS alebo validačnej chyby pre 2^p rôznych submodelov plného modelu s p vysvetľujúcimi premennými. Ani pri použití rôznych trikov nie je možné exponenciálny charakter výpočtovej zložitosti potlačiť, preto pre zhruba $p > 100$ nie sme schopní porovnať všetky možné submodely a garantovať tak optimalitu výsledku. Musíme sa uspokojiť s tým, že nami získaný víťazný submodel bude len aproximáciou skutočne optimálneho. Súčasne je potrebné dbať na čo najvyššiu rýchlosť výpočtu účelovej funkcie (RSS, validačnej chyby) pre každý zo submodelov, aby celkový čas výpočtu bol udržateľný v prijateľných medziach. Toto je zabezpečené využitím formalizmu *sweepovania*.

Matematická podstata sweepovania je ukrytá v kombinácii podmienok (1.21) a (1.22). Zavedením nových (skalárnych) premenných u a $\gamma \equiv -1$ možno (1.21) prepísať do tvaru

$$\begin{pmatrix} \mathbf{w} \\ u \end{pmatrix} = \begin{pmatrix} \mathbf{X}^\top \mathbf{X} & \mathbf{X}^\top \mathbf{y} \\ \mathbf{y}^\top \mathbf{X} & \mathbf{y}^\top \mathbf{y} \end{pmatrix} \begin{pmatrix} \boldsymbol{\beta} \\ \gamma \end{pmatrix}. \tag{1.26}$$

Nová rovnica prislúchajúca slackovej premennej u bola doplnená s cieľom vytvoriť na

pravej strane symetrickú pozitívne semidefinitnú $(p+1)\times(p+1)$ maticu

$$\mathbf{A} = {\mathbf{X}_A}^\top \mathbf{X}_A = \begin{pmatrix} \mathbf{X}^\top\mathbf{X} & \mathbf{X}^\top\mathbf{y} \\ \mathbf{y}^\top\mathbf{X} & \mathbf{y}^\top\mathbf{y} \end{pmatrix}, \tag{1.27}$$

kde $\mathbf{X}_A = (\mathbf{X}, \mathbf{y})$ je „rozšírená matica plánu“. Podmienka komplementarity (1.22) je reprezentovaná rozdelením množiny $2p+2$ premenných $\{w_1, \ldots, w_p, u, \beta_1, \ldots, \beta_p, \gamma\}$ na množinu $p+1$ bázických (nezávislých) premenných a ich doplnok - $p+1$ nebázických (závislých) premenných. Hodnoty bázických premenných volíme, nebázické premenné sú nimi jednoznačne určené. V rovnici (1.26) sú bazickými premennými $\{\beta_1, \ldots, \beta_p, \gamma\}$, pretože vektor $\left(\mathbf{w}^\top \; u\right)^\top$ je jednoznačne určený ako obraz vektora $\left(\boldsymbol{\beta}^\top \; \gamma\right)^\top$ v lineárnom zobrazení reprezentovanom maticou $\mathbf{A}$. Premenná γ zostáva vždy v bázickej množine, pričom jej hodnotu vždy volíme $\gamma = -1$. K nej doplnková premenná u je preto vždy nebázická. Z každej dvojice komplementárnych premenných β_i, w_i je vždy práve jedna bázická, pričom jej hodnotu volíme ako 0. Vychádzajúc z diskusie v predchádzajúcej časti to znamená, že hodnoty nebázických premenných β_i sú LS odhadmi regresných koeficientov pre submodel daný indexovou množinou $\{i|w_i = 0\}$.

Rovnica (1.26) reprezentuje submodel $\{i|w_i = 0\} = \emptyset$, teda model bez premenných. Netriviálny submodel vytvoríme presunutím niektorých z premenných β_i na ľavú stranu rovnice a im komplementárnych w_i na pravú stranu. Označme indexovú množinu presúvaných premenných ako α a doplnkovú množinu $\overline{\alpha}$. Pre zjednodušenie zápisu, bez ujmy na všeobecnosti, nech $\alpha = \{1, \ldots, k\}$, $\overline{\alpha} = \{k+1, \ldots, p+1\}$ sú ucelené bloky po sebe nasledujúcich indexov pre nejaké dané k a súčasne nech $\mathbf{A}_{\alpha,\alpha}$ je regulárna matica. Sústava (1.26) tak prejde do tvaru

$$\begin{aligned} \mathbf{w}_\alpha &= \mathbf{A}_{\alpha,\alpha}\boldsymbol{\beta}_\alpha + \mathbf{A}_{\alpha,\overline{\alpha}} \begin{pmatrix} \boldsymbol{\beta}_{\overline{\alpha}} \\ \gamma \end{pmatrix}, \\ \begin{pmatrix} \mathbf{w}_{\overline{\alpha}} \\ u \end{pmatrix} &= \mathbf{A}_{\overline{\alpha},\alpha}\boldsymbol{\beta}_\alpha + \mathbf{A}_{\overline{\alpha},\overline{\alpha}} \begin{pmatrix} \boldsymbol{\beta}_{\overline{\alpha}} \\ \gamma \end{pmatrix}. \end{aligned} \tag{1.28}$$

Vďaka invertovateľnosti $\mathbf{A}_{\alpha,\alpha}$ možno z prvej rovnosti vyjadriť $\boldsymbol{\beta}_\alpha$ ako funkciu $\mathbf{w}_\alpha, \boldsymbol{\beta}_{\overline{\alpha}}, \gamma$. Dosadením výsledku za $\boldsymbol{\beta}_\alpha$ do druhej rovnice a po následných úpravách získame sústavu

$$\begin{pmatrix} \boldsymbol{\beta}_\alpha \\ \mathbf{w}_{\overline{\alpha}} \\ u \end{pmatrix} = \begin{pmatrix} \left(\mathbf{A}_{\alpha,\alpha}\right)^{-1} & -\left(\mathbf{A}_{\alpha,\alpha}\right)^{-1}\mathbf{A}_{\alpha,\overline{\alpha}} \\ \mathbf{A}_{\overline{\alpha},\alpha}\left(\mathbf{A}_{\alpha,\alpha}\right)^{-1} & \mathbf{A}_{\overline{\alpha},\overline{\alpha}} - \mathbf{A}_{\overline{\alpha},\alpha}\left(\mathbf{A}_{\alpha,\alpha}\right)^{-1}\mathbf{A}_{\alpha,\overline{\alpha}} \end{pmatrix} \begin{pmatrix} \mathbf{w}_\alpha \\ \boldsymbol{\beta}_{\overline{\alpha}} \\ \gamma \end{pmatrix}, \tag{1.29}$$

ktorá pri voľbe bázických premenných

$$\mathbf{w}_\alpha = \mathbf{0}, \quad \boldsymbol{\beta}_{\overline{\alpha}} = \mathbf{0}, \quad \gamma = -1 \tag{1.30}$$

reprezentuje submodel $S = \{i | w_i = 0\} = \alpha$, a preto musí na ľavej strane platiť $\boldsymbol{\beta}_\alpha = \hat{\boldsymbol{\beta}}_S$. Pre doplnkové indexové množiny platí $\overline{\alpha} = \overline{S} \cup \{p+1\}$.

Matica vytvorená na pravej strane (1.29) je v oblasti lineárnej algebry známa ako *principal pivot transform (PPT)* [25, 26, 3, 24] matice $\mathbf{A}$ vzhľadom na indexovú množinu α a označuje sa $\mathrm{ppt}(\mathbf{A}, \alpha)$, t. j.

$$\mathrm{ppt}(\mathbf{A}, \alpha) \equiv \begin{pmatrix} (\mathbf{A}_{\alpha,\alpha})^{-1} & -(\mathbf{A}_{\alpha,\alpha})^{-1} \mathbf{A}_{\alpha,\overline{\alpha}} \\ \mathbf{A}_{\overline{\alpha},\alpha} (\mathbf{A}_{\alpha,\alpha})^{-1} & \mathbf{A}_{\overline{\alpha},\overline{\alpha}} - \mathbf{A}_{\overline{\alpha},\alpha} (\mathbf{A}_{\alpha,\alpha})^{-1} \mathbf{A}_{\alpha,\overline{\alpha}} \end{pmatrix}. \tag{1.31}$$

PPT má mnoho zaujímavých vlastností, tie podstatné pre potreby tejto práce sú zosumarizované v časti 1.4.1.

Pri voľbe $\mathbf{A}$ v tvare (1.27) má PPT špeciálny význam pri analýze LRM, ako prvýkrát upozornil Efroymson v [4], ktorý k myšlienke ekvivalentnej PPT dospel zrejme nezávisle od Tuckera [25]. Použitý formalizmus bol neskôr označený ako *sweepovanie* [9], čo je termín, ktorý je dodnes v štatistickej literatúre takmer výhradne používaný namiesto PPT. Hlavným výsledkom je, že z matice $\mathrm{ppt}(\mathbf{A}, \alpha)$ možno veľmi jednoducho extrahovať veličiny $(\mathbf{X}_S \mathbf{X}_S)^{-1}, \hat{\boldsymbol{\beta}}_S, RSS_S$ pre model $S = \alpha$ podľa schémy z nasledujúcej vety.

Veta 1.4.1. *Nech* $\mathbf{X}$ *je* $N \times p$ *matica plánu,* $\mathbf{y}$ *je výstupný vektor dĺžky* N *a* $\mathbf{A}$ *je* $(p+1) \times (p+1)$ *matica daná vzťahom* (1.27)*. Potom platí:*

$$\mathrm{ppt}(\mathbf{A}, S)_{S\cup\{p+1\},S\cup\{p+1\}} = \begin{pmatrix} \left(\mathbf{X}_S^\top \mathbf{X}_S\right)^{-1} & -\hat{\boldsymbol{\beta}}_S \\ \hat{\boldsymbol{\beta}}_S^\top & RSS_S \end{pmatrix}, \tag{1.32}$$

kde $\mathrm{ppt}(\mathbf{A}, S)_{S\cup\{p+1\},S\cup\{p+1\}}$ *je hlavná podmatica matice* $\mathrm{ppt}(\mathbf{A}, S)$ *určená indexovou množinou submodelu* S *a indexom* $p+1$ *výstupnej premennej* y*.*

Dôkaz:

Na základe (1.31) ihneď máme

$$\mathrm{ppt}(\mathbf{A}, S)_{S,S} = (\mathbf{A}_{S,S})^{-1} = \left(\left[\mathbf{X}^\top \mathbf{X}\right]_{S,S}\right)^{-1} = \left(\mathbf{X}_S^\top \mathbf{X}_S\right)^{-1}, \tag{1.33}$$

kde druhá rovnosť platí vďaka tomu, že index $p+1$ zodpovedajúci premennej y sa nikdy nenachádza v S. Ďalej, využijúc (1.5), dostávame

$$\begin{aligned} \mathrm{ppt}(\mathbf{A}, S)_{p+1,p+1} &= \mathbf{A}_{p+1,p+1} - \mathbf{A}_{p+1,S} \left(\mathbf{A}_{S,S}\right)^{-1} \mathbf{A}_{S,p+1} \\ &= \mathbf{y}^\top \mathbf{y} - \mathbf{y}^\top \mathbf{X}_S \left(\mathbf{X}_S^\top \mathbf{X}_S\right)^{-1} \mathbf{X}_S^\top \mathbf{y} \\ &= \mathbf{y}^\top \left[\mathbf{I} - \mathbf{X}_S \left(\mathbf{X}_S^\top \mathbf{X}_S\right)^{-1} \mathbf{X}_S^\top\right] \mathbf{y} = RSS_S \end{aligned} \tag{1.34}$$

a pomocou (1.4)

$$\begin{aligned} \mathrm{ppt}(\mathbf{A}, S)_{S,p+1} = -\mathrm{ppt}(\mathbf{A}, S)_{p+1,S}^\top &= -\left(\mathbf{A}_{S,S}\right)^{-1} \mathbf{A}_{S,p+1} \\ &= -\left(\mathbf{X}_S^\top \mathbf{X}_S\right)^{-1} \mathbf{X}_S^\top \mathbf{y} = -\hat{\boldsymbol{\beta}}_S. \end{aligned} \tag{1.35}$$

□

Poznámka: V poslednej časti dôkazu je takisto vhodné si uvedomiť, že po dosadení hodnôt (1.30) za bázické premenné v (1.29) je výsledkom $-\mathrm{ppt}(\mathbf{A}, S)_{\bullet,p+1}$, t. j. -1-násobok posledného stĺpca matice $\mathrm{ppt}(\mathbf{A}, S)$. Porovnaním ľavej a pravej strany dostávame $\boldsymbol{\beta}_S = -\mathrm{ppt}(\mathbf{A}, S)_{S,p+1} = \hat{\boldsymbol{\beta}}_S$ a $u = -RSS_S$. Prvá z uvedených rovností je konzistentná s myšlienkou, na základe ktorej sme vybudovali PPT, t.j. že nebázické zložky β_i zodpovedajú LS-odhadu regresných koeficientov v submodeli $S = \{i | w_i = 0\}$.

Doteraz sme uvažovali len transformácie $\mathrm{ppt}(\mathbf{A}, S)$ matice $\mathbf{A}$ reprezentujúcej prázdny submodel. Takáto operácia zodpovedá vytvoreniu submodelu S pridaním naraz všetkých jeho premenných do prázdneho modelu. V ďalšej časti však ukážeme, že submodel S je maticou $\mathrm{ppt}(\mathbf{A}, S)$ reprezentovaný jednoznačne, nezávisle od spôsobu jeho konštrukcie (postupným) pridávaním/odoberaním premenných reprezentovaného zložením zodpovedajúcich PPT transformácií. Vďaka tomu a vete 1.4.1 sme schopní pri relatívne nízkych výpočtových nákladoch počítať veličiny $\left(\mathbf{X}_S \mathbf{X}_S\right)^{-1}, \hat{\boldsymbol{\beta}}_S, RSS_S$ pre postupnosť submodelov S vytvorenú postupným pridávaním/odoberaním premenných. Túto vlastnosť sweepovacieho formalizmu využijeme pri konštrukcii niektorých algoritmov pre selekciu premenných v LRM, čím sa značne urýchlia.

1.4.1 Algebraické vlastnosti PPT

V tejto časti je prezentovaných niekoľko vlastností PPT vo forme liem a viet, ktoré sú užitočné z hľadiska aplikácie PPT (sweepovania) pri analýze LRM. Vety obsahovo

vychádzajú prioritne z prehľadového článku [24] a čiastočne z krátkej kapitoly o sweepovaní v knihe [15], ich výsledné formulácie a konštrukcie dôkazov sú však v mnohých prípadoch vlastným dielom. Kompletne vlastným výsledkom sú vety 1.4.12 a 1.4.20.

V predchádzajúcej časti bola PPT zavedená len vzorcom (1.31). S cieľom vyhnúť sa prípadným nejasnostiam uvádzame na tomto mieste presnejšiu definíciu.

Definícia 1.4.2. *Nech je daná štvorcová matica* $\mathbf{A} \in \mathbb{C}^{n\times n}$ *a indexová množina* $\alpha \in \langle n\rangle$. *Nech hlavná podmatica* $\mathbf{A}_{\alpha,\alpha}$ *je regulárna. Potom definujeme PPT matice* $\mathbf{A}$ *vzhľadom k indexovej množine* α *vzťahom*

$$\mathrm{ppt}(\mathbf{A},\alpha) \equiv \begin{cases} \begin{pmatrix} (\mathbf{A}_{\alpha,\alpha})^{-1} & -(\mathbf{A}_{\alpha,\alpha})^{-1}\mathbf{A}_{\alpha,\overline{\alpha}} \\ \mathbf{A}_{\overline{\alpha},\alpha}(\mathbf{A}_{\alpha,\alpha})^{-1} & \mathbf{A}_{\overline{\alpha},\overline{\alpha}} - \mathbf{A}_{\overline{\alpha},\alpha}(\mathbf{A}_{\alpha,\alpha})^{-1}\mathbf{A}_{\alpha,\overline{\alpha}} \end{pmatrix} & \alpha \neq \emptyset, \\ \mathbf{A} & \alpha = \emptyset. \end{cases} \tag{1.36}$$

Veta 1.4.3. *(O jednoznačnosti a reverzibilite PPT)*
Nech $\mathbf{A} \in \mathbb{C}^{n\times n}$, $\alpha \in \langle n\rangle$, $\mathbf{A}_{\alpha,\alpha}$ *je regulárna. Pre dané vektory* $\mathbf{x},\mathbf{y} \in \mathbb{C}^n$ *definujme vektory* $\mathbf{u}^{x,y},\mathbf{v}^{x,y} \in \mathbb{C}^n$ *vzťahmi* $\mathbf{u}^{x,y}_{\alpha} = \mathbf{y}_{\alpha}$, $\mathbf{u}^{x,y}_{\overline{\alpha}} = \mathbf{x}_{\overline{\alpha}}$, $\mathbf{v}^{x,y}_{\alpha} = \mathbf{x}_{\alpha}$, $\mathbf{v}^{x,y}_{\overline{\alpha}} = \mathbf{y}_{\overline{\alpha}}$. *Potom* $\mathbf{B} = \mathrm{ppt}(\mathbf{A},\alpha)$ *je jediná matica, pre ktorú platí*

$$\mathbf{y} = \mathbf{A}\mathbf{x} \Leftrightarrow \mathbf{v}^{x,y} = \mathbf{B}\mathbf{u}^{x,y}, \quad \forall \mathbf{x},\mathbf{y} \in \mathbb{C}^n. \tag{1.37}$$

Navyše platí $\mathrm{ppt}(\mathbf{B},\alpha) = \mathbf{A}$, *t.j.*

$$\mathrm{ppt}(\ ,\alpha) \circ \mathrm{ppt}(\ ,\alpha) = \mathrm{id}. \tag{1.38}$$

PPT je teda samoinverzné zobrazenie.

Poznámka: Po vhodnom preindexovaní tak, aby indexová podmnožina $\alpha = \{1,2,\ldots,|\alpha|-1,|\alpha|\}$ tvorila ucelený blok, možno podmienku (1.37) prepísať do prehľadnejšieho tvaru

$$\begin{pmatrix} \mathbf{y}_{\alpha} \\ \mathbf{y}_{\overline{\alpha}} \end{pmatrix} = \mathbf{A}\begin{pmatrix} \mathbf{x}_{\alpha} \\ \mathbf{x}_{\overline{\alpha}} \end{pmatrix} \Leftrightarrow \begin{pmatrix} \mathbf{x}_{\alpha} \\ \mathbf{y}_{\overline{\alpha}} \end{pmatrix} = \mathbf{B}\begin{pmatrix} \mathbf{y}_{\alpha} \\ \mathbf{x}_{\overline{\alpha}} \end{pmatrix}, \quad \forall \mathbf{x},\mathbf{y} \in \mathbb{C}^n. \tag{1.39}$$

Ak dosadíme $\boldsymbol{\beta}$ za $\mathbf{x}$ a $\mathbf{w}$ za $\mathbf{y}$, dostaneme už skôr odvodený súvis medzi vzťahmi (1.26) a (1.29).

Dôkaz: Bez ujmy na všeobecnosti, pre zjednodušenie zápisu preindexujme premenné v súlade s predchádzajúcou poznámkou. Dôkaz oboch implikácií vzťahu ekvivalencie

(1.39) sa ľahko overí dosadením (1.36) za $\mathbf{B}$ a využitím regularity $\mathbf{A}_{\alpha,\alpha}$ (analogicky ako pri odvodení vzťahu (1.29)). Platnosť vzťahu reverzibility (1.38) sa ukáže dvojnásobnou aplikáciou definície PPT (1.36) - najprv na maticu $\mathbf{A}$ a potom na výsledok prvej operácie.

Jednoznačnosť možno dokázať sporom. Nech $\exists \mathbf{B}' \neq \mathbf{B}$ spĺňajúca (1.39). Potom musí platiť

$$\begin{aligned}\mathbf{0} &= \begin{pmatrix}\mathbf{x}_\alpha \\ \mathbf{y}_{\overline{\alpha}}\end{pmatrix} - \begin{pmatrix}\mathbf{x}_\alpha \\ \mathbf{y}_{\overline{\alpha}}\end{pmatrix} = (\mathbf{B}-\mathbf{B}')\begin{pmatrix}\mathbf{y}_\alpha \\ \mathbf{x}_{\overline{\alpha}}\end{pmatrix} = (\mathbf{B}-\mathbf{B}')\begin{pmatrix}\mathbf{A}_{\alpha,\alpha}\mathbf{x}_\alpha + \mathbf{A}_{\alpha,\overline{\alpha}}\mathbf{x}_{\overline{\alpha}} \\ \mathbf{x}_{\overline{\alpha}}\end{pmatrix} = \\ &= (\mathbf{B}-\mathbf{B}')\begin{pmatrix}\mathbf{A}_{\alpha,\alpha} & \mathbf{A}_{\alpha,\overline{\alpha}} \\ \mathbf{0} & \mathbf{I}\end{pmatrix}\mathbf{x}, \quad \forall \mathbf{x} \in \mathbb{C}^n.\end{aligned} \tag{1.40}$$

Diagonálne bloky blokovo hornej trojuholníkovej matice $\mathbf{U} = \begin{pmatrix}\mathbf{A}_{\alpha,\alpha} & \mathbf{A}_{\alpha,\overline{\alpha}} \\ \mathbf{0} & \mathbf{I}\end{pmatrix}$ sú regulárne, preto aj $\mathbf{U}$ ako celok je regulárna a následne $\mathbf{U}\mathbf{x}$, $\mathbf{x} \in \mathbb{C}^n$ generuje celý priestor $\mathbb{C}^n$. Podmienka (1.40) teda môže byť splnená len ak $\mathbf{B}'-\mathbf{B} = \mathbf{0}$, čo je spor s pôvodným predpokladom $\mathbf{B}' \neq \mathbf{B}$. □

Veta 1.4.3 hovorí, že matica $\mathbf{B}$ parciálne invertovanej sústavy

$$\mathbf{v}^{x,y} = \mathbf{B}\mathbf{u}^{x,y}, \quad \forall \mathbf{x}, \mathbf{y} \in \mathbb{C}^n \tag{1.41}$$

je (za predpokladu regularity $\mathbf{A}_{\alpha,\alpha}$) jednoznačne určená voľbou $\mathbf{A}$ a α. Bez ohľadu na to, akým spôsobom vytvoríme maticu $\mathbf{B}$ spĺňajúcu (1.41), vždy máme garantované, že táto matica musí byť $\mathbf{B} = \mathrm{ppt}\,(\mathbf{A}, \alpha)$. Pretože v reálnej úlohe je matica $\mathbf{A}$ vopred zadaná a nemenná, každej množine α je takto jednoznačne priradená matica $\mathrm{ppt}\,(\mathbf{A}, \alpha)$, ktorú preto v takýchto prípadoch budeme ďalej označovať $\mathbf{B}^\alpha$. Ak navyše matica $\mathbf{A}$ je regulárna, sú regulárne aj všetky jej hlavné podmatice $\mathbf{A}_{\alpha,\alpha}$, $\forall \alpha \in \langle n \rangle$ (viď napr. lema 1.4.6). Matice $\mathbf{B}^\alpha$ sú potom korektne definované pre $\forall \alpha \in \langle n \rangle$, pretože existujú $\mathrm{ppt}\,(\mathbf{A}, \alpha)$, $\forall \alpha \in \langle n \rangle$. Nižšie navyše ukážeme (lema 1.4.11), že okrem nepodstatných patologických situácií implikuje regularita matice $\mathbf{X}^\top\mathbf{X}$ injektívnosť zobrazenia $\alpha \mapsto \mathbf{B}^\alpha$ pre $\alpha \in \langle p \rangle$ a $\mathbf{A}$ danú vzťahom (1.27), t.j. $\alpha \neq \alpha' \Rightarrow \mathbf{B}^\alpha \neq \mathbf{B}^{\alpha'}$. Indexovú množinu α (submodel S) možno teda ekvivalentne reprezentovať maticou $\mathbf{B}^\alpha$ ($\mathbf{B}^S$).

Vzťah (1.38) garantuje reverzibilitu procesu PPT. Navyše, invertovanie výsledkov je veľmi jednoduché vďaka tomu, že inverzným zobrazením k $\mathrm{ppt}(\ ,\alpha)$ je to isté zobrazenie. Prvá aplikácia zobrazenia $\mathrm{ppt}(\ ,\alpha)$ reprezentuje pridanie bloku premenných

$\mathbf{x}_\alpha$ do nebázickej množiny (výmenou za $\mathbf{y}_\alpha$), druhá aplikácia zodpovedá spätnému presunu $\mathbf{x}_\alpha$ do bázickej množiny. Vychádzajúc z interpretácie efektu PPT na maticu $\mathbf{A}$ v tvare (1.27) pri vzťahu (1.29) zodpovedajú tieto presuny pri sweepovaní v LRM pridávaniu/odoberaniu premenných do/z submodelu. Prvou operáciou PPT (prvým sweepom) pridáme premenné do prázdneho modelu a vytvoríme submodel $S = \alpha$, druhou operáciou (identickou s prvou) premenné z S odoberieme, čím sa naspäť vrátime do prázdneho modelu. V pridávaní/odoberaní premenných zrejme leží pôvod názvu „sweepovanie“ (z angl. sweeping), pretože vlastne vmetáme/vymetáme (*sweep in/out*) premenné do/z submodelu.

Lema 1.4.4. *Nech* $\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}$ *sú reálne matice rozmerov* $m \times m, m \times n, n \times m, n \times n$. *Nech* $\mathbf{A}$ *je regulárna matica a* $\mathbf{M} = \begin{pmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{pmatrix}$. *Potom*

$$\det \begin{pmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{pmatrix} = \det(\mathbf{A}) \det(\mathbf{D} - \mathbf{C}\mathbf{A}^{-1}\mathbf{B}). \tag{1.42}$$

Ak navyše tzv. Schurov doplnok $\mathbf{D} - \mathbf{C}\mathbf{A}^{-1}\mathbf{B}$ *je regulárna matica, tak je regulárna aj* $\mathbf{M}$, *pričom platí*

$$\begin{aligned} \begin{pmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{pmatrix}^{-1} &= \begin{pmatrix} \mathbf{I} & -\mathbf{A}^{-1}\mathbf{B} \\ \mathbf{0} & \mathbf{I} \end{pmatrix} \begin{pmatrix} \mathbf{A}^{-1} & \mathbf{0} \\ \mathbf{0} & (\mathbf{D} - \mathbf{C}\mathbf{A}^{-1}\mathbf{B})^{-1} \end{pmatrix} \begin{pmatrix} \mathbf{I} & \mathbf{0} \\ -\mathbf{C}\mathbf{A}^{-1} & \mathbf{I} \end{pmatrix} \\ &= \begin{pmatrix} \mathbf{A}^{-1} + \mathbf{A}^{-1}\mathbf{B}(\mathbf{D} - \mathbf{C}\mathbf{A}^{-1}\mathbf{B})^{-1}\mathbf{C}\mathbf{A}^{-1} & -\mathbf{A}^{-1}\mathbf{B}(\mathbf{D} - \mathbf{C}\mathbf{A}^{-1}\mathbf{B})^{-1} \\ -(\mathbf{D} - \mathbf{C}\mathbf{A}^{-1}\mathbf{B})^{-1}\mathbf{C}\mathbf{A}^{-1} & (\mathbf{D} - \mathbf{C}\mathbf{A}^{-1}\mathbf{B})^{-1} \end{pmatrix}. \end{aligned} \tag{1.43}$$

Ak navyše matica $\mathbf{M}$ *je symetrická a pozitívne definitná, tak matice* $\mathbf{A}$ *a* $\mathbf{D} - \mathbf{C}\mathbf{A}^{-1}\mathbf{B}$ *sú tiež symetrické a pozitívne definitné, a teda podmienka ich regularity je splnená.*

Dôkaz: Vďaka regularite matice $\mathbf{A}$ môžeme definovať matice

$$\mathbf{N} = \begin{pmatrix} \mathbf{A} & \mathbf{0} \\ \mathbf{0} & \mathbf{D} - \mathbf{C}\mathbf{A}^{-1}\mathbf{B} \end{pmatrix}, \quad \mathbf{L} = \begin{pmatrix} \mathbf{I} & \mathbf{0} \\ -\mathbf{C}\mathbf{A}^{-1} & \mathbf{I} \end{pmatrix}, \quad \mathbf{U} = \begin{pmatrix} \mathbf{I} & -\mathbf{A}^{-1}\mathbf{B} \\ \mathbf{0} & \mathbf{I} \end{pmatrix}, \tag{1.44}$$

pre ktoré platí identita

$$\mathbf{N} = \mathbf{L}\mathbf{M}\mathbf{U}. \tag{1.45}$$

Determinant blokovo diagonálnej matice je rovný súčinu determinantov diagonálnych blokov[10], preto na ľavej strane dostávame $\det(\mathbf{N}) = \det(\mathbf{A})\det(\mathbf{D} - \mathbf{C}\mathbf{A}^{-1}\mathbf{B})$. Determinant trojuholníkovej matice je súčinom prvkov na jej diagonále, preto $\det(\mathbf{L}) = \det(\mathbf{U}) = 1$ a následne $\det(\mathbf{LMU}) = \det(\mathbf{L})\det(\mathbf{M})\det(\mathbf{U}) = \det(\mathbf{M})$. Porovnaním ľavej a pravej strany máme (1.42). Platnosť vzťahu pre blokovú inverziu (1.43) sa ľahko potvrdí overením rovnosti $\mathbf{M}\mathbf{M}^{-1} = \mathbf{I}$. Alternatívne si z (1.45) vďaka regularite matíc $\mathbf{L}, \mathbf{U}$ ($\det(\mathbf{L}) = \det(\mathbf{U}) = 1 \neq 0$) môžeme vyjadriť $\mathbf{M} = \mathbf{L}^{-1}\mathbf{N}\mathbf{U}^{-1}$, a následne $\mathbf{M}^{-1} = \mathbf{U}\mathbf{N}^{-1}\mathbf{L}$.

Pre symetrickú a pozitívne definitnú $\mathbf{M}$ platí $\mathbf{B} = \mathbf{C}^\top$, preto $\mathbf{L} = \mathbf{U}^\top$. Pretože $\mathbf{L}$ je regulárna, matice $\mathbf{M}$ a $\mathbf{N}$ sú kongruentné, a preto aj $\mathbf{N}$ je pozitívne definitná, čo je práve vtedy, keď sú pozitívne definitné jej diagonálne bloky $\mathbf{A}$ a $\mathbf{D} - \mathbf{C}\mathbf{A}^{-1}\mathbf{C}^\top$. Symetria $\mathbf{A}$ aj jej Schurovho doplnku je zrejmá. □

Veta 1.4.5. *Nech $\mathbf{A} \in \mathbb{C}^{n\times n}$, $\alpha \in \langle n\rangle$ a je daný rozklad $\alpha = \bigcup_{i=1}^{k} \alpha_i$ tak, aby maticová postupnosť*

$$\mathbf{A}^{(0)} = \mathbf{A}, \qquad \mathbf{A}^{(i)} = \mathrm{ppt}\left(\mathbf{A}^{(i-1)}, \alpha_i\right),\ i = 1, \ldots, k, \tag{1.46}$$

bola dobre definovaná, t.j. aby hlavné podmatice $\mathbf{A}^{(i-1)}_{\alpha_i,\alpha_i}$ boli regulárne. Potom je regulárna aj podmatica $\mathbf{A}_{\alpha,\alpha}$ a platí $\mathrm{ppt}\,(\mathbf{A}, \alpha) = \mathbf{A}^{(k)}$. Špeciálne, ak $\alpha = \{1, \ldots, n\}$, tak $\mathbf{A}$ je regulárna a platí $\mathrm{ppt}\,(\mathbf{A}, \alpha) = \mathbf{A}^{-1} = \mathbf{A}^{(k)}$.

Poznámka: Výsledok PPT teda nezávisí na tom, či vykonáme postupnosť PPT napr. pre jednotlivé indexy z α alebo len jednu PPT pre celú indexovú množinu α. Nižšie ukážeme (lema 1.4.6), že podmienka regularity postupnosti matíc $\mathbf{A}^{(i-1)}_{\alpha_i,\alpha_i}$ nielen implikuje regularitu matice $\mathbf{A}_{\alpha,\alpha}$, ale je s ňou navyše ekvivalentná. Dokonca tieto podmienky sú ekvivalentné regularite matíc $\mathbf{A}^{(i)}_{\alpha,\alpha}$, $\forall i \in \{0, \ldots, k\}$. Takto získavame nový nástroj na overovanie regularity matíc - stačí zvoliť rozklad $\bigcup_{i=1}^{k} \alpha_i$ vo forme jednoprvkových množín α_i a sledovať (ne)nulovosť diagonálnych prvkov $a^{(i-1)}_{\alpha_i,\alpha_i}$, $i = 1, \ldots, k$.

Dôkaz: Opäť, bez ujmy na všeobecnosti, preznačme indexovú množinu tak, aby $\alpha_1 = \{1, \ldots, |\alpha_1|\}$, $\alpha_2 = \{|\alpha_1| + 1, \ldots, |\alpha_1| + |\alpha_2|\}, \ldots,$ $\alpha_k = \{|\alpha| - |\alpha_k| + 1, \ldots, |\alpha|\}$

[10] čo možno overiť napr. aplikáciou Laplaceovho vzorca pre výpočet determinantu na oba členy rozkladu $\begin{pmatrix} \mathbf{A} & \mathbf{0} \\ \mathbf{0} & \mathbf{D} \end{pmatrix} = \begin{pmatrix} \mathbf{A} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{pmatrix} \begin{pmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{D} \end{pmatrix}$.

tvorili ucelené bloky indexov, pre prehľadnosť usporiadané za sebou. Potom vďaka disjunktnosti množín $\{\alpha_i\}_{i=1}^k$ máme

$$\begin{aligned}\mathbf{Ax}=\mathbf{y} \Leftrightarrow \mathbf{A}^{(1)}\begin{pmatrix}\mathbf{y}_{\alpha_1}\\ \mathbf{x}_{\overline{\alpha_1}}\end{pmatrix}=\begin{pmatrix}\mathbf{x}_{\alpha_1}\\ \mathbf{y}_{\overline{\alpha_1}}\end{pmatrix} \Leftrightarrow \mathbf{A}^{(2)}\begin{pmatrix}\mathbf{y}_{\alpha_1}\\ \mathbf{y}_{\alpha_2}\\ \mathbf{x}_{\overline{\alpha_1\cup\alpha_2}}\end{pmatrix}=\begin{pmatrix}\mathbf{x}_{\alpha_1}\\ \mathbf{x}_{\alpha_2}\\ \mathbf{y}_{\overline{\alpha_1\cup\alpha_2}}\end{pmatrix} \Leftrightarrow \dots \\ \dots \Leftrightarrow \mathbf{A}^{(k)}\begin{pmatrix}\mathbf{y}_{\alpha}\\ \mathbf{x}_{\overline{\alpha}}\end{pmatrix}=\begin{pmatrix}\mathbf{x}_{\alpha}\\ \mathbf{y}_{\overline{\alpha}}\end{pmatrix}, \quad \forall \mathbf{x},\mathbf{y}\in\mathbb{C}^n.\end{aligned} \tag{1.47}$$

Využijúc jednoznačnosť matice $\mathbf{B}$ spĺňajúcej podmienku (1.37) (veta 1.4.3) dostávame, že musí platiť $\mathbf{A}^{(k)} = \mathrm{ppt}\,(\mathbf{A},\alpha)$. Dôkaz regularity potrebných matíc je obsahom nasledovnej lemy. □

Lema 1.4.6. *Uvažujme značenie ako vo vete 1.4.5. Potom platí*

$$\left\{\mathbf{A}^{(i-1)}_{\alpha_i,\alpha_i}\right\}_{i=1}^k \textit{ sú regulárne} \Leftrightarrow \mathbf{A}_{\alpha,\alpha} \textit{ je regulárna} \Leftrightarrow \left\{\mathbf{A}^{(i)}_{\alpha,\alpha}\right\}_{i=0}^k \textit{ sú regulárne.} \tag{1.48}$$

Dôkaz: Očíslujme jednotlivé tvrdenia v leme v poradí $1,2,3$. Na dôkaz $3\Rightarrow 2$ stačí zvoliť $i=0$, pri $3\Rightarrow 1$ využijeme, že každá hlavná podmatica $(\mathbf{A}^{(i-1)}_{\alpha_i,\alpha_i})$ regulárnej matice $(\mathbf{A}^{(i-1)}_{\alpha,\alpha})$ musí byť regulárna[11].

V $2\Rightarrow 3$ si treba uvedomiť, že regularita $\mathbf{A}_{\alpha,\alpha}\equiv\mathbf{A}^{(0)}_{\alpha,\alpha}$ implikuje regularitu hlavných podmatíc $\mathbf{A}^{(0)}_{\alpha_1,\alpha_1}$ aj $\mathbf{A}^{(0)}_{\alpha\setminus\alpha_1,\alpha\setminus\alpha_1}$. Preto $\mathbf{A}^{(1)}_{\alpha_1,\alpha_1}=\left[\mathbf{A}^{(0)}_{\alpha_1,\alpha_1}\right]^{-1}$ je regulárna matica rovnako ako jej Schurov doplnok (v rámci matice $\mathbf{A}^{(1)}_{\alpha,\alpha}$) $\mathbf{A}^{(1)}_{\alpha\setminus\alpha_1,\alpha\setminus\alpha_1}-\mathbf{A}^{(1)}_{\alpha\setminus\alpha_1,\alpha_1}\left[\mathbf{A}^{(1)}_{\alpha_1,\alpha_1}\right]^{-1}\mathbf{A}^{(1)}_{\alpha_1,\alpha\setminus\alpha_1}$ $=\mathbf{A}^{(0)}_{\alpha\setminus\alpha_1,\alpha\setminus\alpha_1}$. Ná základe (1.42) potom aj $\mathbf{A}^{(1)}_{\alpha,\alpha}$ musí byť regulárna. Indukciou sa regularita dokáže pre zvyšné $\mathbf{A}^{(i)}_{\alpha,\alpha}$.

$1\Rightarrow 2$ dokážeme sporom. Označme $\tau_i=\bigcup_{j=i+1}^k\alpha_j$. Nech $\mathbf{A}^{(0)}_{\alpha,\alpha}$ je singulárna. Potom, keďže podľa predpokladu matica $\mathbf{A}^{(0)}_{\alpha_1,\alpha_1}$ je regulárna, musí byť singulárny jej Schurov doplnok (v rámci matice $\mathbf{A}^{(0)}_{\alpha,\alpha}$), ktorý sa zhoduje s $\mathbf{A}^{(1)}_{\tau_1,\tau_1}$. V rámci neho sa nachádza regulárna hlavná podmatica $\mathbf{A}^{(1)}_{\alpha_2,\alpha_2}$, preto musí byť singulárny jej Schurov doplnok (v rámci matice $\mathbf{A}^{(1)}_{\tau_1,\tau_1}$), ktorý sa zhoduje s $\mathbf{A}^{(2)}_{\tau_2,\tau_2}$. Indukciou dostaneme postupnosť stále sa zmenšujúcich matíc, ktoré majú byť singulárne, až skončíme pri $\mathbf{A}^{(k-1)}_{\tau_{k-1},\tau_{k-1}}=\mathbf{A}^{(k-1)}_{\alpha_k,\alpha_k}$, ktorá je ale podľa predpokladu regulárna, čo je spor. □

Ako špeciálny dôsledok dostávame, že ak počiatočná matica $\mathbf{A}=\mathbf{A}^{(0)}$ je regulárna (zvolíme $\alpha=\{1,\dots,n\}=\Omega$), tak sú regulárne aj všetky matice $\mathbf{B}^\alpha=\mathrm{ppt}\,(\mathbf{A},\alpha)\,,\ \forall\alpha\in$

[11] Vidno to napr. z (1.42) - $\det(\mathbf{A})\neq 0$ je nutnou (no nie postačujúcou) podmienkou pre $\det(\mathbf{M})\neq 0$.

$\langle n \rangle$. Preto musí platiť aj $b_{jj}^{\alpha} \neq 0,\ \forall j \in \Omega,\ \forall \alpha \in \langle n \rangle$, ako dôsledok regularity všetkých hlavných podmatíc (teda aj diagonálnych prvkov) regulárnej matice. Tiež dostávame, že ak sweepujeme postupne jednoprvkové indexové množiny $\alpha_i = i,\ i = 1, \ldots, n$, ekvivalentným kritériom pre regularitu $\mathbf{A}$ je $a_{ii}^{(i-1)} \neq 0,\ \forall i \in \{1, \ldots, n\}$.

Veta 1.4.7. *(o zjednodušovaní a komutatívnosti PPT)*
Nech $\mathbf{A} \in \mathbb{C}^{n \times n}$, $\alpha_1, \alpha_2 \in \langle n \rangle$ *a hlavná podmatica* $\mathbf{A}_{\alpha_1 \cup \alpha_2, \alpha_1 \cup \alpha_2}$ *je regulárna. Potom platí*

$$\operatorname{ppt}\left(\operatorname{ppt}\left(\mathbf{A}, \alpha_2\right), \alpha_1\right) = \operatorname{ppt}\left(\operatorname{ppt}\left(\mathbf{A}, \alpha_1\right), \alpha_2\right) = \operatorname{ppt}(\mathbf{A}, \alpha_1 \triangle \alpha_2), \tag{1.49}$$

kde symbol $\triangle$ *označuje symetrickú diferenciu dvoch množín, t. j.* $\alpha_1 \triangle \alpha_2 = (\alpha_1 \setminus \alpha_2) \cup (\alpha_2 \setminus \alpha_1)$.

Dôkaz: Najprv dokážeme druhú z rovností, prvá bude jej triviálnym dôsledkom. Vytvorme rozklady $\alpha_1 = (\alpha_1 \setminus \alpha_2) \cup (\alpha_1 \cap \alpha_2)$ a $\alpha_2 = (\alpha_2 \setminus \alpha_1) \cup (\alpha_1 \cap \alpha_2)$. Podľa vety 1.4.5 máme

$$\operatorname{ppt}(\ , \alpha_i) = \operatorname{ppt}(\ , \alpha_i \setminus \alpha_j) \circ \operatorname{ppt}(\ , \alpha_i \cap \alpha_j) = \operatorname{ppt}(\ , \alpha_i \cap \alpha_j) \circ \operatorname{ppt}(\ , \alpha_i \setminus \alpha_j),$$

kde $i \neq j \in \{1, 2\}$. Druhá rovnosť musí platiť kvôli neexistencii poradia množín v rozklade, pretože rozklad je množina. Potom máme

$$\begin{aligned}
&\operatorname{ppt}\left(\operatorname{ppt}\left(\mathbf{A}, \alpha_1\right), \alpha_2\right) = \operatorname{ppt}(\ , \alpha_2) \circ \operatorname{ppt}(\ , \alpha_1)\, \mathbf{A} \\
&= \operatorname{ppt}(\ , \alpha_2 \setminus \alpha_1) \circ \operatorname{ppt}(\ , \alpha_1 \cap \alpha_2) \circ \operatorname{ppt}(\ , \alpha_1 \cap \alpha_2) \circ \operatorname{ppt}(\ , \alpha_1 \setminus \alpha_2)\, \mathbf{A} \\
&= \operatorname{ppt}(\ , \alpha_2 \setminus \alpha_1) \circ \operatorname{ppt}(\ , \alpha_1 \setminus \alpha_2)\, \mathbf{A} = \operatorname{ppt}(\ , (\alpha_2 \setminus \alpha_1) \cup (\alpha_1 \setminus \alpha_2))\, \mathbf{A} \\
&= \operatorname{ppt}\left(\mathbf{A}, \alpha_1 \triangle \alpha_2\right).
\end{aligned} \tag{1.50}$$

V tretej rovnosti sme využili (1.38) (reverzibilita PPT) a následne opäť vetu 1.4.5, pretože množiny $\alpha_1 \setminus \alpha_2$ a $\alpha_2 \setminus \alpha_1$ tvoria rozklad množiny $\alpha_1 \triangle \alpha_2$.

Poznamenávame, že všetky použité operácie PPT sú na základe podmienky regularity $\mathbf{A}_{\alpha_1 \cup \alpha_2, \alpha_1 \cup \alpha_2}$ vďaka leme 1.4.6 dobre definované. Prvú rovnosť v tvrdení vety hovoriacu o komutatívnosti poradia operácií PPT zodpovedajúcich všeobecným, nedisjunktným indexovým množinám α_1, α_2 možno zdôvodniť napr. komutatívnosťou operácie symetrickej diferencie. □

Veta 1.4.8. *Nech* $\alpha_1, \ldots, \alpha_k \in \langle n \rangle$ *sú ľubovoľné. Označme*

$$\mathrm{I}_{\alpha_i}^{(j)} = \begin{cases} 0, & ak\ j \notin \alpha_i \\ 1, & ak\ j \in \alpha_i \end{cases}$$

indikátor prítomnosti indexu j *v indexovej množine* α_i *a* $\alpha = \left\{ j \middle| \sum_{i=1}^{k} \mathrm{I}_{\alpha_i}^{(j)} \text{ je nepárne} \right\}$ *množinu tých indexov, ktoré sa nachádzajú v nepárnom počte indexových množín z postupnosti* $\alpha_1, \ldots, \alpha_k$. *Potom platí*

$$\mathrm{ppt}\,(\ ,\alpha_k) \circ \ldots \circ \mathrm{ppt}\,(\ ,\alpha_1)\,\mathbf{A} = \mathrm{ppt}\,(\mathbf{A}, \alpha)\,. \tag{1.51}$$

Dôkaz: Indukciou pomocou vety 1.4.7. □

Veta 1.4.9. *Nech* $\mathbf{A} \in \mathbb{C}^{n\times n}$, $\alpha \in \langle n \rangle$ *a* $\mathbf{A}_{\alpha,\alpha}$ *je regulárna. Potom*

1.

$$\det\left(\mathrm{ppt}\,(\mathbf{A}, \alpha)\right) = \det \mathbf{A}_{\overline{\alpha},\overline{\alpha}} / \det \mathbf{A}_{\alpha,\alpha}. \tag{1.52}$$

2. Ak navyše $\mathbf{A}_{\overline{\alpha},\overline{\alpha}}$ *je regulárna, tak aj* $\mathbf{A}$ *je regulárna a platí*

$$\mathrm{ppt}\,(\mathbf{A}, \alpha) = \mathrm{ppt}\left(\mathbf{A}^{-1}, \overline{\alpha}\right) \tag{1.53}$$

a

$$\left[\mathrm{ppt}\,(\mathbf{A}, \alpha)\right]^{-1} = \mathrm{ppt}\,(\mathbf{A}, \overline{\alpha}) = \mathrm{ppt}\left(\mathbf{A}^{-1}, \alpha\right). \tag{1.54}$$

Dôkaz:

1. Označme $\mathbf{B} = \mathrm{ppt}\,(\mathbf{A}, \alpha)$. Potom

$$\det \mathbf{B}_{\alpha,\alpha} = \det\left[\left(\mathbf{A}_{\alpha,\alpha}\right)^{-1}\right] = 1/\det \mathbf{A}_{\alpha,\alpha}$$

a Schurov doplnok podmatice $\mathbf{B}_{\alpha,\alpha}$ je

$$\mathbf{B}_{\overline{\alpha},\overline{\alpha}} - \mathbf{B}_{\overline{\alpha},\alpha}\left(\mathbf{B}_{\alpha,\alpha}\right)^{-1}\mathbf{B}_{\alpha,\overline{\alpha}} = \mathbf{A}_{\overline{\alpha},\overline{\alpha}}.$$

Dosadením do (1.42) ($\mathbf{M} = \mathbf{B}$, $\mathbf{A} = \mathbf{B}_{\alpha,\alpha}$) dostávame želaný výsledok.

2. Ak matice $\mathbf{A}_{\alpha,\alpha}, \mathbf{A}_{\overline{\alpha},\overline{\alpha}}$ sú regulárne, tak podľa bodu 1. je regulárna aj matica $\mathbf{B}$, a teda jej inverzia $\mathbf{B}^{-1} = \left[\mathrm{ppt}\,(\mathbf{A}, \alpha)\right]^{-1}$ je dobre definovaná. Regularita $\mathbf{B}$ súčasne implikuje regularitu hlavnej podmatice $\mathbf{B}_{\overline{\alpha},\overline{\alpha}}$, teda Schurovho doplnku $\mathbf{A}_{\overline{\alpha},\overline{\alpha}}$ −

$\mathbf{A}_{\overline{\alpha},\alpha}\left(\mathbf{A}_{\alpha,\alpha}\right)^{-1}\mathbf{A}_{\alpha,\overline{\alpha}}$, preto aj matica $\mathbf{A}$ je regulárna a má dobre definovanú inverziu $\mathbf{A}^{-1}$. Vzťah (1.53) možno dokázať priamo, využijúc

$$\mathbf{A}^{-1} = \mathrm{ppt}\left(\mathbf{A}, \alpha \cup \overline{\alpha}\right) = \mathrm{ppt}\left(\ ,\alpha\right) \circ \mathrm{ppt}\left(\ ,\overline{\alpha}\right)\mathbf{A}.$$

Potom

$$\mathrm{ppt}\left(\mathbf{A}^{-1},\overline{\alpha}\right) = \mathrm{ppt}\left(\ ,\overline{\alpha}\right) \circ \mathrm{ppt}\left(\ ,\overline{\alpha}\right) \circ \mathrm{ppt}\left(\ ,\alpha\right)\mathbf{A} = \mathrm{ppt}\left(\mathbf{A},\alpha\right),$$

kde v druhej rovnosti sme využili reverzibilitu operácií PPT (1.38).

Pri vzťahu (1.54) upozorňujeme na to, že sa invertuje matica $\mathrm{ppt}\left(\mathbf{A},\alpha\right)$, nie PPT operácia, ktorá je samoinverzná. Prvú rovnosť možno overiť vynásobením oboch strán maticou $\mathrm{ppt}\left(\mathbf{A},\alpha\right)$ danou blokovou schémou (1.36), čím získame na oboch stranách identitu $\mathbf{I}$. Druhá rovnosť vyplynie z (1.53) po preznačení $\alpha \leftrightarrows \overline{\alpha}$. □

1.4.2 Nesymetrizované sweepovanie

Nesymetrizované sweepovanie je v štatistickej literatúre ekvivalentné označenie pre PPT dané definíciou (1.36). V užšom zmysle slova pod týmto označením budeme rozumieť aplikáciu postupnosti PPT operácií na symetrickú pozitívne definitnú[12] $(p+1)\times(p+1)$ maticu $\mathbf{B}^{\emptyset} = \mathbf{A}$ danú vzťahom (1.27). Tento stav zodpovedá prázdnemu submodelu $S^{(0)} = \emptyset$, ktorý neobsahuje žiadne vysvetľujúce premenné, t.j. $\mathbf{y} \sim \epsilon$. V i-tom kroku postupnosti sa nový submodel $S^{(i)}$ vytvorí pridaním/odstránením časti premenných do/z modelu $S^{(i-1)}$. Tento krok reprezentujeme PPT operáciou

$$\mathbf{B}^{S^{(i)}} = \mathrm{ppt}\left(\mathbf{B}^{S^{(i-1)}}, \alpha_i\right), \tag{1.55}$$

kde $\alpha_i = S^{(i)} \triangle S^{(i-1)}$. Pridávajú sa premenné z množiny $\alpha_i \setminus S^{(i-1)} = S^{(i)} \setminus S^{(i-1)}$, odoberajú sa tie z $\alpha_i \cap S^{(i-1)} = S^{(i-1)} \setminus S^{(i)}$. Veličiny $RSS_{S^{(i)}}$, $\hat{\boldsymbol{\beta}}_{S^{(i)}}$, $\left(\mathbf{X}_{S^{(i)}}^{\top}\mathbf{X}_{S^{(i)}}\right)^{-1}$ sú potom uložené v hlavnej podmatici $\left(\mathbf{B}^{S^{(i)}}\right)_{S^{(i)}\cup\{p+1\},S^{(i)}\cup\{p+1\}}$ v súlade so schémou (1.32) z vety 1.4.1. Detailnejšie štruktúru matice $\mathbf{B}^{S}$ reprezentujúcu submodel S charakterizuje nasledovná lema.

[12] Matica $\mathbf{A}$ je v prípade neplnej hodnosti rozšírenej matice plánu $\mathbf{X}_A$ len semidefinitná. Pre korektné fungovanie algoritmu sweepovania je však potrebná regularita matice $\mathbf{A}$, t.j. jej pozitívna definitnosť. Niekoľko návrhov, ako postupovať v semidefinitnom prípade je uvedených v časti 1.4.4.

Lema 1.4.10. *Nech matica* $\mathbf{B}^S = \mathrm{ppt}\,(\mathbf{A}, S)$ *reprezentuje submodel* S *v súlade s procedúrou nesymetrizovaného sweepovania aplikovanou na maticu* $\mathbf{A} = (\mathbf{X}, \mathbf{y})^\top (\mathbf{X}, \mathbf{y})$. *Nech* $\mathbf{X}_S$ *je matica plánu submodelu* S, $\mathbf{P}_S$ *je projektor na priestor* $\mathcal{M}(\mathbf{X}_\mathcal{S})$ *a* $\mathbf{X}_R$ *je zložená zo zvyšných stĺpcov matice plánu* $\mathbf{X}$ *zodpovedajúcich premenným* $R = \{1, \ldots, p\} \setminus S$, *t.j. platí*[13] $\mathbf{X} = (\mathbf{X}_S, \mathbf{X}_R)$ *a*

$$\mathbf{A} = \begin{pmatrix} \mathbf{X}_S^\top \mathbf{X}_S & \mathbf{X}_S^\top \mathbf{X}_R & \mathbf{X}_S^\top \mathbf{y} \\ \mathbf{X}_R^\top \mathbf{X}_S & \mathbf{X}_R^\top \mathbf{X}_R & \mathbf{X}_R^\top \mathbf{y} \\ \mathbf{y}^\top \mathbf{X}_S & \mathbf{y}^\top \mathbf{X}_R & \mathbf{y}^\top \mathbf{y} \end{pmatrix}. \tag{1.56}$$

Potom pre maticu $\mathbf{B}^S$ *platí*

$$\mathbf{B}^S = \begin{pmatrix} \left(\mathbf{X}_S^\top \mathbf{X}_S\right)^{-1} & -\left(\mathbf{X}_S^\top \mathbf{X}_S\right)^{-1} \mathbf{X}_S^\top \mathbf{X}_R & -\left(\mathbf{X}_S^\top \mathbf{X}_S\right)^{-1} \mathbf{X}_S^\top \mathbf{y} \\ \mathbf{X}_R^\top \mathbf{X}_S \left(\mathbf{X}_S^\top \mathbf{X}_S\right)^{-1} & \mathbf{X}_R^\top (\mathbf{I} - \mathbf{P}_S) \mathbf{X}_R & \mathbf{X}_R^\top (\mathbf{I} - \mathbf{P}_S) \mathbf{y} \\ \mathbf{y}^\top \mathbf{X}_S \left(\mathbf{X}_S^\top \mathbf{X}_S\right)^{-1} & \mathbf{y}^\top (\mathbf{I} - \mathbf{P}_S) \mathbf{X}_R & \mathbf{y}^\top (\mathbf{I} - \mathbf{P}_S) \mathbf{y} \end{pmatrix}. \tag{1.57}$$

Dôkaz: Priamym dosadením blokovej matice (1.56) do definičného vzťahu pre PPT (1.36) podobne ako v dôkaze vety 1.4.1 a využitím $\mathbf{P}_S = \mathbf{X}_S \left(\mathbf{X}_S^\top \mathbf{X}_S\right)^{-1} \mathbf{X}_S^\top$. □

V schéme (1.57) ľahko identifikujeme veličiny $\hat{\boldsymbol{\beta}}_S = \left(\mathbf{X}_S^\top \mathbf{X}_S\right)^{-1} \mathbf{X}_S^\top \mathbf{y}$ a $RSS_S = \mathbf{y}^\top (\mathbf{I} - \mathbf{P}_S) \mathbf{y}$. Vzhľadom na pozitívnu semidefinitnosť hlavných podmatíc $\left(\mathbf{X}_S^\top \mathbf{X}_S\right)^{-1}$, $\mathbf{X}_R^\top (\mathbf{I} - \mathbf{P}_S) \mathbf{X}_R = [(\mathbf{I} - \mathbf{P}_S) \mathbf{X}_R]^\top [(\mathbf{I} - \mathbf{P}_S) \mathbf{X}_R]$ a nezápornosť $\mathbf{y}^\top (\mathbf{I} - \mathbf{P}_S) \mathbf{y} = RSS_S \geq 0$ máme vždy zabezpečené, že diagonála matice $\mathbf{B}^S$ obsahuje iba nezáporné prvky. Za predpokladu regularity $\mathbf{A}$ v kombinácii s lemou 1.4.6 dostávame dokonca

$$\mathbf{B}^S_{j,j} > 0, \quad j \in \{1, \ldots, p+1\}. \tag{1.58}$$

Porovnaním $(p+1)$-vého stĺpca a riadka sme schopní určiť, aký submodel S matica $\mathbf{B}^S$ reprezentuje. Tieto vektory sú totiž takmer identické, líšia sa len znamienkom pri indexoch z množiny S. Nejednoznačnosť by mohla nastať len v prípade $\left(\hat{\boldsymbol{\beta}}_S\right)_j = 0$ pre nejaké $j \in S$. Vtedy však je jedno, či j zaradíme do S alebo nie.

Zo vzťahu (1.57) je tiež zrejmé, že vykonaním operácie nesymetrizovaného sweepovania sa pôvodne symetrická matica $\mathbf{A}$ vo všeobecnosti zmení na nesymetrickú (aj neantisymetrickú) maticu $\mathbf{B}^S$. Na druhú stranu, $\mathbf{B}^S$ nie je úplne všeobecná. Zmenou

[13] Pre zjednodušenie zápisu uvažujeme, že množiny S a R tvoria ucelené bloky indexov.

znamienka pri blokoch $-\left(\mathbf{X}_S^\top \mathbf{X}_S\right)^{-1} \mathbf{X}_S^\top \mathbf{X}_R$ a $-\left(\mathbf{X}_S^\top \mathbf{X}_S\right)^{-1} \mathbf{X}_S^\top \mathbf{y}$ v pravej hornej časti sa $\mathbf{B}^S$ transformuje na symetrickú maticu. Vždy však platí, že podmatica $\mathbf{B}^S_{S,S}$ musí byť symetrická, pretože reprezentuje symetrickú (a pozitívne definitnú) maticu $\left(\mathbf{X}_S^\top \mathbf{X}_S\right)^{-1}$. Strata symetrie je daňou za jednoduchú formu inverznej PPT operácie (1.38). Symetriu matice $\mathbf{B}^S$ možno zachovať pri použití symetrizovaného sweepovania (časť 1.4.3). V tomto prípade však pridávanie a odoberanie premenných je nutné realizovať dvomi odlišne definovanými operáciami.

Pri analýze sweepovacieho algoritmu je vhodné, hoci nie nevyhnutné, vedieť, či existuje jedno-jednoznačný súvis medzi postupnosťou submodelov $\left\{S^{(i)}\right\}_{i=0}^k$ a matíc $\left\{\mathbf{B}^{S^{(i)}}\right\}_{i=0}^k$. V diskusii pod vetou 1.4.3 sme zdôvodnili, že z jednoznačnosti PPT vyplýva, že $S \mapsto \mathbf{B}^S$ je dobre definované zobrazenie. Teraz ukážeme, že v *štandardnej* situácii je toto zobrazenie navyše injektívne, a preto popis postupnosti submodelov pomocou $\left\{S^{(i)}\right\}_{i=0}^k$ alebo $\left\{\mathbf{B}^{S^{(i)}}\right\}_{i=0}^k$ je ekvivalentný.

Lema 1.4.11. *Nech matica* $\mathbf{A}$ *je daná vzťahom* (1.27)*. Nech* $\mathbf{X}^\top\mathbf{X}$ *je regulárna, pričom žiadny zo stĺpcov* $\mathbf{x}_j$ *matice plánu* $\mathbf{X}$ *nie je ortogonálny s vektorom* $\mathbf{y}$*, t. j.*

$$\mathbf{x}_j^\top \mathbf{y} = \left(\mathbf{X}_{\bullet,j}\right)^\top \mathbf{y} \neq 0, \quad j = 1, \ldots, p. \tag{1.59}$$

Potom zobrazenie $S \mapsto \mathbf{B}^S \equiv \mathrm{ppt}\,(\mathbf{A}, S)$*,* $S \in \mathcal{S} = \langle p \rangle$ *je injektívne, t.j.* $S \neq S' \Rightarrow \mathbf{B}^S \neq \mathbf{B}^{S'}$.

Čo znamenajú podmienky lemy? Regularita $\mathbf{X}^\top\mathbf{X}$ je ekvivalentná plnej hodnosti matice plánu $\mathbf{X}$, teda podmienke, že v množine vysvetľujúcich premenných nemáme redundatné premenné, ktoré by sa dali vyjadriť ako lineárna kombinácia iných premenných. Podmienka (1.59) požaduje, aby vstupné premenné x_j mali potenciál byť užitočné pre modelovanie výstupu y, teda ovplyvňovať predikcie. Vzhľadom na to, že premenné sú štandardizované tak, aby ich priemery $\bar{x}_j$, $\bar{y}$ boli nulové, (1.59) vlastne žiada nenulovú koreláciu medzi každým zo vstupov a výstupom. Triviálnym príkladom symetrickej pozitívne definitnej matice $\mathbf{A}$, ktorá spĺňa podmienku regularity, ale kvôli nesplneniu (1.59) nie je zobrazenie $S \mapsto \mathbf{B}^S$ injektívne, je matica identity $\mathbf{I}_{p+1}$. Platí totiž $\mathbf{B}^S = \mathrm{ppt}\,(\mathbf{I}_{p+1}, S) = \mathbf{I}_{p+1},\ \forall S \in \langle p \rangle$, a teda $\mathbf{B}^S = \mathbf{B}^{S'},\ \forall S, S' \in \langle p \rangle$.

Dôkaz: Sporom. Vďaka regularite $\mathbf{X}^\top\mathbf{X}$ sú všetky matice $\mathbf{B}^S$ pre $S \in \langle p \rangle$ dobre definované. Nech existujú 2 submodely $S \neq S'$ tak, aby $\mathbf{B}^S = \mathbf{B}^{S'} \equiv \mathbf{B}$. Bez ujmy

na všeobecnosti môžeme predpokladať $S \cap S' = \emptyset$. Ak by to tak nebolo, využijeme, že z rovnosti $\mathbf{B}^S = \mathbf{B}^{S'}$ musí platiť aj $\text{ppt}\left(\mathbf{B}^S, S \cap S'\right) = \text{ppt}\left(\mathbf{B}^{S'}, S \cap S'\right)$, čo sú matice v dôsledku jednoznačnosti PPT zodpovedajúce maticiam $\mathbf{B}^{S\setminus S'}$ a $\mathbf{B}^{S'\setminus S}$. Pritom $S\setminus S' \neq S'\setminus S$, lebo $S \neq S'$. Celý spor by sa potom ďalej vytvoril pre dvojicu submodelov $S \setminus S'$, $S' \setminus S$ a im zodpovedajúcich matíc $\mathbf{B}^{S\setminus S'}$ a $\mathbf{B}^{S'\setminus S}$.

Porovnaním schém (1.57) pre $\mathbf{B}^S$ a $\mathbf{B}^{S'}$ dostávame, že musí platiť

$$\mathbf{B}_{S\cup S',\overline{S\cup S'}} = \mathbf{0}, \quad \mathbf{B}_{\overline{S\cup S'},S\cup S'} = \mathbf{0}. \tag{1.60}$$

Na overenie tohto tvrdenia si treba rozdeliť maticu $\mathbf{B}$ na bloky podľa rozkladu S, S', $\overline{S \cup S'}$ indexovej množiny Ω. Využijeme, že podľa rozkladu (1.57) pre $\mathbf{B}^S$ je daná dvojica transponovaných blokov súčasťou nejakej symetrickej podmatice, podľa rozkladu (1.57) pre $\mathbf{B}^{S'}$ má byť ale rovnaká dvojica blokov vzájomne antisymetrická. To je ale súčasne možné splniť len vtedy, ak každý blok z uvedenej dvojice je nulová matica $\mathbf{0}$. Napr. z rozkladu pre $\mathbf{B}^S$ dostaneme $\mathbf{B}_{S,\overline{S\cup S'}} = -\left(\mathbf{B}_{\overline{S\cup S'},S}\right)^\top$, ale z rozkladu pre $\mathbf{B}^{S'}$ plynie $\mathbf{B}_{S,\overline{S\cup S'}} = \left(\mathbf{B}_{\overline{S\cup S'},S}\right)^\top$. Preto nutne $\mathbf{B}_{S,\overline{S\cup S'}} = \left(\mathbf{B}_{\overline{S\cup S'},S}\right)^\top = \mathbf{0}$.

Analyzujme ďalej rovnosti (1.60). Napríklad podmienka $\mathbf{B}_{S\cup S',\overline{S\cup S'}} = \mathbf{0}$ je ekvivalentná dvojici podmienok $\mathbf{B}_{S,\overline{S\cup S'}} = \mathbf{0}$ a $\mathbf{B}_{S',\overline{S\cup S'}} = \mathbf{0}$. Matica $\mathbf{B}_{S,\overline{S\cup S'}}$ je súčasťou bloku $\mathbf{B}_{S,\overline{S}} = \mathbf{B}^S_{S,\overline{S}} = -\left(\mathbf{A}_{S,S}\right)^{-1}\mathbf{A}_{S,\overline{S}}$, preto $\mathbf{0} = \mathbf{B}_{S,\overline{S\cup S'}} = -\left(\mathbf{A}_{S,S}\right)^{-1}\mathbf{A}_{S,\overline{S\cup S'}}$. Vynásobením oboch strán maticou $\mathbf{A}_{S,S}$ dostaneme $\mathbf{0} = \mathbf{A}_{S,\overline{S\cup S'}} = \left(\mathbf{A}_{\overline{S\cup S'},S}\right)^\top$ a analogicky aj $\mathbf{0} = \mathbf{A}_{S',\overline{S\cup S'}} = \left(\mathbf{A}_{\overline{S\cup S'},S'}\right)^\top$. Podmienka nulovosti istých blokov matice $\mathbf{B}$ (1.60) tak znamená nulovosť tých istých blokov v pôvodnej matici $\mathbf{A}$

$$\mathbf{A}_{S\cup S',\overline{S\cup S'}} = \mathbf{0}, \quad \mathbf{A}_{\overline{S\cup S'},S\cup S'} = \mathbf{0}. \tag{1.61}$$

Keďže $p+1 \in \overline{S \cup S'}$ $(S, S' \in \langle p \rangle)$, ako dôsledok musí špeciálne platiť aj $\mathbf{A}_{S\cup S',p+1} = \left(\mathbf{X}_{\bullet,S\cup S'}\right)^\top \mathbf{y} = \mathbf{0}$, resp. $\mathbf{x}_j^\top \mathbf{y} = 0$, $j \in S \cup S' \supset \emptyset$. Tým sme ale dospeli k sporu s predpokladom (1.59) o nenulovej korelácii každého z prediktorov x_j s výstupom y. □

Ukázali sme, že sweepovanie (PPT) reprezentuje efektívny nástroj pre výpočet $RSS_{S^{(i)}}$, $\hat{\boldsymbol{\beta}}_{S^{(i)}}$, $\left(\mathbf{X}^\top_{S^{(i)}}\mathbf{X}_{S^{(i)}}\right)^{-1}$ postupnosti submodelov $S^{(i)}$. Pri konštrukcii tejto postupnosti je dôležité vedieť čo najjednoduchšie vypočítať zmenu $RSS_{S'} - RSS_S$ pri prechode z aktuálneho submodelu $S = S^{(i)}$ do kandidátneho S'. Pri tom nám pomôže nasledovná veta.

Veta 1.4.12. *(Vzťah pre priamy výpočet RSS)*
Nech $\mathbf{X}_A = (\mathbf{X}, \mathbf{y}) \in \mathbb{R}^{N\times(p+1)}$ *je rozšírená matica plánu, v ktorej* $(p+1)$*-vý stĺpec zodpovedá výstupnej premennej* $\mathbf{y}$*,* $\mathbf{A} = \mathbf{X}_A^\top \mathbf{X}_A$ *a* $S, S' \in \langle p\rangle$ *sú indexové množiny ľubovoľných dvoch submodelov. Označme* $\Delta S = S \triangle S'$ *množinu tých indexov z* S, S'*, ktoré nemajú spoločné a* $\mathbf{B}^S = \operatorname{ppt}(\mathbf{A}, S)$ *maticu reprezentujúcu submodel* S*. Nech podmatica* $\mathbf{B}^S_{\Delta S,\Delta S}$ *je regulárna. Potom platí*

$$RSS_{S'} = \frac{\det \mathbf{B}^S_{\Delta S\cup\{p+1\},\Delta S\cup\{p+1\}}}{\det \mathbf{B}^S_{\Delta S,\Delta S}}. \tag{1.62}$$

Dôkaz: Množina ΔS zahŕňa práve tie indexy, ktoré treba sweepovať, aby sme z $\mathbf{B}^S$ prešli do $\mathbf{B}^{S'} = \operatorname{ppt}(\mathbf{A}, S') = \operatorname{ppt}\left(\mathbf{B}^S, \Delta S\right)$ reprezentujúcej systém S'. Podľa (1.36) preto platí

$$RSS_{S'} = \mathbf{B}^S_{p+1,p+1} - \mathbf{B}^S_{p+1,\Delta S}\left(\mathbf{B}^S_{\Delta S,\Delta S}\right)^{-1}\mathbf{B}^S_{\Delta S,p+1}. \tag{1.63}$$

Súčasne, využijúc vzťah (1.42) pre determinant blokovej matice, kde zvolíme $\mathbf{A} = \mathbf{B}^S_{\Delta S,\Delta S}$, $\mathbf{B} = \mathbf{B}^S_{\Delta S,p+1}$, $\mathbf{C} = \mathbf{B}^S_{p+1,\Delta S}$, $\mathbf{D} = \mathbf{B}^S_{p+1,p+1}$, dostávame rovnosť

$$\det \mathbf{B}^S_{\Delta S\cup\{p+1\},\Delta S\cup\{p+1\}} = \det \mathbf{B}^S_{\Delta S,\Delta S} \det\left[\mathbf{B}^S_{p+1,p+1} - \mathbf{B}^S_{p+1,\Delta S}\left(\mathbf{B}^S_{\Delta S,\Delta S}\right)^{-1}\mathbf{B}^S_{\Delta S,p+1}\right]. \tag{1.64}$$

Výraz v argumente druhého determinantu na pravej strane je podľa (1.63) rovný $RSS_{S'}$, čo je skalár, a preto $\det RSS_{S'} = RSS_{S'}$. Vydelením rovnosti (1.64) výrazom $\det \mathbf{B}^S_{\Delta S,\Delta S}$, ktorého nenulovosť je zabezpečená podmienkou regularity zo znenia vety, dostávame požadovaný výsledok. □

Dôsledok 1.4.13. *Špeciálne, pre prípad pridania jedinej premennej* $j \in \Omega \setminus S$ *(*$\Omega = \{1,\dots,p\}$*), t. j.* $S' = S \cup \{j\}$*, dostávame*

$$\begin{aligned} RSS_{S'} &= \frac{\det\left(\mathbf{B}^S{}_{\{j,p+1\},\{j,p+1\}}\right)}{\det\left(\mathbf{B}^S{}_{j,j}\right)} = \frac{\mathbf{B}^S_{j,j}\mathbf{B}^S_{p+1,p+1} - \left(\mathbf{B}^S_{j,p+1}\right)^2}{\mathbf{B}^S_{j,j}} \\ &= RSS_S - \frac{\left(\mathbf{B}^S_{j,p+1}\right)^2}{\mathbf{B}^S_{j,j}}, \end{aligned} \tag{1.65}$$

kde sme využili $\mathbf{B}^S_{p+1,p+1} = RSS_S$ *a* $\mathbf{B}^S_{j,p+1} = \mathbf{B}^S_{p+1,j}$ *pre* $j \in \Omega \setminus S$*. Podľa očakávania*[14] $RSS_{S'} \le RSS_S$*, keďže pri nesymetrizovanom sweepovaní* $\mathbf{B}^S_{j,j} > 0$ *pre* $j \in \{1,\dots,p+$

[14] Pretože $S' \supset S$

$1\} \supset \Omega\backslash S$ *(vzťah* (1.58)*). Najnižšie* $RSS_{S'}$ *(najväčší pokles* $RSS_{S'} - RSS_S$*) dosiahneme, ak do* S *pridáme premennú* x_{j^*}*, kde* $j^* = \underset{j\in\Omega\backslash S}{\operatorname{argmax}} \frac{\left(\mathbf{B}^S_{j,p+1}\right)^2}{\mathbf{B}^S_{j,j}}$.

Analogický vzťah ako (1.65) *platí aj v prípade, ak odoberáme jedinú premennú* $j \in S$*, t.j.* $S' = S \setminus \{j\}$*. Keďže však pre* $j \in S$ *platí* $\mathbf{B}^S_{j,p+1} = -\mathbf{B}^S_{p+1,j}$*, dostávame* $RSS_S \leq RSS'_S$*, čo je očakávaný výsledok, lebo teraz* $S \supset S'$*. Bez ujmy na všeobecnosti predpokladajme, že* $S = \{1, \ldots, k\}$[15]*. Potom*

$$\begin{aligned} RSS_{S'} &= \frac{\mathbf{B}^S_{j,j}\mathbf{B}^S_{p+1,p+1} - \mathbf{B}^S_{j,p+1}\mathbf{B}^S_{p+1,j}}{\mathbf{B}^S_{j,j}} = RSS_S - \frac{\left[-\left(\hat{\boldsymbol{\beta}}_S\right)_j\right]\left(\hat{\boldsymbol{\beta}}_S\right)_j}{\left(\left(\mathbf{X}_S^\top\mathbf{X}_S\right)^{-1}\right)_{j,j}} \\ &= RSS_S + \frac{\left(\hat{\boldsymbol{\beta}}_S\right)_j^2}{\left(\left(\mathbf{X}_S^\top\mathbf{X}_S\right)^{-1}\right)_{j,j}}, \end{aligned} \tag{1.66}$$

kde sme využili, že podľa schémy (1.57) *pre* $j \in S$ *platí* $\mathbf{B}^S_{j,j} = \left(\left(\mathbf{X}_S^\top\mathbf{X}_S\right)^{-1}\right)_{j,j} > 0$ *a* $\mathbf{B}^S_{p+1,j} = -\mathbf{B}^S_{j,p+1} = \left(\hat{\boldsymbol{\beta}}_S\right)_j$*. Najnižšie* $RSS_{S'}$ *(najmenší nárast* $RSS_{S'} - RSS_S$*) teda dostaneme vtedy, keď odoberieme premennú s indexom* $j^* = \underset{j\in S}{\operatorname{argmin}} \frac{\left(\hat{\beta}_S\right)_j^2}{\left(\left(\mathbf{X}_S^\top\mathbf{X}_S\right)^{-1}\right)_{j,j}}$.

Poznámka: Vzťah (1.62) má význam najmä z teoretického hľadiska, podobne ako Cramérovo pravidlo pri riešení sústav lineárnych rovníc. V prípade použitia vhodných algoritmov pre výpočet determinantu však môže byť využitý aj pri praktickej realizácii numerických výpočtov.

Na záver tejto časti ešte spomenieme, že (nesymetrizované) sweepovanie sa v štatistickej literatúre zvyčajne zavádza pomocou sweepovacieho operátora $\mathbf{O}_r = \mathrm{ppt}(\ , \{r\})$ ($r \in \Omega$ je ľubovoľné), definovaného len pre jednoprvkové indexové množiny $\alpha = \{r\}$. Pôsobením operátora $\mathbf{O}_r$ sa pridáva/odstraňuje r-tá premenná z/do modelu a matica $\mathbf{A}$ sa transformuje na maticu $\mathbf{A}^* = \mathbf{O}_r\mathbf{A}$ podľa predpisu:

$$\begin{aligned} a^*_{rr} &= \frac{1}{a_{rr}}, & a^*_{ir} &= -\frac{a_{ir}}{a_{rr}}, & i &\neq r, \\ a^*_{rj} &= \frac{a_{rj}}{a_{rr}}, \quad j \neq r, & a^*_{ij} &= a_{ij} - \frac{a_{ir}a_{rj}}{a_{rr}}, & i, j &\neq r. \end{aligned} \tag{1.67}$$

Všimnime si, že transformačný vzťah $a^*_{ij} = a_{ij} - \frac{a_{ir}a_{rj}}{a_{rr}}$ v prípade $i, j \neq r$ je identický s predpisom symetrickej gaussovej eliminačnej procedúry pre pivot a_{rr}. Ekvivalenciu

[15] Tento predpoklad zabezpečí, že indexy $1, \ldots, k$ v maticiach $\mathbf{B}^S$ aj $\left(\mathbf{X}_S^\top\mathbf{X}_S\right)^{-1}$ si vzájomne zodpovedajú.

definície (1.67) s blokovým predpisom (1.36) možno ukázať indukciou. Táto formulácia umožňuje jednoduché overenie vlastností samoinverzie (reverzibility) $\mathbf{O}_r \circ \mathbf{O}_r = \mathrm{id}$ a komutatívnosti $\mathbf{O}_r \circ \mathbf{O}_s = \mathbf{O}_s \circ \mathbf{O}_r,\ \forall r, s \in \Omega$ operácií sweepovania priamym dosadením transformačných vzťahov (1.67).

1.4.3 Symetrizované sweepovanie

Ak nám záleží na tom, aby sa v procese sweepovania zachovala symetrickosť transformovanej matice, môžeme analogické výsledky ako v predchádzajúcej časti získať aj procesom symetrizovaného sweepovania [16, 15]. Musíme však definovať osobitne operáciu pre pridávanie premenných (PPT^+) a osobitne pre ich odoberanie (PPT^-)[16].

Definícia 1.4.14. *Nech je daná štvorcová matica $\mathbf{A} \in \mathbb{C}^{n\times n}$ a indexová množina $\alpha \in \langle n \rangle$. Nech hlavná podmatica $\mathbf{A}_{\alpha,\alpha}$ je regulárna. Potom definujeme transformácie* PPT^+ *a* PPT^- *matice $\mathbf{A}$ vzhľadom k indexovej množine α vzťahmi*

$$\mathrm{ppt}^+(\mathbf{A},\alpha) \equiv \begin{cases} \begin{pmatrix} -\left(\mathbf{A}_{\alpha,\alpha}\right)^{-1} & \left(\mathbf{A}_{\alpha,\alpha}\right)^{-1}\mathbf{A}_{\alpha,\overline{\alpha}} \\ \mathbf{A}_{\overline{\alpha},\alpha}\left(\mathbf{A}_{\alpha,\alpha}\right)^{-1} & \mathbf{A}_{\overline{\alpha},\overline{\alpha}} - \mathbf{A}_{\overline{\alpha},\alpha}\left(\mathbf{A}_{\alpha,\alpha}\right)^{-1}\mathbf{A}_{\alpha,\overline{\alpha}} \end{pmatrix}, & \alpha \neq \emptyset, \\ \mathbf{A}, & \alpha = \emptyset, \end{cases} \tag{1.68}$$

a

$$\mathrm{ppt}^-(\mathbf{A},\alpha) \equiv \begin{cases} \begin{pmatrix} -\left(\mathbf{A}_{\alpha,\alpha}\right)^{-1} & -\left(\mathbf{A}_{\alpha,\alpha}\right)^{-1}\mathbf{A}_{\alpha,\overline{\alpha}} \\ -\mathbf{A}_{\overline{\alpha},\alpha}\left(\mathbf{A}_{\alpha,\alpha}\right)^{-1} & \mathbf{A}_{\overline{\alpha},\overline{\alpha}} - \mathbf{A}_{\overline{\alpha},\alpha}\left(\mathbf{A}_{\alpha,\alpha}\right)^{-1}\mathbf{A}_{\alpha,\overline{\alpha}} \end{pmatrix}, & \alpha \neq \emptyset, \\ \mathbf{A}, & \alpha = \emptyset. \end{cases} \tag{1.69}$$

Zo vzťahov (1.68) a (1.69) je zrejmé, že ak počiatočná matica $\mathbf{A}$ bola symetrická, budú symetrické aj transformované matice $\mathrm{ppt}^+(\mathbf{A},\alpha)$ a $\mathrm{ppt}^-(\mathbf{A},\alpha)$, čo bolo cieľom. Pre symetrizované sweepovanie platí nasledovná analógia vety 1.4.3.

Veta 1.4.15. *(O jednoznačnosti a reverzibilite* PPT^+*/*PPT^-*)*
Nech $\mathbf{A} \in \mathbb{C}^{n\times n}$, $\alpha \in \langle n \rangle$, bez ujmy na všeobecnosti nech $\alpha = \{1, \ldots, |\alpha|\}$ a $\mathbf{A}_{\alpha,\alpha}$ je regulárna. Potom $\mathbf{B}^+ = \mathrm{ppt}^+(\mathbf{A},\alpha)$ a $\mathbf{B}^- = \mathrm{ppt}^-(\mathbf{A},\alpha)$ sú jediné matice, pre ktoré platí

$$\begin{pmatrix} \mathbf{y}_\alpha \\ \mathbf{y}_{\overline{\alpha}} \end{pmatrix} = \mathbf{A}\begin{pmatrix} \mathbf{x}_\alpha \\ \mathbf{x}_{\overline{\alpha}} \end{pmatrix} \Leftrightarrow \begin{pmatrix} -\mathbf{x}_\alpha \\ \mathbf{y}_{\overline{\alpha}} \end{pmatrix} = \mathbf{B}^+\begin{pmatrix} \mathbf{y}_\alpha \\ \mathbf{x}_{\overline{\alpha}} \end{pmatrix}, \quad \forall \mathbf{x}, \mathbf{y} \in \mathbb{C}^n \tag{1.70}$$

[16] Značenie PPT^+, PPT^- nie je štandardné, vzhľadom na vzájomný súvis symetrizovaného a nesymetrizovaného sweepovania sa nám však zdalo byť vhodné.

a

$$\begin{pmatrix}\mathbf{y}_\alpha\\ \mathbf{y}_{\overline{\alpha}}\end{pmatrix} = \mathbf{A}\begin{pmatrix}\mathbf{x}_\alpha\\ \mathbf{x}_{\overline{\alpha}}\end{pmatrix} \Leftrightarrow \begin{pmatrix}\mathbf{x}_\alpha\\ \mathbf{y}_{\overline{\alpha}}\end{pmatrix} = \mathbf{B}^-\begin{pmatrix}-\mathbf{y}_\alpha\\ \mathbf{x}_{\overline{\alpha}}\end{pmatrix}, \quad \forall \mathbf{x},\mathbf{y}\in\mathbb{C}^n. \tag{1.71}$$

Navyše platí $\mathrm{ppt}^+(\mathbf{B}^-,\alpha) = \mathrm{ppt}^-(\mathbf{B}^+,\alpha) = \mathbf{A}$*, t. j.*

$$\mathrm{ppt}^+(\ ,\alpha)\circ\mathrm{ppt}^-(\ ,\alpha) = \mathrm{ppt}^-(\ ,\alpha)\circ\mathrm{ppt}^+(\ ,\alpha) = \mathrm{id} \tag{1.72}$$

a

$$\begin{aligned}\mathrm{ppt}^+(\ ,\alpha) &= \left[\mathrm{ppt}^-(\ ,\alpha)\right]^3 = \mathrm{ppt}^-(\ ,\alpha)\circ\mathrm{ppt}^-(\ ,\alpha)\circ\mathrm{ppt}^-(\ ,\alpha),\\ \mathrm{ppt}^-(\ ,\alpha) &= \left[\mathrm{ppt}^+(\ ,\alpha)\right]^3,\\ \mathrm{id} &= \left[\mathrm{ppt}^+(\ ,\alpha)\right]^4 = \left[\mathrm{ppt}^-(\ ,\alpha)\right]^4.\end{aligned} \tag{1.73}$$

PPT^+ *a* PPT^- *sú teda vzájomne inverzné zobrazenia, pričom jedno možno získať trojnásobným zložením druhého.*

Dôkaz: Platnosť (1.70), (1.71) a jednoznačnosť matíc $\mathbf{B}^+$, $\mathbf{B}^-$ spĺňajúcich tieto podmienky sa dokáže analogicky ako v dôkaze vety 1.4.3. Vzťah reverzibility (1.72) a rovnosti (1.73) možno dokázať napríklad priamym dosadením definičných vzťahov (1.68) a (1.69). Alternatívne si možno uvedomiť, že sústava zodpovedajúca matici $\mathrm{ppt}^+(\ ,\alpha)\circ\mathrm{ppt}^-(\ ,\alpha)\,\mathbf{A}$ má tvar

$$\begin{pmatrix}\mathbf{y}_\alpha\\ \mathbf{y}_{\overline{\alpha}}\end{pmatrix} = \mathrm{ppt}^+(\ ,\alpha)\circ\mathrm{ppt}^-(\ ,\alpha)\,\mathbf{A}\begin{pmatrix}\mathbf{x}_\alpha\\ \mathbf{x}_{\overline{\alpha}}\end{pmatrix}.$$

Z dôvodu jednoznačnosti musí preto platiť $\mathrm{ppt}^+(\ ,\alpha)\circ\mathrm{ppt}^-(\ ,\alpha)\,\mathbf{A} = \mathbf{A}$. Podobne stačí sledovať výmeny vo vektoroch na ľavej a pravej strane sústavy pri dôkaze rovností (1.73). Napríklad trojnásobnou aplikáciou $\mathrm{ppt}^-(\ ,\alpha)$ dostaneme sústavu

$$\begin{pmatrix}-\mathbf{x}_\alpha\\ \mathbf{y}_{\overline{\alpha}}\end{pmatrix} = \left[\mathrm{ppt}^-(\ ,\alpha)\right]^3\mathbf{A}\begin{pmatrix}\mathbf{y}_\alpha\\ \mathbf{x}_{\overline{\alpha}}\end{pmatrix}$$

a využijúc jednoznačnosť ihneď máme $[\mathrm{ppt}^-(\ ,\alpha)]^3 = \mathrm{ppt}^+(\ ,\alpha)$. □

Vyššie uvedená veta platí pre všeobecné $\alpha\in\langle n\rangle$, podmienka $\alpha = \{1,\ldots,|\alpha|\}$ bola pridaná len kvôli sprehľadneniu zápisu. Vzhľadom na formálnu analógiu operácií symetrizovaného a nesymetrizovaného sweepovania uvedieme ďalšie vety bez dôkazu. Možno

ich dokázať podobne ako v časti 1.4.1. Obsah týchto viet je intuitívne zrejmý, ak analyzujeme vplyv PPT^+ a PPT^- operácií na výmeny premenných vo vektoroch v zodpovedajúcej sústave. Ľubovoľnú operáciu $\mathrm{ppt}^\sigma(\ ,\alpha)$, $\sigma \in \{+,-\}$, vieme v dôsledku jednoznačnosti matíc sústav (1.70), (1.71) faktorizovať ako zloženie elementárnych (jednoindexových) transformácií $\mathrm{ppt}^\sigma(\ ,\{r\})$, $r \in \alpha$. Tieto zobrazenia vzájomne komutujú, pretože v sústave vymieňajú len premenné $x_r \leftrightarrows y_r$, s ostatnými nehýbu. Takto sa zdôvodní komutatívnosť $\mathrm{ppt}^\sigma(\ ,\{r\})$ a $\mathrm{ppt}^{\sigma'}(\ ,\{r'\})$ pre $r \neq r'$ a σ,σ' ľubovoľné. Vzťah komutativity (1.72) pre $r = r'$ bol súčasťou vety 1.4.15. Pri dôkaze vety 1.4.17 o ekvivalentných podmienkach regularity je možné opäť vhodne využiť vlastnosti Schurovho doplnku matice.

Veta 1.4.16. *Nech* $\mathbf{A} \in \mathbb{C}^{n\times n}$, $\alpha_1,\alpha_2 \in \langle n\rangle$ *a hlavná podmatica* $\mathbf{A}_{\alpha_1\cup\alpha_2,\alpha_1\cup\alpha_2}$, *je regulárna. Potom platí*

$$\begin{aligned}&\mathrm{ppt}^+(\ ,\alpha_1)\circ\mathrm{ppt}^-(\ ,\alpha_2)\,\mathbf{A} = \mathrm{ppt}^+(\ ,\alpha_1\setminus\alpha_2)\circ\mathrm{ppt}^-(\ ,\alpha_2\setminus\alpha_1)\,\mathbf{A}\\ &= \mathrm{ppt}^-(\ ,\alpha_2\setminus\alpha_1)\circ\mathrm{ppt}^+(\ ,\alpha_1\setminus\alpha_2)\,\mathbf{A} = \mathrm{ppt}^-(\ ,\alpha_2)\circ\mathrm{ppt}^+(\ ,\alpha_1)\,\mathbf{A}.\end{aligned} \tag{1.74}$$

Veta 1.4.17. *Nech* $\mathbf{A} \in \mathbb{C}^{n\times n}$, $\alpha \in \langle n\rangle$, $k \in \mathbb{N}$, $\alpha_i \subset \alpha$, $\forall i \in \{1,\dots,k\}$ *a* $\mathbf{A}_{\alpha,\alpha}$ *je regulárna. Potom maticová postupnosť*

$$\mathbf{A}^{(0)} = \mathbf{A}, \qquad \mathbf{A}^{(i)} = \mathrm{ppt}^{\sigma_i}\left(\mathbf{A}^{(i-1)},\alpha_i\right),\ i = 1,\dots,k, \tag{1.75}$$

kde $\sigma_i \in \{+,-\}$, $\forall i \in \{1,\dots,k\}$, *sú ľubovoľné, je dobre definovaná, t. j. hlavné podmatice* $\mathbf{A}^{(i-1)}_{\alpha_i,\alpha_i}$ *sú regulárne. Navyše platí*

$$\left\{\mathbf{A}^{(i-1)}_{\alpha_i,\alpha_i}\right\}_{i=1}^{k} \text{ sú regulárne} \Leftrightarrow \mathbf{A}_{\alpha,\alpha} \text{ je regulárna} \Leftrightarrow \left\{\mathbf{A}^{(i)}_{\alpha,\alpha}\right\}_{i=0}^{k} \text{ sú regulárne.} \tag{1.76}$$

Špeciálne, ak $k = n$, $\alpha_i = \{i\}$, *a* $\sigma_i = +$, $i = 1,\dots,k$, *platí*

$$\mathbf{A} \text{ je regulárna} \Leftrightarrow a^{(i-1)}_{i,i} \neq 0, \quad \forall i \in \{1,\dots,n\}, \tag{1.77}$$

a

$$\det\mathbf{A} = \prod_{i=1}^{n} a^{(i-1)}_{i,i}, \tag{1.78}$$

$$\mathbf{A}^{(n)} = \mathrm{ppt}^+(\mathbf{A},\Omega) = -\mathbf{A}^{-1}, \tag{1.79}$$

kde $\Omega = \{1,\dots,n\}$.

Veta 1.4.18. *Nech* $\mathbf{A} \in \mathbb{C}^{n\times n}$, $\alpha \in \langle n\rangle$ *a* $\mathbf{A}$ *je regulárna. Potom platí*

$$\mathrm{ppt}^{\sigma}\left(\mathbf{A},\alpha\right) = -\mathrm{ppt}^{\sigma'}\left(\mathbf{A}^{-1},\overline{\alpha}\right), \tag{1.80}$$

$$\left[\mathrm{ppt}^{\sigma}\left(\mathbf{A},\alpha\right)\right]^{-1} = -\mathrm{ppt}^{\sigma'}\left(\mathbf{A},\overline{\alpha}\right) = \mathrm{ppt}^{\sigma}\left(\mathbf{A}^{-1},\alpha\right), \tag{1.81}$$

kde $\sigma \neq \sigma' \in \{+,-\}$.

Analyzujme teraz situáciu, keď sú operácie PPT^{+} a PPT^{-} postupne aplikované na symetrickú pozitívne definitnú $(p+1)\times(p+1)$ maticu $\mathbf{B}^{\emptyset} = \mathbf{A}$ danú vzťahom (1.27), reprezentujúcu prázdny submodel $S^{(0)} = \emptyset$. V i-tom kroku postupnosti sa nový submodel $S^{(i)}$ vytvorí pridaním/odstránením časti premenných do/z modelu $S^{(i-1)}$. Tento krok reprezentujeme zložením PPT^{+} a PPT^{-} operácií

$$\mathbf{B}^{S^{(i)}} = \mathrm{ppt}^{+}(\ ,\alpha_i^{+}) \circ \mathrm{ppt}^{-}(\ ,\alpha_i^{-})\mathbf{B}^{S^{(i-1)}}, \tag{1.82}$$

kde $\alpha_i^{+} = S^{(i)} \setminus S^{(i-1)}$ je množina pridávaných a $\alpha_i^{-} = S^{(i-1)} \setminus S^{(i)}$ množina odoberaných premenných.

Vyššie uvedenú postupnosť PPT^{+} a PPT^{-} operácií označíme ako prípustnú, ak v nej operácia odobratia premennej nasleduje vždy až po pridaní danej premennej a nikde sa nevyskytujú dve operácie pridania alebo odobratia rovnakej premennej za sebou. Vďaka vete 1.4.16 týmto dosiahneme, že v každom iteračnom kroku $i \in \{0,\ldots,k\}$ možno transformovanú maticu zapísať v tvare $\mathbf{B}^{S^{(i)}} = \mathrm{ppt}^{+}(\mathbf{A},S^{(i)})$. Vzhľadom na komutativitu PPT^{+} a PPT^{-} operácií by sme sa k rovnakej finálnej matici $\mathrm{ppt}^{+}(\mathbf{A},S^{(k)})$ dopracovali pri ľubovoľnom preusporiadaní postupnosti týchto operácií, avšak transformované matice v jednotlivých medzikrokoch by nemuseli reprezentovať žiaden z množiny submodelov $\mathcal{S} = \langle p\rangle$.

Pozitívna definitnosť a teda aj regularita matice $\mathbf{A}$ vďaka vete 1.4.17 zabezpečuje, že ľubovoľná transformácia $\mathrm{ppt}^{\sigma}(\ ,\alpha)$ je v každom iteračnom kroku dobre definovaná. Transformované matice $\mathbf{B}^{S^{(i)}}$ sú vďaka symetrickosti pôvodnej matice $\mathbf{A}$ tiež symetrické. Štruktúru matíc $\mathbf{B}^{S^{(i)}} = \mathrm{ppt}^{+}(\mathbf{A},S^{(i)})$ vytvorených procedúrou symetrizovaného sweepovania popisuje nasledovná lema.

Lema 1.4.19. *Nech matica* $\mathbf{B}^{S} = \mathrm{ppt}^{+}\left(\mathbf{A},S\right)$ *reprezentuje submodel* S *v súlade s procedúrou symetrizovaného sweepovania aplikovanou na maticu* $\mathbf{A} = \left(\mathbf{X},\mathbf{y}\right)^{\top}\left(\mathbf{X},\mathbf{y}\right)$.

Uvažujme ďalšie značenie ako v leme 1.4.10. Potom pre maticu $\mathbf{B}^S$ *platí*

$$\mathbf{B}^S = \begin{pmatrix} -\left(\mathbf{X}_S^\top \mathbf{X}_S\right)^{-1} & \left(\mathbf{X}_S^\top \mathbf{X}_S\right)^{-1} \mathbf{X}_S^\top \mathbf{X}_R & \left(\mathbf{X}_S^\top \mathbf{X}_S\right)^{-1} \mathbf{X}_S^\top \mathbf{y} \\ \mathbf{X}_R^\top \mathbf{X}_S \left(\mathbf{X}_S^\top \mathbf{X}_S\right)^{-1} & \mathbf{X}_R^\top \left(\mathbf{I} - \mathbf{P}_S\right) \mathbf{X}_R & \mathbf{X}_R^\top \left(\mathbf{I} - \mathbf{P}_S\right) \mathbf{y} \\ \mathbf{y}^\top \mathbf{X}_S \left(\mathbf{X}_S^\top \mathbf{X}_S\right)^{-1} & \mathbf{y}^\top \left(\mathbf{I} - \mathbf{P}_S\right) \mathbf{X}_R & \mathbf{y}^\top \left(\mathbf{I} - \mathbf{P}_S\right) \mathbf{y} \end{pmatrix}. \tag{1.83}$$

Dôkaz: Priamym dosadením blokového zápisu (1.56) matice $\mathbf{A}$ do definičného vzťahu pre PPT$^+$ (1.68) a využitím $\mathbf{P}_S = \mathbf{X}_S \left(\mathbf{X}_S^\top \mathbf{X}_S\right)^{-1} \mathbf{X}_S^\top$. □

Veličiny RSS_S, $\hat{\boldsymbol{\beta}}_S$, $\left(\mathbf{X}_S^\top \mathbf{X}_S\right)^{-1}$ sú teda uložené v hlavnej podmatici $\left(\mathbf{B}^S\right)_{S\cup\{p+1\},S\cup\{p+1\}}$ $(p+1) \times (p+1)$ matice $\mathbf{B}^S$ podľa schémy[17]

$$\left(\mathbf{B}^S\right)_{S\cup\{p+1\},S\cup\{p+1\}} = \begin{pmatrix} -\left(\mathbf{X}_S^\top \mathbf{X}_S\right)^{-1} & \hat{\beta}_S \\ \hat{\beta}_S^\top & RSS_S \end{pmatrix}. \tag{1.84}$$

Na základe znamienok diagonálnych prvkov matice $\mathbf{B}^S$ vieme jednoznačne určiť, aký model S táto matica reprezentuje. Vychádzajúc z pozitívne definitnej počiatočnej matice $\mathbf{A}$ totiž musí platiť $b_{jj}^S < 0,\ j \in S$ a $b_{jj}^S > 0,\ j \in \Omega \setminus S$. Dôvodom je negatívna definitnosť $-\left(\mathbf{X}_S^\top \mathbf{X}_S\right)^{-1}$ a pozitívna semidefinitnosť $\mathbf{X}_R^\top \left(\mathbf{I} - \mathbf{P}_S\right) \mathbf{X}_R$ (nachádzajúcich sa na diagonále blokovej schémy (1.83)) v kombinácii s vetou 1.4.17, ktorá zabezpečuje $b_{jj}^S \neq 0,\ j \in \Omega$, v prípade regulárnej (t. j. pozitívne definitnej, nie semidefinitnej) počiatočnej matice $\mathbf{A}$.

Aj pre symetrizované sweepovanie možno odvodiť vzťah pre priamy výpočet RSS vo forme podielu determinantov.

Veta 1.4.20. *(Vzťah pre priamy výpočet RSS)*
Nech $\mathbf{X}_A = (\mathbf{X}, \mathbf{y}) \in \mathbb{R}^{N\times(p+1)}$ *je rozšírená matica plánu, v ktorej* $(p+1)$*-vý stĺpec zodpovedá výstupnej premennej* $\mathbf{y}$, $\mathbf{A} = \mathbf{X}_A^\top \mathbf{X}_A$ *a* $S, S' \in \mathcal{S} = \langle p \rangle$ *sú indexové množiny ľubovoľných dvoch submodelov. Označme* $\Delta S = S \triangle S'$ *množinu tých indexov z* S, S'*, ktoré nemajú spoločné a* $\mathbf{B}^S = \mathrm{ppt}^+(\mathbf{A}, S)$ *maticu reprezentujúcu systém* S*. Nech podmatica* $\mathbf{B}_{\Delta S,\Delta S}^S$ *je regulárna. Potom platí*

$$RSS_{S'} = \frac{\det \mathbf{B}_{\Delta S\cup\{p+1\},\Delta S\cup\{p+1\}}^S}{\det \mathbf{B}_{\Delta S,\Delta S}^S}. \tag{1.85}$$

[17]Ekvivalentne by sme vo formalizme symetrizovaného sweepovania mohli pridávanie premenných reprezentovať operáciou PPT$^-$ a odoberanie operáciou PPT$^+$. Potom by sa v schéme (1.84) zamenila $\hat{\boldsymbol{\beta}}_S$ za $-\hat{\boldsymbol{\beta}}_S$.

Dôkaz: Rozdeľme množinu $\triangle S$ na dve disjunktné podmnožiny - $\alpha_0 = S' \setminus S$ (pridávané premenné) a $\alpha_1 = S \setminus S'$ (odoberané premenné). Vďaka regularite matice $\mathbf{B}^S_{\Delta S,\Delta S}$ sú (veta 1.4.17) dobre definované transformácie $\mathrm{ppt}^+\left(\mathbf{B}^S, \alpha_0\right) = \mathrm{ppt}^+\left(\mathbf{A}, S \cup \alpha_0\right) = \mathbf{B}^{S'\cup S}$ a $\mathrm{ppt}^-\left(\mathbf{B}^{S'\cup S}, \alpha_1\right) = \mathrm{ppt}^+\left(\mathbf{A}, [S' \cup S] \setminus \alpha_1\right) = \mathbf{B}^{S'}$. Platnosť identít týchto transformovaných matíc s maticami $\mathbf{B}^{S'\cup S}$ a $\mathbf{B}^{S'}$ je dôsledkom vety 1.4.16. Platí teda

$$\mathbf{B}^S \xrightarrow{\mathrm{ppt}^+(\ ,\alpha_0)} \mathbf{B}^{S'\cup S} \xrightarrow{\mathrm{ppt}^-(\ ,\alpha_1)} \mathbf{B}^{S'}. \tag{1.86}$$

Na dôkaz vety vyjadríme $RSS_{S'} = \mathbf{B}^{S'}_{p+1,p+1}$ spätne najprv pomocou blokov matice $\mathbf{B}^{S'\cup S}$ a následne $\mathbf{B}^S$, v súlade so schémou (1.86).

Pretože $p+1 \in \overline{\alpha_1}$, platí $\mathbf{B}^{S'}_{p+1,p+1} = \mathbf{B}^{S'\cup S}_{p+1,p+1} - \mathbf{B}^{S'\cup S}_{p+1,\alpha_1}\left(\mathbf{B}^{S'\cup S}_{\alpha_1,\alpha_1}\right)^{-1}\mathbf{B}^{S'\cup S}_{\alpha_1,p+1}$, čo je Schurov doplnok matice $\mathbf{B}^{S'\cup S}_{\alpha_1,\alpha_1}$ v rámci matice $\mathbf{B}^{S'\cup S}_{\alpha_1\cup\{p+1\},\alpha_1\cup\{p+1\}}$. Podľa (1.42) preto platí

$$\mathbf{B}^{S'}_{p+1,p+1} = \det \mathbf{B}^{S'}_{p+1,p+1} = \frac{\det \mathbf{B}^{S'\cup S}_{\alpha_1\cup\{p+1\},\alpha_1\cup\{p+1\}}}{\det \mathbf{B}^{S'\cup S}_{\alpha_1,\alpha_1}}. \tag{1.87}$$

Analogicky, keďže $(\alpha_1 \cup \{p+1\}) \cap \alpha_0 = \emptyset$, platí

$$\begin{aligned}
\mathbf{B}^{S'\cup S}_{\alpha_1,\alpha_1} &= \mathbf{B}^S_{\alpha_1,\alpha_1} - \mathbf{B}^S_{\alpha_1,\alpha_0}\left(\mathbf{B}^S_{\alpha_0,\alpha_0}\right)^{-1}\mathbf{B}^S_{\alpha_0,\alpha_1},\\
\mathbf{B}^{S'\cup S}_{\alpha_1\cup\{p+1\},\alpha_1\cup\{p+1\}} &= \mathbf{B}^S_{\alpha_1\cup\{p+1\},\alpha_1\cup\{p+1\}} - \mathbf{B}^S_{\alpha_1\cup\{p+1\},\alpha_0}\left(\mathbf{B}^S_{\alpha_0,\alpha_0}\right)^{-1}\mathbf{B}^S_{\alpha_0,\alpha_1\cup\{p+1\}},
\end{aligned}$$

čo sú Schurove doplnky matice $\mathbf{B}^S_{\alpha_0,\alpha_0}$ v rámci matíc $\mathbf{B}^S_{\alpha_0\cup\alpha_1,\alpha_0\cup\alpha_1} = \mathbf{B}^S_{\triangle S,\triangle S}$ a $\mathbf{B}^S_{\alpha_0\cup\alpha_1\cup\{p+1\},\alpha_0\cup\alpha_1\cup\{p+1\}} = \mathbf{B}^S_{\triangle S\cup\{p+1\},\triangle S\cup\{p+1\}}$. Opätovným využitím (1.42) dostávame

$$\begin{aligned}
\det \mathbf{B}^{S'\cup S}_{\alpha_1,\alpha_1} &= \frac{\det \mathbf{B}^S_{\triangle S,\triangle S}}{\det \mathbf{B}^S_{\alpha_0,\alpha_0}},\\
\det \mathbf{B}^{S'\cup S}_{\alpha_1\cup\{p+1\},\alpha_1\cup\{p+1\}} &= \frac{\det \mathbf{B}^S_{\triangle S\cup\{p+1\},\triangle S\cup\{p+1\}}}{\det \mathbf{B}^S_{\alpha_0,\alpha_0}},
\end{aligned}$$

čo po dosadení do (1.87) dáva

$$\mathbf{B}^{S'}_{p+1,p+1} = \frac{\frac{\det \mathbf{B}^S_{\triangle S\cup\{p+1\},\triangle S\cup\{p+1\}}}{\det \mathbf{B}^S_{\alpha_0,\alpha_0}}}{\frac{\det \mathbf{B}^S_{\triangle S,\triangle S}}{\det \mathbf{B}^S_{\alpha_0,\alpha_0}}} = \frac{\det \mathbf{B}^S_{\triangle S\cup\{p+1\},\triangle S\cup\{p+1\}}}{\det \mathbf{B}^S_{\triangle S,\triangle S}}. \tag{1.88}$$

To už je hľadaný výsledok, pretože $\mathbf{B}^{S'}_{p+1,p+1} = RSS_{S'}$. Pripomíname, že všetky determinanty vystupujúce v menovateľoch vyššie uvedených zlomkov sú nenulové, nakoľko regularita $\mathbf{B}^S_{\Delta S,\Delta S}$ implikuje regularitu matíc $\mathbf{B}^{S'\cup S}_{\alpha_1,\alpha_1}$ a $\mathbf{B}^S_{\alpha_0,\alpha_0}$. □

Špeciálne pre prípad pridania/odobratia jedinej premennej sa dajú odvodiť vzťahy identické s (1.65) a (1.66) z dôsledku vety 1.4.12. Líši sa iba spôsob odvodenia v prípade

odoberania premennej, kde zmena znamienka pri druhom člene nastane z dôvodu $\mathbf{B}^S_{j,j} = -\left(\left(\mathbf{X}_S^\top \mathbf{X}_S\right)^{-1}\right)_{j,j} < 0$.

1.4.4 Prípad neplnej hodnosti matice plánu X

V predchádzajúcich častiach sme videli, aký podstatný je predpoklad regularity matice $\mathbf{X}^\top\mathbf{X}$. Bez neho nie je možné garantovať uskutočniteľnosť postupnosti transformácií $\mathbf{B}^{S^{(i+1)}} = \text{ppt}\left(\mathbf{B}^{S^{(i)}}, \alpha_i\right)$ v procese nesymetrizovaného sweepovania, resp. $\mathbf{B}^{S^{(i+1)}} = \text{ppt}^-\left(\ , \alpha_i^-\right) \circ \text{ppt}^+\left(\ , \alpha_i^+\right) \mathbf{B}^{S^{(i)}}$ v procese symetrizovaného sweepovania. Vďaka leme 1.4.6 a vete 1.4.17 totiž vieme, že v prípade singulárnej matice $\mathbf{X}^\top\mathbf{X}$ možno v každej matici $\mathbf{B}^{S^{(i)}}$ nájsť aspoň jednu singulárnu hlavnú podmaticu $\mathbf{B}^{S^{(i)}}_{\alpha_i^*,\alpha_i^*}$ pre nejaké $\alpha_i^* \in \langle p \rangle$. Ak potom napr. pri nesymetrizovanom sweepovaní nastane situácia $\alpha_i^* \subseteq \alpha_i$, transformáciu $\text{ppt}\left(\mathbf{B}^{S^{(i)}}, \alpha_i\right)$ nie je možné vykonať kvôli singularite podmatice $\mathbf{B}^{S^{(i)}}_{\alpha_i,\alpha_i}$. Pri praktickej realizácii sa v lepšom prípade výpočet na tomto mieste zastaví, v horšom sa vďaka naakumulovaným numerickým nepresnostiam bude matica $\mathbf{B}^{S^{(i)}}_{\alpha_i,\alpha_i}$ javiť numericky ako regulárna, avšak bude veľmi zle podmienená. Transformácia teda prebehne, avšak prvky novej matice $\mathbf{B}^{S^{(i+1)}}$, ako aj všetkých jej prípadných ďalších transformácií $\mathbf{B}^{S^{(i+2)}}, \mathbf{B}^{S^{(i+3)}}, \ldots$, budú mať pôvod prevažne v rôznych numerických nepresnostiach, a preto sú z hľadiska štatistickej analýzy bezcenné.

Regularita $\mathbf{X}^\top\mathbf{X}$ je ekvivalentná plnej hodnosti matice plánu $\mathbf{X}$. Znamená to, že pri tvorbe modelu nevyužívame prediktory x_j, ktoré by boli lineárne závislé. Toto je zvyčajne splnené, ak hodnoty vstupných premenných pre jednotlivé pozorovania vznikli len meraním. Pri tvorbe LRM však množinu prediktorov zvyčajne dopĺňame aj množstvom umelých doplnkových premenných, vytvorených transformáciou pôvodných. Ľahko nahliadneme, že napr. pridaním novej premennej v tvare diferencie $x_i - x_j$ sa lineárna nezávislosť stĺpcov matice $\mathbf{X}$ poruší. Napriek tomu môžeme chcieť, aby diferencie boli zaradené do množiny premenných, z ktorých budeme selektovať. Napríklad preto, že z hľadiska bias-variance trade-off-u môže byť výhodnejšie mať v modeli diferencie namiesto niektorých pôvodných premenných.

V tejto práci sme na regularizáciu matice $\mathbf{X}^\top\mathbf{X}$, resp. $\mathbf{A} = (\mathbf{X}, \mathbf{y})^\top (\mathbf{X}, \mathbf{y})$, využívali nasledovné úpravy:

1. Vytvor QR-rozklad matice $\mathbf{X}$. Z matice $\mathbf{X}$ odstráň všetky stĺpce, ktoré nie sú

pivotové[18] v matici $\mathbf{R}$ z QR-rozkladu.

2. Vykonaj náhodnú perturbáciu matice $\mathbf{X}$:

$$x_{ij} \mapsto x_{ij} + u_{ij},\ u_{ij} \sim \mathcal{U}(-\kappa, \kappa),\ i \in \{1, \ldots, N\},\ j \in \{1, \ldots, p\}, \tag{1.89}$$

kde $\mathcal{U}(a, b)$ označuje rovnomerné rozdelenie na intervale (a, b) a $\kappa > 0$ je zvolené malé číslo.

3. Rovnomerne zdvihni diagonálu matice $\mathbf{A}$:

$$\mathbf{A} \mapsto \mathbf{A} + \lambda\mathbf{I}, \tag{1.90}$$

kde $\lambda > 0$ je zvolené malé číslo.

Operácie 1-3 možno pre účely regularizácie ľubovoľne kombinovať (pri zachovaní poradia).

V prvom kroku (QR-rozklad) sú ponechané tie stĺpce $\mathbf{x}_j$ z $\mathbf{X}$, ktoré neležia v lineárnom obale (zľava) predchádzajúcich stĺpcov $\mathbf{x}_1, \ldots, \mathbf{x}_{j-1}$. Výsledok tejto hrubej predselekcie teda závisí na usporiadaní stĺpcov v matici $\mathbf{X}$. Navyše, nijako neberie do úvahy koreláciu medzi $\mathbf{x}_j$ a výstupom $\mathbf{y}$. Finálny selekčný algoritmus preto môže byť nútený zvoliť submodel s väčším počtom premenných, než je v skutočnosti potrebné. Napr. ak $y \sim x_1 - x_2$ a QR-rozkladom sa z množiny $\{x_1, x_2, x_1 - x_2\}$ vyberie $\{x_1, x_2\}$, budú vo všeobecnosti na konštrukciu dobrého modelu potrebné obe premenné x_1, x_2, pričom by stačila len jedna diferencia $x_1 - x_2$.

Pripočítaním náhodnej zložky k matici $\mathbf{X}$ dosiahneme lineárnu nezávislosť jej stĺpcov (pre $p < N$), tento dodatočný šum však môže zhoršiť predikčnú chybu výsledného modelu. V treťom kroku sa pripočítaním kladnej diagonály z kladne semidefinitnej matice $\mathbf{A}$ vytvára kladne definitná, pričom jej vlastné hodnoty sa posúvajú ako $\lambda(\mathbf{A}) \mapsto \lambda(\mathbf{A}) + \lambda$, čím vieme výrazne zlepšiť číslo podmienenosti matice $\mathrm{cond}(\mathbf{A}) = \frac{\lambda_{\max}(\mathbf{A})}{\lambda_{\min}(\mathbf{A})}$, ak $\lambda_{\min}(\mathbf{A}) \approx 0$. Vplyv tohto kroku na podmienenosť matíc $\mathbf{B}^{S^{(i)}}$, si vyžaduje ďalšiu analýzu.

[18] Matica $\mathbf{R}$ je horná trojuholníková, pričom jej stĺpce korešpondujú so stĺpcami $\mathbf{X}$. Pivot tu označuje prvý nenulový prvok (zľava) v ľubovoľnom riadku matice $\mathbf{R}$, pivotový stĺpec je taký, v ktorom sa nachádza niektorý z pivotov.

Voliteľné parametre κ, λ nastavujeme pre zlepšenie numerickej stability čo najväčšie, avšak súčasne tak malé, aby relatívna odchýlka napr. $RSS_{S^{(i)}}$ získaných ako prvok $\mathbf{B}^{S^{(i)}}_{p+1,p+1}$ oproti skutočným hodnotám bola prijateľne nízka. Pri použití len prvého kroku (QR-rozklad) nevznikajú dodatočné[19] nepresnosti pri výpočte $RSS_{S^{(i)}}$, avšak množina dostupných submodelov je obmedzená predvýberom premenných na základe pivotizácie v QR-rozklade.

1.5 Gaussova eliminácia v lineárnej regresii

Pri lineárnej regresii sa bežne môžeme stretnúť s Gaussovou elimináciou (GE) pri riešení sústavy normálnych rovníc (1.3). GE však možno použiť aj na efektívny výpočet $RSS_{S^{(i)}}$ postupnosti do seba vnorených submodelov $S^{(i-1)} \subset S^{(i)}$, resp. $S^{(i)} \subset S^{(i-1)}$, bez nutnosti určenia vektora odhadov regresných koeficientov $\hat{\boldsymbol{\beta}}_{S^{(i)}}$, či projekcií $\hat{\mathbf{y}} = \mathbf{P}_{\mathbf{X}_{S^{(i)}}}\mathbf{y} = \mathbf{X}_{S^{(i)}}\hat{\boldsymbol{\beta}}_{S^{(i)}}$ vektora $\mathbf{y}$ do stĺpcového priestoru $\mathcal{M}(\mathbf{X}_{S^{(i)}})$. V prípade potreby sme schopní vektor $\hat{\boldsymbol{\beta}}_{S^{(i)}}$ extrahovať dodatočnou spätnou substitúciou, pre výpočet len $RSS_{S^{(i)}}$ tento krok ale nie je potrebný. Nevýhodou je, že, narozdiel od sweepovania, postupnosť submodelov $S^{(i)}$ môže byť vytváraná výhradne buď len pridávaním premenných alebo len ich odoberaním. Kombináciu pridávania a odoberania premenných, ani rozloženú do viacerých krokov, nie je možné realizovať. Príčinou je, že procedúra GE neumožňuje rekonštruovať predchádzajúce kroky výhradne len z parciálne eliminovanej sústavy rovníc.

Veta 1.5.1. *Nech* $\mathbf{A} = {\mathbf{X}_A}^\top \mathbf{X}_A$ *je* $(p+1) \times (p+1)$ *matica, pričom* $\mathbf{X}_A = (\mathbf{X}, \mathbf{y})$ *je rozšírená matica plánu. Nech matica* $\mathbf{B}$ *vznikla z* $\mathbf{A}$ *elimináciou premenných z indexovej množiny (submodelu)* $S \in \langle p \rangle$ *podľa pravidiel Gaussovej eliminácie bez spätnej substitúcie*[20]*. Potom*

$$RSS_S = \mathbf{B}_{p+1,p+1}. \tag{1.91}$$

Nech navyše $\mathbf{A}$ *je regulárna. Nech matica* $\mathbf{B}'$ *vznikla z* $\mathbf{B}$ *spätnou substitúciou premenných z množiny* S[21]*. Potom*

$$\hat{\boldsymbol{\beta}}_S = \mathbf{B}'_{S,p+1}. \tag{1.92}$$

[19] Myslí sa okrem numerických chýb pri sčítavaní a násobení.

[20] Podmatica $\mathbf{B}_{S,S}$ je teda horná trojuholníková a $\mathbf{B}_{\overline{S},S} = \mathbf{0}$.

[21] Platí teda $\mathbf{B}'_{S,S} = \mathbf{I}$ a $\mathbf{B}'_{\overline{S},S} = \mathbf{0}$.

Nech matica $\mathbf{C}$ *vznikla z* $\mathbf{A}^{-1}$ *elimináciou premenných z indexovej množiny* $S \in \langle p \rangle$ *podľa pravidiel Gaussovej eliminácie bez spätnej substitúcie. Potom*

$$RSS_{\Omega \setminus S} = 1/\mathbf{C}_{p+1,p+1}, \tag{1.93}$$

kde $\Omega = \{1, \ldots, p\}$.

Dôkaz: Technicky nenáročný, ale trochu dlhší dôkaz pre vzťah (1.91) možno nájsť napr. v [20] na s. 333-335. Pre dôkaz (1.92) si stačí uvedomiť, že GE so spätnou substitúciou na množine $S \in \langle p \rangle$ transformuje blok $\mathbf{A}_{S,p+1} = \mathbf{X}_S^\top \mathbf{y}$ rovnako ako pri riešení sústavy $\mathbf{X}_S^\top \mathbf{X}_S \boldsymbol{\beta} = \mathbf{X}_S^\top \mathbf{y}$, pretože $\mathbf{A}_{S,S} = \mathbf{X}_S^\top \mathbf{X}_S$. Blok $\mathbf{B}'_{S,p+1}$ preto musí obsahovať riešenie tejto sústavy, t. j. $\left(\mathbf{X}_S^\top \mathbf{X}_S\right)^{-1} \mathbf{X}_S^\top \mathbf{y} = \hat{\boldsymbol{\beta}}$. V dôkaze (1.93) je vhodné využiť vzťah pre inverziu blokovej matice (1.43). □

Na základe vzťahov (1.91) a (1.93) možno skonštruovať dve verzie algoritmu. V prvom prípade (vzťah (1.91)) eliminácia ľubovoľne zvolenej množiny stĺpcov S matice $\mathbf{A}$ zodpovedá vytvoreniu submodelu S pridaním zodpovedajúcej skupiny premenných do prázdneho modelu S_{NULL}. Matica $\mathbf{A}$ reprezentuje prázdny submodel S_{NULL} s $RSS_{S_{\text{NULL}}} = \mathbf{y}^\top \mathbf{y}$. V druhom prípade (vzťah (1.93)) sa elimináciou stĺpcov S inverznej matice $\mathbf{A}^{-1}$ odoberajú premenné S z úplného modelu $S_{\text{FULL}} = \{1, \ldots, p\}$, čím získame submodel $\Omega \setminus S$. Matica $\mathbf{A}^{-1}$ reprezentuje úplný model S_{FULL} s $RSS_{S_{\text{FULL}}} = \mathbf{y}^\top \left[\mathbf{I} - \mathbf{X}\left(\mathbf{X}^\top \mathbf{X}\right)^{-1} \mathbf{X}^\top\right] \mathbf{y}$.

GE je výhodná z hľadiska výpočtovej zložitosti, ak nás zaujíma $RSS_{S^{(i)}}$ postupnosti submodelov $S^{(i)}$ generovaných postupným pridávaním[22] alebo odoberaním[23] premenných. RSS nového submodelu ľahko získame po eliminácii stĺpca zodpovedajúceho práve pridávanej/odoberanej premennej. Výpočtová výhoda tohto algoritmu sa stratí, ak by sme chceli pomocou GE počítať RSS_S len pre jediný submodel a nie ich postupnosť. Vtedy je výhodnejšie použiť prístup pomocou QR alebo Choleského rozkladu, čo zvyčajne vedie ku kratšiemu výpočtu s numericky presnejším výsledkom.

[22] Takýto postup zodpovedá tzv. doprednej (forward) selekcii (viď časť 2.2.1).

[23] Táto schéma zodpovedá tzv. spätnej (backward) selekcii (viď časť 2.2.2).

Kapitola 2

Selekčné algoritmy

2.1 Best subset selekcia

Best subset selekcia reprezentuje naivný prístup (hrubú silu), kedy počítame sumu štvorcov odchýlok RSS_S pre každý z 2^p možných submodelov S úplného modelu $S_{\text{FULL}} = \{1, \ldots, p\}$ s p vysvetľujúcimi premennými. V teoreticky ideálnom prípade je výsledkom postupnosť $\{S_k\}_{k=1}^p$ submodelov $S_k = \underset{S \in \mathcal{S}_k}{\operatorname{argmin}}\, RSS_S$ s najnižším RSS_S pri danej veľkosti $|S| = k$ pre každú veľkosť $k = 1, \ldots, p$. Víťazný model sa nakoniec vyberie spomedzi $\{S_k\}_{k=1}^p$ na základe porovnania hodnôt zvoleného informačného kritéria alebo (cross)validačnej chyby submodelov $\{S_k\}_{k=1}^p$. Tento postup je v súlade s usporiadaním submodelov $S \in \mathcal{S}$ podľa AIC/BIC, pretože usporiadanie podľa týchto IC sa v rámci množiny submodelov $\mathcal{S}_k = \{S \in \mathcal{S} : |S| = k\}$ rovnakej veľkosti k redukuje na usporiadanie podľa RSS_S. Pri reálnom výpočte sme schopní vo väčšine prípadov získať len odhady teoreticky optimálnych modelov $\{S_k\}_{k=1}^p$, ktoré budeme označovať $\{S^{(k)}\}_{k=1}^p$

2.1.1 Branch and bound algoritmus

Best subset selekcia priamym prehľadávaním všetkých 2^p submodelov $S \in \mathcal{S}$ je kvôli exponenciálnej zložitosti v rozumnom čase neuskutočniteľná už pre relatívne malé hodnoty p (rádovo desiatky). Výpočtovo efektívnejšiu modifikáciu tohto prístupu repre-

zentuje *branch and bound* algoritmus podľa Furnivala a Wilsona [5][1]. Tento si síce zachováva exponenciálnu zložitosť, no umožňuje urýchliť výpočet rádovo 1000-krát[2].

Redukcia problému na úroveň porovnávania RSS_S v rámci množiny $\mathcal{S}_k$ umožňuje využiť známu nerovnosť

$$RSS_S \leq RSS_{S'} \tag{2.1}$$

medzi RSS modelu S a jeho submodelu $S' \subseteq S$. Pri usporiadaní modelov $S \in \mathcal{S}$ do vhodnej stromovej štruktúry, reflektujúcej vzťah model-submodel, sa totiž možno vyhnúť výpočtom súvisiacim s regresiou pre modely v celej vetve stromu, pokiaľ RSS_S modelu S (veľkosti $|S| = m$) v báze vetvy je väčší než dosiaľ najnižšie hodnoty RSS pre modely veľkostí $1, \ldots, m$. Vzhľadom na to, že v danej vetve sú len modely $S' \subseteq S$, platí (2.1), a teda žiadny zo submodelov S' nemôže mať nižšie RSS než dosiaľ najnižšie pre model rovnakej veľkosti $|S'|$.

Algoritmus

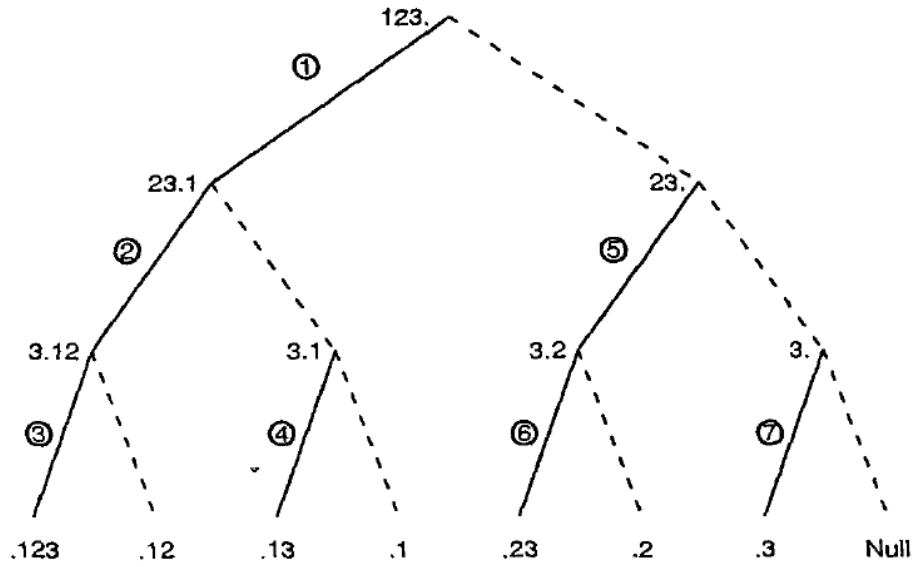
1. Preindexujeme vysvetľujúce premenné x_j podľa hodnôt $f_j = \frac{\left(\hat{\beta}_{S_{\text{FULL}}}\right)_j^2}{\left(\left(\mathbf{X}^\top\mathbf{X}\right)^{-1}\right)_{j,j}}$ tak, aby f_j bola najväčšia pre x_1 a postupne klesala až na minimum pre x_p[3].

2. Vytvoríme 2 stromy - *regresný* a *inverzný*. Regresný strom (obr. 2.1) je tvorený všetkými 2^{p-1} modelmi neobsahujúcimi najmenej významnú premennú x_p. V koreni je prázdny model S_{NULL}, každý syn nejakého vrchola reprezentuje model, ktorý sa od otcovského líši pridaním nejakého prediktora. Počet prediktorov v modeli teda rastie od koreňa smerom k listom stromu. Inverzný strom (obr. 2.2)

[1]Originálny článok je dosť ťažko čitateľný, pedagogicky lepšie je myšlienka algoritmu popísaná napr. v [20], s. 442-446.

[2]Pri výpočtoch na počítači s procesormi P8400, špecifikovanom v kapitole 3, bolo možné pomocou branch and bound algoritmu kompletne prehľadať množinu $\mathcal{S}$ pre $p \approx 50$ približne za 2 hodiny. Podobne dlho trvali výpočty v prípade $p \approx 90$, ak sme sa obmedzili na hľadanie optimálnych submodelov S_k pre $k < 20$.

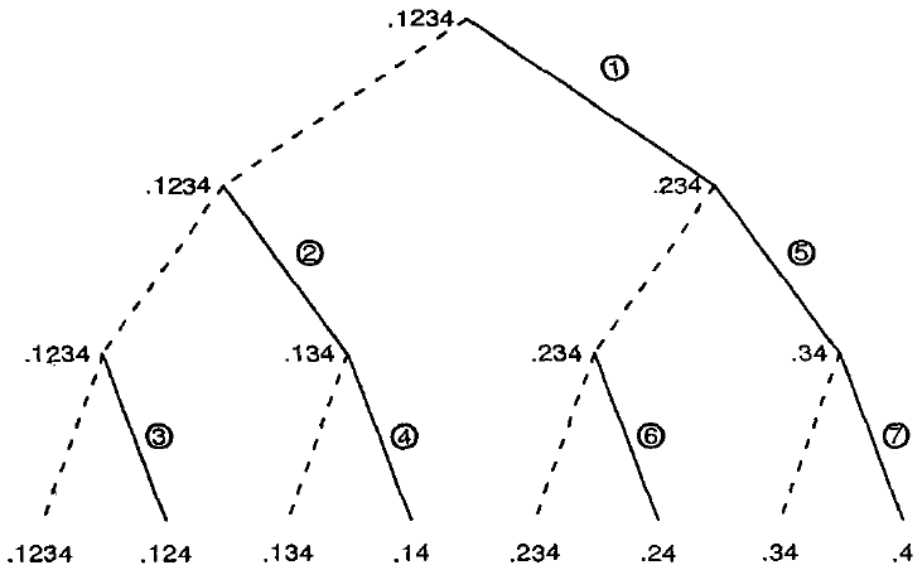
[3]Porovnaním so vzťahom (1.66) máme, že $f_j = RSS_{S_{\text{FULL}}\setminus\{j\}} - RSS_{S_{\text{FULL}}}$ udáva veľkosť nárastu RSS po odobratí premennej x_j z úplného modelu S_{FULL}. Veličina f_j okrem toho súvisí s F-štatistikou $F_j = \frac{f_j/1}{RSS_{S_{\text{FULL}}}/(N-p)}$ pre testovanie hypotézy $\mathcal{H}_0 : \beta_j = 0$ o významnosti premennej x_j v úplnom modeli. Uvedené usporiadanie podľa f_j teda zodpovedá usporiadaniu premenných od najvýznamnejšej (x_1) po najmenej dôležitú (x_p).



Obr. 2.1: Regresný strom pre $p = 4$ (rozšírená schéma). Značenie vrcholov (napr. 23.1): modely vo vetve začínajúcej v tomto vrchole vzniknú pridaním kombinácie premenných s indexami pred čiarkou (x_2, x_3) do modelu s premennými za čiarkou (x_1). Pridanie premennej je znázornené plnou hranou. Prerušovaná hrana znamená nepridanie premennej, spája teda vrcholy zodpovedajúce rovnakému modelu (označenému v dolnom riadku). Zlúčením vrcholov spojených prerušovanými hranami (a odstránením týchto hrán), získame redukovaný regresný strom, kde každý vrchol zodpovedá práve jednému modelu. Indexy v krúžku nad hranami označujú poradie, v akom prechádzame medzi modelmi v redukovanom regresnom strome pri Branch and bound algoritme. Prevzaté z [20], s. 443, obr. 12.5.

je tvorený zvyšnými 2^{p-1} modelmi obsahujúcimi x_p. Modely v inverznom strome sú podľa veľkosti usporiadané opačne, v koreni je úplný model S_{FULL} a smerom k listom počet prediktorov klesá.

3. Modely vo vrcholoch oboch stromov prechádzame pomocou zodpovedajúcej postupnosti operácií sweepovania súčasne v poradí, ktoré je vyznačené zakrúžkovanými číslami na obr. 2.1 a 2.2. Pre prechod medzi dvoma nasledujúcimi vrcholmi stačí vždy odobrať alebo pridať práve jednu premennú, čím sa môže naplno prejaviť výpočtová efektivita formalizmu sweepovania. V regresnom strome vychádzame z matice $\mathbf{A}$ (1.27) reprezentujúcej prázdny model S_{NULL}. V inverznom strome začíname s maticou $\text{ppt}^{+}(\mathbf{A}, S_{\text{FULL}})$ $(\text{ppt}(\mathbf{A}, S_{\text{FULL}}))$, zodpovedajúcou úplnému modelu S_{FULL} vo formalizme (ne)symetrizovaného sweepovania.



Obr. 2.2: Inverzný strom (tiež duálny strom alebo strom ohraničení) pre $p = 4$ (rozšírená schéma). Odobratie (neodobratie) premennej je znázornené plnou (prerušovanou) hranou. Zlúčením vrcholov spojených prerušovanými hranami, získame redukovaný inverzný strom, kde každý vrchol zodpovedá práve jednému modelu. Každý z modelov obsahuje premennú $x_p \equiv x_4$, ako ľahko vidno v dolnom riadku. Indexy v krúžku nad hranami označujú poradie, v akom prechádzame medzi modelmi v redukovanom inverznom strome pri Branch and bound algoritme. V nakreslenej schéme rozšíreného regresného aj inverzného stromu teda prechádzame cez rovnakú postupnosť vrcholov (pričom však rovnaký vrchol v schéme zodpovedá rôznym modelom v regresnom a inverznom strome). Avšak kým vo vrchole regresného stromu vždy volíme ľavú hranu, v inverznom strome pravú. Dualita teda spočíva v tom, že ak v jednom strome je hrana plná (pridanie/odobratie premennej z modelu), v druhom je prerušovaná (žiadna zmena v modeli). Prevzaté z [20], s. 444, obr. 12.6.

4. **Orezávanie vetiev**. Prišli sme do nejakého vrcholu (napr. 23. $\equiv S_{\text{NULL}}$) v regresnom strome. Zaujíma nás, či má zmysel prehľadávať nejakú vetvu redukovaného regresného stromu (napr. $\{⑤, ⑥\}$) pod ním. Modely v tejto vetve $(.2, .23)$ sú nadmodelmi modelu z daného vrcholu (S_{NULL}), jeho $RSS_{S_{\text{NULL}}}$ nám teda ako ohraničenie nepomôže. Pre účely ohraničovania bol ale skonštruovaný inverzný strom. Zodpovedajúci vrchol v inverznom strome (.234) je totiž kvôli inverznému usporiadaniu stromu nadmodelom pre modely z jeho vetvy $(.4, .24, .34, .234)$, ktoré sú ďalej nadmodelmi pre modely $(.2, .23)$ z vetvy regresného stromu. Ak teda

doteraz najnižšie RSS pre modely veľkosti $1, 2$ sú menšie než RSS modelu na začiatku vetvy v inverznom strome (.234), nemá zmysel danú vetvu prehľadávať v regresnom ani inverznom strome.

5. **Výsledok**: Po prehľadaní celého (regresného a súčasne inverzného) stromu získame postupnosť $\{S_k\}_{k=1}^{p}$ submodelov S_k s garantovane najnižším RSS_S pri danej veľkosti $|S| = k$ pre každú veľkosť $k = 1, \ldots, p$.

Poznámka: Branch and bound algoritmus by bolo možné realizovať aj len na jedinom strome so všetkými 2^p submodelmi $S \in \mathcal{S}$, ktorý by bol inverzne usporiadaný (t. j. s úplným modelom S_{FULL} v koreni a najmenšími modelmi v listoch). Pri takejto schéme by sa ale veľmi často stávalo, že splnenie podmienky orezania vetvy by zlyhalo len kvôli modelom najmenších veľkostí ([5], s. 9).

2.1.2 MIO formulácia

Alternatívne možno hľadať best subset riešenie na základe formulácie úlohy selekcie premenných v tvare MIO úlohy (1.16) navrhnutej v [2]. Úlohy zmiešaného celočíselného programovania (MIO) možno efektívne riešiť pomocou špeciálnych solverov. Jedným z nich je aj GUROBI, ktorý bol využitý v pôvodnom článku [2]. Zodpovedajúca implementácia MIO prístupu k riešeniu best subset selekcie s využitím GUROBI je voľne dostupná prostredníctvom balíčka `bestsubset` [13] pre programovací jazyk R. Pri riešení minimalizačných MIO úloh ukladá GUROBI v každom okamihu dosiaľ najnižšiu dosiahnutú hodnotu účelovej funkcie ako horný odhad minima. Jeho veľkou výhodou však je, že okrem toho poskytuje súčasne aj garantovaný dolný odhad hľadaného minima. Šírka intervalu medzi týmito dvoma ohraničeniami, označovaná ako MIO medzera, nám umožňuje odhadnúť, nakoľko je aktuálny výsledok dobrou aproximáciou globálneho minima. Ako je však v článkoch [2, 12] viackrát spomenuté, je bežné, že GUROBI nájde veľmi kvalitný horný odhad minima, nie je však schopné v reálnom čase sa k nemu priblížiť spodným odhadom, a preto MIO medzera zostáva veľká. Aj napriek veľkej hodnote MIO medzery tak môžu byť získané výsledky (horný odhad) dostatočne blízko globálnemu minimu.

V článku [2] je uvedených niekoľko spôsobov ako v (1.16) správne určiť neznámu

konštantu M_U tak, aby bola zabezpečená ekvivalencia medzi úlohami (1.16) a (1.15). Jeden z nich, použiteľný v prípade $N > p$ (čo predpokladáme), je založený na riešení systému úloh kvadratického programovania

$$\begin{aligned} u_j^+ &\equiv \max_{\boldsymbol{\beta}} \beta_j & u_j^- &\equiv \min_{\boldsymbol{\beta}} \beta_j \\ s.t. \quad & ||\mathbf{y} - \mathbf{X}\boldsymbol{\beta}||_2^2 \leq UB, & s.t. \quad & ||\mathbf{y} - \mathbf{X}\boldsymbol{\beta}||_2^2 \leq UB, \end{aligned} \tag{2.2}$$

kde $j = 1, \ldots, p$ a UB je nejaké konečné horné ohraničenie minima úlohy (1.15). Potom pre optimálne riešenie úlohy (1.15), reprezentované odhadom regresných koeficientov $\hat{\boldsymbol{\beta}}$, platí $u_j^- \leq \hat{\beta}_j \leq u_j^+$, pre $j = 1, \ldots, p$. Ak označíme $M_U^j = \max\left\{|u_j^-|, |u_j^+|\right\}$, môžeme hľadanú konštantu M_U zhora odhadnúť hodnotou $M_U = \max_j M_U^j$. Za horný odhad UB možno dosadiť napríklad $RSS_{S^{(k-1)}}$ odhadu optimálneho modelu $S^{(k-1)}$ menšej veľkosti $|S^{(k-1)}| = k-1$ alebo výsledok iného selekčného algoritmu. Závery článku [2] naznačujú, že najlepšie výsledky z hľadiska presnosti a rýchlosti konvergencie možno dosiahnuť, ak je MIO výpočet inicializovaný pomocou približného odhadu minima získaného špeciálnym optimalizačným algoritmom založeným na gradientných metódach prvého rádu. Gradientný algoritmus štartuje z viacerých bodov, na inicializáciu MIO výpočtu je použitý najlepší získaný výsledok. Štartovacie body môžu byť zvolené náhodne, prípadne náhodnou perturbáciou nejakého zadaného bodu. Výsledky prezentované v kapitole 3 boli získané tak, že pri riešení MIO úlohy (1.16) pre veľkosť submodelu k boli štartovacie body generované perturbáciami submodelu $S^{(k-1)}$ vypočítaného ako odhad optima pre veľkosť $k-1$. Bližšie detaily tu neuvádzame, čitateľ ich môže nájsť v článku [2]. Pre naše účely postačuje, že balíček [13] poskytuje algoritmus pre riešenie MIO úlohy (1.16) detailne implementovaný podľa inštrukcií z uvedeného článku.

2.2 Dopredná a spätná selekcia

2.2.1 Dopredná selekcia

Dopredná selekcia (forward stepwise selection) je heuristický pažravý algoritmus, ktorý sa vyhýba exponenciálne zložitému problému prehľadávania všetkých 2^p možných submodelov. Začína sa s prázdnym modelom $S^{(0)} \equiv S_{\text{NULL}}$. V k-tom kroku ($k \in \{1, \ldots p\}$)

sa vytvorí nový model $S^{(k)} = S^{(k-1)} \cup \{j^*\}$ pridaním prediktora x_{j^*},

$$j^* = \underset{j \in S_{\text{FULL}} \setminus S^{(k-1)}}{\operatorname{argmin}} RSS_{S^{(k-1)} \cup \{j\}}, \tag{2.3}$$

ktorý spôsobí najväčší pokles $RSS_{S^{(k)}}$ oproti $RSS_{S^{(k-1)}}$. Výsledkom je postupnosť $p+1$ do seba vnorených modelov $S^{(0)} \subset S^{(1)} \subset \ldots \subset S^{(p)}$, pričom $|S^{(k)}| = k$. Víťazný model, ktorý by mal minimalizovať očakávanú predikčnú chybu sa z tejto postupnosti vyberie pomocou nejakého informačného kritéria alebo (cross)validácie.

Tento algoritmus nezaručuje nájdenie optimálnej množiny $\{S_k\}_{k=1}^p$ submodelov S_k s najnižším RSS_S pri danej veľkosti $|S| = k$ pre každú veľkosť $k \in \{1, \ldots, p\}$. Pri konštrukcii kontrapríkladu stačí vytvoriť situáciu, kedy optimálne modely $\{S_k\}_{k=1}^p$ nie sú do seba vnorené. Napriek tomu dopredná selekcia je v praxi mnohokrát postačujúca.

Výber správneho prediktora

Ako nájsť prediktor x_{j^*}, $j^* \in S_{\text{FULL}} \setminus S^{(k)}$, ktorého pridaním do $S^{(k)}$ dosiahneme najnižšie $RSS_{S^{(k)} \cup \{j\}}$? Mohli by sme skonštruovať regresiu pre všetkých $p-k$ prípustných modelov $S^{(k)} \cup \{j\}$ a na základe hodnôt $RSS_{S^{(k)} \cup \{j\}}$ vybrať optimálny model. To by ale nebolo výpočtovo efektívne. Výhodnejšie je tento krok realizovať s použitím QR rozkladu matice plánu $\mathbf{X}$. Rovnakou postupnosťou ako pridávame premenné do modelu, Gramm-Schmidtovou ortogonalizáciou vytvárame ortonormálnu bázu zo zodpovedajúcich stĺpcov matice $\mathbf{X}$.

Majme submodel $S^{(k)}$ veľkosti $|S^{(k)}| = k$ daný indexovou množinou $\{i_1, \ldots, i_k\}$ (prediktormi $\{x_{i_1}, \ldots, x_{i_k}\}$). Pomocou Gramm-Schmidtovej ortogonalizácie bola zo stĺpcov $\mathbf{x}_{i_1}, \ldots, \mathbf{x}_{i_k}$ matice $\mathbf{X}$ vytvorená ortonormálna báza $\mathbf{q}_{i_1}, \ldots, \mathbf{q}_{i_k}$, t. j.

$$\mathbf{X}_k \equiv (\mathbf{x}_{i_1} \ldots \mathbf{x}_{i_k}) = (\mathbf{q}_{i_1} \ldots \mathbf{q}_{i_k})\, \mathbf{R}_k \equiv \mathbf{Q}_k \mathbf{R}_k.$$

Pridaním prediktora $x_{i_{k+1}}$ sa ako ďalší v poradí zortogonalizuje stĺpec $\mathbf{x}_{i_{k+1}}$ na $\mathbf{q}_{i_{k+1}}$, maticovo dostaneme $\mathbf{X}_{k+1} = \mathbf{Q}_{k+1}\mathbf{R}_{k+1}$. Cieľom je voľbou i_{k+1} minimalizovať

$$\begin{aligned}
RSS_{S^{(k)} \cup \{i_{k+1}\}} =& \mathbf{y}^\top \left[\mathbf{I} - \mathbf{X}_{k+1} \left(\mathbf{X}_{k+1}^\top \mathbf{X}_{k+1}\right)^{-1} \mathbf{X}_{k+1}^\top\right] \mathbf{y} \\
=& \mathbf{y}^\top \left[\mathbf{I} - \mathbf{Q}_{k+1} \left(\mathbf{Q}_{k+1}^\top \mathbf{Q}_{k+1}\right)^{-1} \mathbf{Q}_{k+1}^\top\right] \mathbf{y} \\
=& \mathbf{y}^\top \left[\mathbf{I} - \mathbf{Q}_{k+1}\mathbf{Q}_{k+1}^\top\right] \mathbf{y} = \mathbf{y}^\top \left[\mathbf{I} - \sum_{j=1}^{k} \mathbf{q}_{i_j} \mathbf{q}_{i_j}^\top\right] \mathbf{y} - \mathbf{y}^\top \mathbf{q}_{i_{k+1}} \mathbf{q}_{i_{k+1}}^\top \mathbf{y} \\
=& RSS_{S^{(k)}} - \left(\mathbf{y}^\top \mathbf{q}_{i_{k+1}}\right)^2,
\end{aligned} \tag{2.4}$$

kde sme v druhej rovnosti využili regularitu $(k+1) \times (k+1)$ matice $\mathbf{R}_{k+1}$ a v tretej rovnosti sme aplikovali identitu $\mathbf{Q}_{k+1}^{\top}\mathbf{Q}_{k+1} = \mathbf{I}_{k+1}$, vyplývajúcu z ortonormality stĺpcov $N \times (k+1)$ matice $\mathbf{Q}_{k+1}$. Kedže $RSS_{S^{(k)}}$ nezávisí od voľby i_{k+1}, nový prediktor treba zvoliť tak, aby sa maximalizoval výraz $\left(\mathbf{y}^{\top}\mathbf{q}_{i_{k+1}}\right)^2$, t. j. dĺžka projekcie vektora $\mathbf{y}$ do priestoru[4] $\mathcal{M}\left(\mathbf{X}_k\right)^{\perp} \cap \mathcal{M}\left(\mathbf{X}_{k+1}\right)$.

Okrem QR-rozkladu možno postupovať aj metódou sweepovania, pričom správny prediktor x_j, ktorý treba pridať, sa určí pomocou vzťahu (1.65). Nevýhodou je zhoršenie numerickej presnosti výsledkov v porovnaní s QR-rozkladom. Pri doprednej selekcii sa neprejaví hlavný benefit sweepovania, ktorým je možnosť jednoducho reprezentovať pridávanie aj odoberanie premenných. Naopak, pri QR-rozklade je pridávanie premenných jednoduché, avšak spätné odoberanie značne komplikované.

Poznámka: Možno sa stretnúť[5] aj s modifikáciou doprednej selekcie, pri ktorej sa konštrukcia postupnosti $\{S^{(k)}\}_{k=1}^{p}$ ukončí v okamihu, keď F-štatistika[6] porovnania modelu $S^{(k^*)}$ a úplného modelu S_{FULL} už je dostatočne malá. T. j. kým nemožno na zvolenej hladine významnosti považovať regresné koeficienty pri všetkých dosiaľ nepridaných premenných za nulové. Víťazný model sa potom nevyberá podľa informačného kritéria, ale je ním priamo $S^{(k^*)}$.

2.2.2 Spätná selekcia

Komplementárnym algoritmom k doprednej selekcii je spätná selekcia (backward stepwise selection). Začína sa s úplným modelom S_{FULL} a postupnosť do seba vnorených submodelov $S^{(0)} \subset S^{(1)} \subset \ldots \subset S^{(p)}$ sa generuje postupným odoberaním premenných. Konkrétne, pre $k \in \{1, \ldots, p\}$ platí $S^{(k-1)} = S^{(k)} \setminus \{j^*\}$, kde

$$j^* = \underset{j \in S^{(k)}}{\operatorname{argmin}} RSS_{S^{(k)} \setminus \{j\}}. \tag{2.6}$$

[4]Uvedený priestor má v regulárnom prípade dimenziu 1, ide teda o projekciu do určitého smeru.

[5]Napr. v [20], s. 414.

[6]Spomenutá F-štatistika je daná vzťahom

$$F^{k^*} = \frac{(RSS_{S^{(k^*)}} - RSS_{S_{\text{FULL}}})/(p - k^*)}{RSS_{S_{\text{FULL}}}/(N - p)}. \tag{2.5}$$

Zodpovedá testovaniu hypotézy $\mathcal{H}_0 : \beta_j = 0$, $j \in S_{\text{FULL}} \setminus S^{(k^*)}$ o významnosti množiny premenných $\left\{x_j \,|\, j \in S_{\text{FULL}} \setminus S^{(k^*)}\right\}$ v úplnom modeli S_{FULL}.

Keďže prechádzame od modelu $S^{(k)}$ k jeho submodelu $S^{(k-1)}$, RSS odobratím x_{j^*} neklesne, v praxi vzrastie. V každom kroku algoritmu teda hľadáme takú premennú x_{j^*}, pri odobratí ktorej vzrastie RSS čo najmenej. Je to teda tá premenná, ktorá je (v zmysle F-štatistiky[7]) najmenej dôležitá pre regresiu v modeli $S^{(k)}$.

Na identifikáciu prediktora, ktorý treba odstrániť, existuje opäť veľmi rýchly postup. Predpokladajme, že sme v situácii, keď chceme odstrániť prediktor z modelu $S^{(k)} = \{i_1, \ldots, i_k\}$ (s premennými $x_{i_1}, \ldots, x_{i_k}$) veľkosti $|S^{(k)}| = k$, máme k dispozícii LS odhad regresných koeficientov $\hat{\boldsymbol{\beta}}_{S^{(k)}}$ tohto modelu a maticu $\left(\mathbf{X}_{S^{(k)}}^\top \mathbf{X}_{S^{(k)}}\right)^{-1}$. Submodel $S^{(k-1)} = S^{(k)} \setminus \{j\}$ vytvorený odstránením premennej x_j, $j \in S^{(k)}$, zodpovedá podmienke $\left(\boldsymbol{\beta}_{S^{(k)}}\right)_j = \mathbf{e}_j^\top \boldsymbol{\beta}_{S^{(k)}} = 0$. Pre odhad regresných koeficientov submodelu $S^{(k-1)}$ potom platí

$$\begin{aligned}\hat{\boldsymbol{\beta}}_{S^{(k-1)}} =& \hat{\boldsymbol{\beta}}_{S^{(k)}} - \left(\mathbf{X}_{S^{(k)}}^\top \mathbf{X}_{S^{(k)}}\right)^{-1} \mathbf{e}_j \left[\mathbf{e}_j^\top \left(\mathbf{X}_{S^{(k)}}^\top \mathbf{X}_{S^{(k)}}\right)^{-1} \mathbf{e}_j\right]^{-1} \mathbf{e}_j^\top \hat{\boldsymbol{\beta}}_{S^{(k)}} \\ =& \hat{\boldsymbol{\beta}}_{S^{(k)}} - \frac{\left(\mathbf{X}_{S^{(k)}}^\top \mathbf{X}_{S^{(k)}}\right)^{-1} \mathbf{e}_j \mathbf{e}_j^\top \hat{\boldsymbol{\beta}}_{S^{(k)}}}{v_{jj}^k} \equiv \hat{\boldsymbol{\beta}}_{S^{(k)}} - \Delta\hat{\boldsymbol{\beta}}_{S^{(k)},S^{(k-1)}},\end{aligned} \tag{2.8}$$

kde sme kvôli sprehľadneniu zaviedli označenie $v_{ij}^k = \left[\left(\mathbf{X}_{S^{(k)}}^\top \mathbf{X}_{S^{(k)}}\right)^{-1}\right]_{ij}$ a korekčný člen označili $\Delta\hat{\boldsymbol{\beta}}_{S^{(k)},S^{(k-1)}}$. Z rozmerových dôvodov používame mierne nekonzistentné značenie, kde vektor $\hat{\boldsymbol{\beta}}_{S^{(k-1)}}$ nemá dĺžku $k-1$ ale k a platí $\left(\hat{\boldsymbol{\beta}}_{S^{(k-1)}}\right)_j = 0$.

Následne

$$\begin{aligned}RSS_{S^{(k-1)}} =& \left[\mathbf{y} - \mathbf{X}_{S^{(k)}}\hat{\boldsymbol{\beta}}_{S^{(k-1)}}\right]^\top \left[\mathbf{y} - \mathbf{X}_{S^{(k)}}\hat{\boldsymbol{\beta}}_{S^{(k-1)}}\right] \\ =& \left[\left(\mathbf{y} - \mathbf{X}_{S^{(k)}}\hat{\boldsymbol{\beta}}_{S^{(k)}}\right) + \mathbf{X}_{S^{(k)}}\Delta\hat{\boldsymbol{\beta}}_{S^{(k)}}\right]^\top \left[\left(\mathbf{y} - \mathbf{X}_{S^{(k)}}\hat{\boldsymbol{\beta}}_{S^{(k)}}\right) + \mathbf{X}_{S^{(k)}}\Delta\hat{\boldsymbol{\beta}}_{S^{(k)}}\right] \\ =& || \left(\mathbf{y} - \mathbf{X}_{S^{(k)}}\hat{\boldsymbol{\beta}}_{S^{(k)}}\right) ||_2^2 + \left(\Delta\hat{\boldsymbol{\beta}}_{S^{(k)},S^{(k-1)}}\right)^\top \mathbf{X}_{S^{(k)}}^\top \mathbf{X}_{S^{(k)}} \Delta\hat{\boldsymbol{\beta}}_{S^{(k)},S^{(k-1)}} \\ =& RSS_{S^{(k)}} + \frac{\hat{\boldsymbol{\beta}}_{S^{(k)}}^\top \mathbf{e}_j \mathbf{e}_j^\top \left(\mathbf{X}_{S^{(k)}}^\top \mathbf{X}_{S^{(k)}}\right)^{-1} \mathbf{e}_j \mathbf{e}_j^\top \hat{\boldsymbol{\beta}}_{S^{(k)}}}{\left(v_{jj}^k\right)^2} \\ =& RSS_{S^{(k)}} + \frac{\hat{\boldsymbol{\beta}}_{S^{(k)}}^\top \mathbf{e}_j \mathbf{e}_j^\top \hat{\boldsymbol{\beta}}_{S^{(k)}}}{v_{jj}^k} = RSS_{S^{(k)}} + \frac{\left(\hat{\boldsymbol{\beta}}_{S^{(k)}}\right)_j^2}{v_{jj}^k},\end{aligned} \tag{2.9}$$

[7] V tomto prípade je F-štatistika daná vzorcom

$$F_j^k = \frac{\left(RSS_{S'} - RSS_{S^{(k)}}\right)/1}{RSS_{S^{(k)}}/(N-k)}, \tag{2.7}$$

kde $S' = S^{(k)} \setminus \{j\}$ pre nejaké fixné $j \in S^{(k)}$. Toto zodpovedá testovaniu hypotézy $\mathcal{H}_0 : \beta_j = 0$ o významnosti premennej x_j v modeli $S^{(k)}$.

kde sme v tretej rovnosti využili $\left(\mathbf{y} - \mathbf{X}_{S^{(k)}}\hat{\boldsymbol{\beta}}_{S^{(k)}}\right)^\top \mathbf{X}_{S^{(k)}}\Delta\hat{\boldsymbol{\beta}}_{S^{(k)},S^{(k-1)}} = 0$, nakoľko reziduálny vektor $\mathbf{y} - \mathbf{X}_{S^{(k)}}\hat{\boldsymbol{\beta}}_{S^{(k)}}$ patrí do ortognálneho doplnku $\mathcal{M}\left(\mathbf{X}_{S^{(k)}}\right)^\perp$ a súčasne $\mathbf{X}_{S^{(k)}}\Delta\hat{\boldsymbol{\beta}}_{S^{(k)},S^{(k-1)}} \in \mathcal{M}\left(\mathbf{X}_{S^{(k)}}\right)$; v piatej rovnosti sme využili definíciu v_{ij}^k. Keďže $RSS_{S^{(k)}}$ nezávisí od voľby odoberaného prediktora x_j, submodel $S^{(k-1)}$ s minimálnym $RSS_{S^{(k-1)}}$ získame z $S^{(k)}$ vyhodením premennej x_{j^*},

$$j^* = \underset{j\in S^{(k)}}{\operatorname{argmin}} \frac{\left(\hat{\boldsymbol{\beta}}_{S^{(k)}}\right)_j^2}{v_{jj}^k} = \underset{j\in S^{(k)}}{\operatorname{argmin}} F_j^k, \tag{2.10}$$

kde F_j^k, daná vzorcom (2.7), je F-štatistika zodpovedajúca testovaniu hypotézy $\mathcal{H}_0 : \beta_j = 0$ o významnosti premennej x_j v modeli $S^{(k)}$. Odvodili sme teda rovnaký výsledok ako pri (1.66).

Opäť, víťazný model možno z postupnosti $S^{(0)} \subset S^{(1)} \subset \ldots \subset S^{(p)}$ vybrať pomocou určitého informačného kritéria alebo (cross)validácie. Alebo sa konštrukcia postupnosti $\{S^{(k)}\}_{k=1}^p$ ukončí v okamihu, keď F-štatistika (2.5) porovnania modelu $S^{(k^*)}$ a úplného modelu $S_{\text{FULL}} \equiv S^{(p)}$ už je dostatočne malá a víťazom je $S^{(k^*)}$.

Pre výpočtovú efektívnosť tohto algoritmu je výhodné, aby sme po zvolení x_j mali ľahko k dispozícii LS odhad $\hat{\beta}_{S^{(k-1)}}$ a maticu $\left(\mathbf{X}_{S^{(k-1)}}^\top \mathbf{X}_{S^{(k-1)}}\right)^{-1}$ pre model $S^{(k-1)} = S^{(k)} \setminus \{j\}$. Uvedené veličiny sú totiž potrebné vo vzorci (2.10) a ich dodatočné dopočítavanie by spomaľovalo celkový výpočet. Našťastie, pri použití formalizmu sweepovania sú oba tieto objekty po každom prechode do nového modelu updateované pri nízkych výpočtových nákladoch.

2.2.3 Kombinovaná stratégia

Pri doprednej aj spätnej selekcii je povolené výhradne len pridávanie premenných alebo len ich odoberanie. Môžeme však umožniť prechody medzi modelmi oboma spôsobmi - odoberaním aj pridávaním premenných (ale v rôznych krokoch). Ktorá z akcií sa vykoná a ktorá premenná sa pridá/odoberie, sa opäť dá riadiť F-štatistikami. Na tejto myšlienke je založený algoritmus *stepwise regression algorithm*, ktorý má zaručenú konvergenciu do netriviálneho modelu ([20], s. 418-420).

Rovnako je však tento prístup vhodný na prehľadávanie množiny submodelov $\mathcal{S}$ pomocou algoritmov stochastickej optimalizácie (napr. simulované žíhanie, genetické algoritmy). Vďaka sweepovaniu možno výpočtovo lacno pridávať aj odoberať premenné,

a teda prechádzať medzi submodelmi. Táto idea je základom nami navrhnutého výmenného KL algoritmu pre selekciu premenných.

2.3 Vlastné návrhy selekčných algoritmov

2.3.1 Výmenný KL algoritmus - VS.KL

Je to vlastný, zatiaľ prototypový, návrh algoritmu, ktorý vychádza z KL výmenného algoritmu z oblasti optimálneho návrhu experimentu [19, 1], kde je cieľom zvoliť také body merania (hodnoty p-tíc $\{x_{i1}, \ldots, x_{ip}\}$), aby sa pri danom počte meraní N maximalizovala informácia získaná z experimentu, súvisiaca s Fisherovou informačnou maticou. Písmená VS v označení VS.KL sú skratkou z **V**ariable **S**election, KL odkazuje na spôsob výberu kandidátnych premenných pre výmenu.

Algoritmus sa skladá z dvoch častí. Vnútorná časť, reprezentovaná funkciou `VS.KL.in`, sa snaží nájsť submodel $S_k = \underset{S \in \mathcal{S}_k}{\operatorname{argmin}} RSS_S$ s minimálnou hodnotou RSS_S v rámci množiny submodelov $\mathcal{S}_k$ veľkosti $k \in \{1, \ldots, p\}$. Vonkajšia časť (funkcia `VS.KL.seq`) vytvára postupnosť $\left\{S^{(k)}\right\}_{k=1}^{p}$ odhadov submodelov $\{S_k\}_{k=1}^{p}$ získaných postupným volaním funkcie `VS.KL.in` pre zvyšujúce sa k. Pri inicializácii výpočtu modelu $S^{(k)}$ je využitá znalosť predchádzajúceho modelu $S^{(k-1)}$. V každom l-tom kroku (napr. v každom ($l = 1$), každom druhom ($l = 2$), ...) sa navyše výpočet inicializuje aj z úplne náhodného základu, aby sa znížilo riziko záchytu v lokálnom minime.

V nasledujúcom popise algoritmu sa každý model S reprezentuje zodpovedajúcou maticou $\mathbf{B}^S = \mathrm{ppt}^+(\mathbf{A}, S)$ ($\mathbf{B}^S = \mathrm{ppt}(\mathbf{A}, S)$) v súlade s procedúru (ne)symetrizovaného sweepovania, pričom $RSS_S = \mathbf{B}^S_{p+1,p+1}$. Pridávanie/odoberanie premenných sa tiež reprezentuje zodpovedajúcimi operáciami sweepovania.

function `VS.KL.IN`(int K, L, k, m_1, m_2, m_3, matica $\mathbf{A}$, model S_{IN}, double $t_{\max}$)

$RSS_{\mathrm{BEST}} \leftarrow RSS_{S_{\mathrm{IN}}}$, $S_{\mathrm{BEST}} \leftarrow S_{\mathrm{IN}}$

while čas výpočtu $< t_{\max}$ **do**

① **reštart**: Vytvor model S veľkosti k ako zjednotenie S_{I} (náhodný výber m_1 premenných z prvej polovice „najdôležitejších“ (viď usporiadanie v ②) premenných v S_{BEST}), S_{II} (náhodný výber m_2 premenných z množiny $S_{\mathrm{BEST}} \setminus S_{\mathrm{I}}$) a S_{III} (náhodný

výber m_3 premenných z $S_{\text{FULL}} \setminus (S_{\text{I}} \cup S_{\text{II}})$. Doplň premenné do veľkosti k doprednou selekciou.

② Zoraď indexy $i \in S$ do postupnosti $\{i_r\}_{r=1}^{|S|}$ vzostupne podľa $RSS_{S\setminus\{i\}}$ (vzťah (1.66)) a $j \notin S$ do postupnosti $\{j_s\}_{s=1}^{p-|S|}$ vzostupne podľa $RSS_{S\cup\{j\}}$ (vzťah (1.65)).

for r in $1 : \min\{K, |S|\}$ **do**
for s in $1 : \min\{L, p - |S|\}$ **do**
$S' \leftarrow (S \setminus \{i_r\}) \cup \{j_s\}$, $i_r \in S$, $j_s \notin S$
if $RSS_{S'} < RSS_S$ **then**
$S \leftarrow S'$, prejdi na ②
end if
end for
end for
if $RSS_S < RSS_{\text{BEST}}$ **then**
$RSS_{\text{BEST}} \leftarrow RSS_S$, $S_{\text{BEST}} \leftarrow S$
end if
prejdi na ①
end while
return RSS_{BEST}, S_{BEST}
end function

function `VS.KL.SEQ`(int $K_1, K_2, L_1, L_2, p, m_{11}, m_{12}, m_{13}, m_{21}, m_{22}, m_{23}$, matica $\mathbf{A}$, double $t^1_{\max}, t^2_{\max}, t^3_{\max}, t^4_{\max}$)
$S^{(0)} \leftarrow \emptyset$
for k in $1 : p$ **do**
if $k \bmod l == 1$ OR $l == 1$ **then**
$S_{\text{IN}} \leftarrow \emptyset$
$S' \leftarrow$ `VS.KL.IN`$(K_1, L_1, k, m_{11}, m_{12}, m_{13}, \mathbf{A}, S_{\text{IN}}, t^1_{\max})$
$RSS_{S'}, S' \leftarrow$ `VS.KL.IN`$(K_2, L_2, k, m_{21}, m_{22}, m_{23}, \mathbf{A}, S', t^2_{\max})$
end if
$S_{\text{IN}} \leftarrow S^{(k-1)}$
$S \leftarrow$ `VS.KL.IN`$(K_1, L_1, k, m_{11}, m_{12}, m_{13}, \mathbf{A}, S_{\text{IN}}, t^3_{\max})$

$RSS_S, S \leftarrow$ `VS.KL.IN`$(K_2, L_2, k, m_{21}, m_{22}, m_{23},\ \mathbf{A},\ S,\ t^4_{\max})$

if $(k \bmod l == 1$ OR $l == 1)$ AND $RSS_S \geq RSS_{S'}$ **then**

$S^{(k)} \leftarrow S'$, $RSS_{S^{(k)}} \leftarrow RSS_{S'}$

else

$S^{(k)} \leftarrow S$, $RSS_{S^{(k)}} \leftarrow RSS_S$

end if

end for

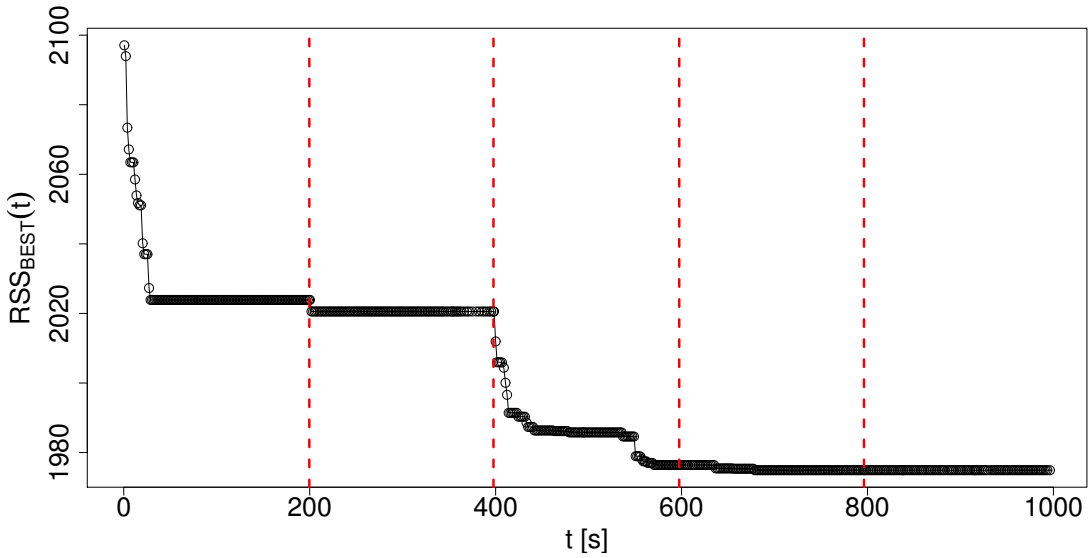
return $\left\{S^{(k)}\right\}_{k=0}^{p}$, $\{RSS_{S^{(k)}}\}_{k=0}^{p}$

end function

Vo funkcii `VS.KL.seq` je funkcia `VS.KL.in` volaná v bloku 2-krát po sebe s rôznymi nastaveniami parametrov. Prax totiž ukazuje, že (v analógii s ideou simulovaného žíhania) je vhodné pri prvom volaní funkcie `VS.KL.in` zvoliť relatívne vysoký počet m_{13} náhodne vybratých nových premenných pri reštarte (cca $m_{13} \approx 0.1 * k$), aby bolo umožnené kvalitne nahrubo preskúmať množinu $\mathcal{S}_k$. Vysoký podiel nových náhodne pridaných premenných však súčasne spôsobí, že pri reštartovaní nie je `VS.KL.in` schopný jemnejšie prehľadať okolie aktuálne najlepšieho modelu S_{BEST}, preto sa po krátkom čase konvergencia k optimálnemu submodelu S_k výrazne spomalí, prakticky zastaví.

Výsledný model S_{BEST} prvého behu `VS.KL.in` preto vložíme ako vstup druhého volania funkcie `VS.KL.in`, kde $m_{23} \approx 0$ alebo priamo $m_{23} = 0$, teda pri reštartovaní nepridávame žiadne nové premenné náhodne, ale iba pomocou doprednej selekcie. Aj napriek tomu reštartovanie dobre funguje, lebo množina ponechaných $m_{21} + m_{22} < k$ premenných je náhodná. Pritom sa ale stále držíme v okolí aktuálne najlepšieho modelu, takže sa opäť rýchlo rozbehne konvergencia k (lokálnemu) minimu chyby RSS_S. Keď výraznejší pokles RSS_{BEST} opäť ustane, môžeme už predpokladať, že sme dokonvergovali do blízkosti nejakého lokálneho minima. Opísaný priebeh je ilustrovaný na obr. 2.3. Jasne vidno, že po počiatočnom rýchlom poklese RSS_{BEST} pri $m_3 = 10 > 0$ bolo potrebné na docielenie ďalšieho poklesu nastaviť $m_3 = 0$. Toto viedlo k obnoveniu rýchlej konvergencie do oblasti nejakého lokálneho minima.

Ďalšie zlepšenie (skonvergovanie do blízkosti lepšieho lokálneho či dokonca globálneho minima RSS_{S_k}) je možné docieliť viacnásobným spustením tohto bloku dvoch



Obr. 2.3: Typický priebeh konvergencie RSS_{BEST} v závislosti od doby výpočtu t pri použití funkcie `VS.KL.in`. Výsledky boli získané na dátach z časti 3.1 pre $k = 80$, $p = 382$ a $N = 21792$ (všetky dáta boli použité na trénovanie). Prezentovaná závislosť zodpovedá postupnosti piatich volaní funkcie `VS.KL.in` s parametrami $(K, L, m_1, m_2, m_3) = (5, 10, 25, 25, 10)$, $(20, 20, 30, 30, 10)$, $(20, 20, 35, 35, 0)$, $(30, 30, 35, 35, 0)$ a $(40, 40, 35, 35, 0)$, aplikovanými v uvedenom poradí. Doba behu každého volania bola $t_{\max} = 200s$, jednotlivé etapy sú oddelené zvislou prerušovanou červenou čiarou. Pri prvom volaní sme začínali s $S_{\text{IN}} = \emptyset$ (náhodná inicializácia S_{BEST}), vstupom S_{IN} každého ďalšieho volania bol výstupný model S_{BEST} z predchádzajúcej etapy.

volaní s náhodným začiatkom. Nikdy však nebudeme vedieť, ako ďaleko sme od globálneho minima. Pri výpočtoch na malých modeloch s $p \lesssim 50$ (a za istých podmienok pre $p \lesssim 100$) je ale možné nájsť optimálne modely $\{S_k\}_{k=1}^{p}$ prehľadávaním celej množiny 2^p submodelov $\mathcal{S}$ s pomocou branch and bound algoritmu z časti 2.1.1. V takýchto prípadoch výmenný KL algoritmus pri našich výpočtoch zatiaľ vždy našiel optimálne submodely S_k získané branch and bound algoritmom, alebo sa chyba RSS_S ním zvolených modelov od RSS_{S_k} líšila zanedbateľne málo. Pritom výpočet pomocou VS.KL trval v porovnaní s branch and bound výrazne kratší čas (rádovo minúty až desiatky minút).

Samotná idea výmenného algoritmu pre selekciu premenných nie je nová. Známy je algoritmus *sequential replacement* pochádzajúci od Millera [18]. V rámci neho sa

postupne testujú možnosti výmeny každej premennej z modelu S (jednotlivo) za každú z premenných mimo S. Avšak tento algoritmus je deterministický, čo v praxi zvyčajne zhoršuje konvergenciu. Výpočet je navyše oproti VS.KL pomalší, pretože sa v každej iterácii testuje výrazne viac kombinácií premenných než $K \times L$ najperspektívnejších.

2.3.2 Validačné verzie selekčných algoritmov

Ako už bolo diskutované v časti 1.3, submodel S_k veľkosti k, minimalizujúci trénovaciu chybu RSS_S v rámci množiny $\mathcal{S}_k$, nemusí byť optimálnou voľbou z hľadiska minimalizácie očakávanej predikčnej chyby. Pre väčšinu veľkostí k totiž existuje veľmi veľa modelov S s len nepatrne horším RSS_S oproti RSS_{S^k}. Tieto modely by ale mohli viesť k nižšej validačnej chybe, ktorá je pre nás podstatnejšia. VS.KL algoritmus či doprednú (alebo spätnú) selekciu preto môžeme modifikovať tak, aby namiesto úlohy (1.15) riešili (1.17). Pri tejto úprave stačí v schémach algoritmov nahradiť RSS_S validačnou chybou $MSE_S^{\mathrm{Val}} = ||\mathbf{y}^{\mathrm{Val}} - \mathbf{X}^{\mathrm{Val}}\hat{\boldsymbol{\beta}}_S||_2^2$, kde $\hat{\boldsymbol{\beta}}_S$ je LS odhad regresných koeficientov pre model S vypočítaný na trénovacej množine. Základom je stále formalizmus sweepovania, ktorý nám v každom kroku umožňuje extrahovať potrebný vektor ako $\hat{\boldsymbol{\beta}}_S = -\mathbf{B}^S_{S,p+1}$ (pre nesymetrizované sweepovanie) alebo $\hat{\boldsymbol{\beta}}_S = \mathbf{B}^S_{S,p+1}$ (pre symetrizované sweepovanie). Validačná chyba MSE_S^{Val} je vypočítaná dosadením vektora $\hat{\boldsymbol{\beta}}_S$ (doplneného nulami pre zložky $j \notin S$) do kvadratickej formy

$$MSE_S^{\mathrm{Val}} = \hat{\boldsymbol{\beta}}_S^\top \left(\mathbf{X}^{\mathrm{Val}}\right)^\top \mathbf{X}^{\mathrm{Val}}\hat{\boldsymbol{\beta}}_S - 2\left(\mathbf{y}^{\mathrm{Val}}\right)^\top \mathbf{X}^{\mathrm{Val}}\hat{\boldsymbol{\beta}}_S + \left(\mathbf{y}^{\mathrm{Val}}\right)^\top \mathbf{y}^{\mathrm{Val}}. \tag{2.11}$$

Vo formalizme nesymetrizovaného sweepovania využívame pri výpočte $MSE^{\mathrm{Val}}_{S\cup\{j\}}$, $j \notin S$, a $MSE^{\mathrm{Val}}_{S\setminus\{i\}}$, $i \in S$, vzťahy

$$\hat{\boldsymbol{\beta}}_{S\cup\{j\}} = \begin{pmatrix} 0 \\ \hat{\boldsymbol{\beta}}_S \end{pmatrix} - \frac{b^S_{j,p+1}}{b^S_{j,j}} \begin{pmatrix} 1 \\ \mathbf{B}^S_{S,j} \end{pmatrix}, \quad \hat{\boldsymbol{\beta}}_{S\setminus\{i\}} = \left(\hat{\boldsymbol{\beta}}_S\right)_{S\setminus\{i\}} - \frac{b^S_{i,p+1}}{b^S_{i,i}}\mathbf{B}^S_{S\setminus\{i\},i}, \tag{2.12}$$

ktoré sa ľahko odvodia z (1.36). Prakticky identické vzorce možno odvodiť aj pri symetrizovanom sweepovaní.

Výpočet MSE_S^{Val} dosadzovaním $\hat{\boldsymbol{\beta}}_S$ do kvadratickej formy (2.11) spôsobuje zvýšenie výpočtovej náročnosti validačných verzií algoritmov oproti pôvodným metódam. Na druhej strane, tento spôsob výpočtu chyby je numericky stabilnejší. Aj v prípade vý-

počtu RSS_S vedie analogický postup k (výrazne) presnejším výsledkom než poskytuje hodnota $\mathbf{B}^S_{p+1,p+1}$.

2.4 Penalizačné metódy

Best subset selekcia, dopredná/spätná selekcia aj výmenný KL algoritmus sú algoritmy diskrétnej optimalizácie prehľadávajúce množinu 2^p submodelov $\mathcal{S}$. Spolu s MIO ich preto budeme súhrnne označovať ako prehľadávacie algoritmy. Konceptuálne odlišný prístup k problému selekcie premenných predstavujú tzv. penalizačné[8] metódy (shrinkage methods), pri ktorých dochádza k identifikácii nedôležitých prediktorov vynulovaním (alebo výrazným potlačením) zodpovedajúcich regresných koeficientov $\hat{\beta}_j$ vďaka dodatočnej penalizácii. Pri týchto metódach je však nutnou podmienkou ich úspešnosti naškálovanie pozorovaných hodnôt prediktorov, t. j. stĺpcov $\mathbf{x}_j$ matice plánu $\mathbf{X}$ tak, aby hodnoty v každom stĺpci mali približne rovnaký rozsah[9]. Inak by vplyv penalizácie na premennú s menšími (absolútnymi) hodnotami bol výrazne vyšší než na premennú s väčšími hodnotami. Medzi základné penalizačné algoritmy patrí ridge regression a LASSO.

2.4.1 Ridge Regression

Riešime LS problém s dodatočnou penalizáciou na koeficienty β_j v tvare kvadrátu ℓ_2 normy, teda s účelovou funkciou

$$\sum_{i=1}^{N}\left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j\right)^2 + \lambda\sum_{j=1}^{p}\beta_j^2, \tag{2.13}$$

kde $\lambda \geq 0$ je tuning parameter, na ktorého správnom nastavení závisí kvalita získaného výsledku. Penalizačný člen zámerne neobsahuje intercept β_0, nakoľko ten súvisí s $\overline{y}$, čo je vo všeobecnosti nenulová hodnota, ktorú nemáme dôvod penalizovať. Pri voľbe $\lambda = 0$ dostávame klasickú LS úlohu, v limite $\lambda \to \infty$ je riešením triviálny model obsahujúci len intercept β_0. Nenulový koeficient pri hociktorej z premenných by totiž viedol k divergujúcej hodnote účelovej funkcie.

[8]Skupinu prehľadávacích algoritmov budeme ekvivalentne označovať aj ako nepenalizačné metódy.

[9]Zaužívané je štandardizovať všetky stĺpce $\mathbf{x}_j$ tak, aby priemer stĺpca bol nulový a štandardná odchýlka bola rovná 1.

Uvedená formulácia je úloha na voľný extrém. Ekvivalentne ju však možno formulovať aj v tvare úlohy na viazaný extrém

$$\min_{\sum_{j=1}^{p}\beta_j^2 \leq t} \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2, \tag{2.14}$$

kde medzi parametrami t a λ existuje jednojednoznačný súvis.

Výhodou je, že vďaka použitiu ℓ_2 normy existuje analytické riešenie problému. Navyše, $\lambda > 0$ zlepšuje stabilitu riešenia sústavy normálnych rovníc. V dôsledku bias-variance trade-off existuje optimálne $\lambda > 0$, ktoré vedie k minimalizácii testovacej chyby MSE^{Test}. Toto možno vysvetliť nasledovne. Pri $\lambda = 0$, čo zodpovedá klasickej úlohe LS, máme vďaka veľkému počtu premenných v modeli nízky bias, no vysokú varianciu. Pri $\lambda \to \infty$, čo zodpovedá triviálnemu modelu, je odhad jedinej premennej v modeli - interceptu β_0 - len slabo citlivý na zmeny v dátach, má teda nízku varianciu. Samotný model je však príliš jednoduchý na to, aby dobre popisoval dáta, má preto vysoký bias. Oba tieto extrémne prípady teda vedú k vysokej testovacej chybe MSE^{Test}, ktorá je súčtom príspevkov od biasu a variancie.

Našim cieľom je zvoliť parameter λ čo najbližšie k optimálnemej hodnote minimalizujúcej MSE^{Test}. Za týmto účelom sa navzorkuje prípustný interval $\langle 0, \infty)$, čím získame množinu $\{\lambda_1, \ldots, \lambda_m\}$ a v týchto bodoch vypočítame validačnú chybu MSE^{Val} ako odhad MSE^{Test}. Zvolí sa λ v okolí minima postupnosti $\left\{MSE^{\text{Val}}(\lambda_i)\right\}_{i=1}^{m}$.

Nevýhodou je, že ridge regression nedokáže koeficienty β_j menej dôležitých premenných úplne znulovať, iba ich minimalizovať. V tomto zmysle teda nedochádza k selekcii premenných, lebo výsledný model obsahuje maximálnu množinu prediktorov $\mathcal{X}$.

2.4.2 LASSO

Obdobou ridge regression s použitím penalizácie v tvare ℓ_1 normy je algoritmus LASSO, pri ktorom je odhad regresných koeficientov $\hat{\boldsymbol{\beta}}^{\text{LASSO}}(\lambda)$[10] riešením optimalizačného problému na voľný extrém

$$\min_{\boldsymbol{\beta} \in \mathbb{R}^{p+1}} \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2 + \lambda \sum_{j=1}^{p} |\beta_j|. \tag{2.15}$$

[10]Explicitným označením „LASSO“ zdôrazňujeme, že odhad $\hat{\boldsymbol{\beta}}^{\text{LASSO}}(\lambda)$ sa vo všeobecnosti líši od LS odhadu $\hat{\boldsymbol{\beta}}_S = \hat{\boldsymbol{\beta}}_S^{LS}$, kde $S = \{j | \hat{\beta}_j^{\text{LASSO}}(\lambda) \neq 0\}$.

Uvedenú úlohu možno tiež ekvivalentne preformulovať ako optimalizačnú úlohu na viazaný extrém v tvare

$$\min_{\sum_{j=1}^{p}|\beta_j|\leq t} \sum_{i=1}^{N}\left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j\right)^2, \tag{2.16}$$

pre nejaké $t \equiv t(\lambda)$ jednoznačne dané hodnotou λ. Optimálnu hodnotu λ určujeme identickým postupom ako pri ridge regression. Platí[11], že pre $\lambda > \lambda_{\max} = \max_j |\mathbf{x}_j^\top \mathbf{y}|/N$, kde N je počet pozorovaní v regresii, je LASSO penalizácia tak silná, že $\hat{\beta}_j^{\text{LASSO}}(\lambda) = 0$, $j = 1, \ldots, p$, teda výsledkom LASSO selekcie je triviálny model bez premenných (len s interceptom). Pri hľadaní optimálnej λ je preto postačujúce sa obmedziť na interval $\lambda \in [0, \lambda_{\max}]$.

Použitie ℓ_1 normy má síce za následok, že pre riešenie neexistuje analytické vyjadrenie a je ho potrebné hľadať mierne komplikovanejším spôsobom, napr. pomocou lineárneho programovania. Napriek tomu má LASSO oproti ridge regression významnú výhodu v tom, že dokáže vyhadzovať premenné z modelu, t. j. získať riešenie s $\hat{\beta}_j^{\text{LASSO}} = 0$ pre nevýznamné prediktory. Využijúc formuláciu (2.16) možno túto vlastnosť zdôvodniť nasledovne. Odhliadnuc od β_0, množina prípustných riešení $\sum_{j=1}^{p} |\beta_j| \leq t$ je špeciálny typ symetrického polyédra v $\mathbb{R}^p$, centrovaného v počiatku, ktorého všetky vrcholy ležia na súradnicových osiach. Je preto vysoko pravdepodobné, že optimálne riešenie bude ležať na hrane tohto polyédra. Tie sú ale charakterizované nulovými hodnotami niektorých koeficientov β_j. (viď [6], obr. 6.7, s. 222). Čím vyššia je hodnota λ, tým menšie je t, teda prípustný polyéder sa zmenšuje. S rastúcim λ sa postupne zväčšuje množina indexov j, pre ktoré $\hat{\beta}_j^{\text{LASSO}} = 0$ (množina odstránených prediktorov), až nakoniec dostaneme triviálny model, kde jediným prediktorom s $\hat{\beta}_j^{\text{LASSO}} \neq 0$ je intercept.

LASSO je jediná penalizačná metóda založená na ℓ_p norme, ktorá dokáže selektovať premenné (t. j. priradiť nevýznamným prediktorom nulové koeficienty), a pritom zachovať konvexnosť riešenej úlohy. Konvexnosť sa zachováva pre $p \geq 1$, selektujúci algoritmus vzniká pre $p \leq 1$ (porov. [11], s. 2).

Množinu tých indexov, ktorým pri danej λ algoritmus LASSO priradil nenulové koeficienty $\hat{\beta}_j^{\text{LASSO}}(\lambda)$, budeme označovať α_λ. Inak povedané, α_λ je množina indexov

[11] Viď [11], s. 17

tých premenných, ktoré LASSO vybralo do modelu pri danom λ. Platí teda

$$\hat{\beta}_j^{\text{LASSO}}(\lambda) \neq 0,\ j \in \alpha_\lambda, \qquad \hat{\beta}_j^{\text{LASSO}}(\lambda) = 0,\ j \notin \alpha_\lambda. \tag{2.17}$$

Pri splnení podmienky, že stĺpce matice $\mathbf{X}$ sú v tzv. *všeobecnej polohe*[12] a $\lambda > 0$, platí (viď [23, 12]):

1. Optimálne riešenie $\hat{\boldsymbol{\beta}}^{\text{LASSO}}(\lambda)$ úlohy (2.15) je jednoznačné a pre jeho nenulovú časť $\hat{\boldsymbol{\beta}}_{\alpha_\lambda}^{\text{LASSO}}(\lambda)$ platí

$$\hat{\boldsymbol{\beta}}_{\alpha_\lambda}^{\text{LASSO}}(\lambda) = \left(\mathbf{X}_{\alpha_\lambda}^\top \mathbf{X}_{\alpha_\lambda}\right)^{-1} \left(\mathbf{X}_{\alpha_\lambda}^\top \mathbf{y} - \lambda \mathbf{s}\right), \tag{2.18}$$

   kde $\mathbf{s}$ je vektor znamienok koeficientov $\hat{\boldsymbol{\beta}}_{\alpha_\lambda}^{\text{LASSO}}(\lambda)$, t. j.

$$s_j = \operatorname{sgn}\left(\hat{\beta}_j^{\text{LASSO}}(\lambda)\right), j \in \alpha_\lambda. \tag{2.19}$$

2. Matica plánu $\mathbf{X}_{\alpha_\lambda}$ submodelu α_λ má plnú hodnosť. Preto LS odhad $\hat{\boldsymbol{\beta}}_{\alpha_\lambda}^{\text{LS}}$ tohto submodelu je tiež jednoznačný a platí

$$\hat{\boldsymbol{\beta}}_{\alpha_\lambda}^{\text{LS}} = \left(\mathbf{X}_{\alpha_\lambda}^\top \mathbf{X}_{\alpha_\lambda}\right)^{-1} \mathbf{X}_{\alpha_\lambda}^\top \mathbf{y}. \tag{2.20}$$

   Vektor $\hat{\boldsymbol{\beta}}_{\alpha_\lambda}^{\text{LS}}$ doplnený nulami pre zložky $j \notin \alpha_\lambda$ budeme označovať $\hat{\boldsymbol{\beta}}^{\text{LS}}(\lambda)$. Platí teda $\hat{\boldsymbol{\beta}}_{\alpha_\lambda}^{\text{LS}}(\lambda) = \hat{\boldsymbol{\beta}}_{\alpha_\lambda}^{\text{LS}}$ a $\hat{\boldsymbol{\beta}}_{\overline{\alpha_\lambda}}^{\text{LS}}(\lambda) = \mathbf{0}$.

Dôsledkom penalizácie veľkých koeficientov v LASSO úlohe (2.15) je, že zložky $\hat{\beta}_j^{\text{LASSO}}(\lambda)$ LASSO odhadu sú ako odhady skutočných hodnôt β_j vychýlené smerom k nulovej hodnote. Kvôli tomuto stláčaniu (shrinkage) smerom k menším absolútnym hodnotám sa vektor $\hat{\boldsymbol{\beta}}^{\text{LASSO}}(\lambda)$ často označuje aj ako *shrunk* LASSO odhad. Naopak, pri konštrukcii odhadu $\hat{\boldsymbol{\beta}}^{\text{LS}}(\lambda)$ je pomocou LASSO zvolená len množina aktívnych indexov α_λ. Vektor $\hat{\boldsymbol{\beta}}_{\alpha_\lambda}^{\text{LS}}$ je riešenie LS problému pre submodel α_λ, čo je optimalizačná úloha na voľný extrém v priestore $\mathbb{R}^{|\alpha_\lambda|}$, čím sa eliminuje vychýlenosť (shrinkage). Vektor $\hat{\boldsymbol{\beta}}^{\text{LS}}(\lambda)$ sa preto označuje ako *unshrunk* LASSO odhad. Kvôli väčšej voľnosti (väčšej prípustnej množine) pri výpočte $\hat{\boldsymbol{\beta}}_{\alpha_\lambda}^{\text{LS}}$ má však unshrunk odhad $\hat{\boldsymbol{\beta}}^{\text{LS}}(\lambda)$ vyššiu varianciu než shrunk odhad $\hat{\boldsymbol{\beta}}^{\text{LASSO}}(\lambda)$. Opäť je teda vidno, že bias (výchylku) odhadu možno znížiť len za cenu zvýšenia variancie.

[12]Presnú definíciu pojmu „všeobecná poloha“ možno nájsť v [23] alebo v [11] na s. 19. V prípade $N > p$ je zrejme postačujúcou podmienkou plná hodnosť matice $\mathbf{X}$.

2.4.3 Relaxované LASSO

Shrunk a unshrunk LASSO odhady reprezentujú navzájom opačné póly bias-variance trade-off krivky pre celkovú chybu odhadu vektora $\boldsymbol{\beta}$. Je preto zmysluplné uvažovať o odhade, ktorý by bol kompromisom medzi oboma uvedenými s cieľom priblížiť sa minimu trade-off krivky. Jedným z možných riešení je tzv. *relaxované* LASSO [17], pri ktorom sa odhad $\hat{\boldsymbol{\beta}}^{\text{relax}}(\lambda, \gamma)$ konštruuje ako konvexná kombinácia odhadov $\hat{\boldsymbol{\beta}}^{\text{LASSO}}(\lambda)$ a $\hat{\boldsymbol{\beta}}^{\text{LS}}(\lambda)$. Konkrétne

$$\hat{\boldsymbol{\beta}}^{\text{relax}}(\lambda, \gamma) = \gamma \hat{\boldsymbol{\beta}}^{\text{LASSO}}(\lambda) + (1 - \gamma)\hat{\boldsymbol{\beta}}^{\text{LS}}(\lambda), \tag{2.21}$$

kde $\gamma \in [0, 1]$. Špeciálne, pre hraničné hodnoty $\gamma \in \{0, 1\}$ máme $\hat{\boldsymbol{\beta}}^{\text{relax}}(\lambda, \gamma = 1) = \hat{\boldsymbol{\beta}}^{\text{LASSO}}(\lambda)$ a $\hat{\boldsymbol{\beta}}^{\text{relax}}(\lambda, \gamma = 0) = \hat{\boldsymbol{\beta}}^{\text{LS}}(\lambda)$, teda shrunk resp. unshrunk LASSO odhad. Relaxované LASSO je definované len v prípade, ak stĺpce matice $\mathbf{X}$ sú vo všeobecnej polohe. V opačnom prípade totiž nie je zabezpečená jednoznačnosť odhadov $\hat{\boldsymbol{\beta}}^{\text{LASSO}}(\lambda)$ a $\hat{\boldsymbol{\beta}}^{\text{LS}}(\lambda)$ pre $\forall\, \lambda \in [0, \infty)$.

Nenulové zložky $\hat{\boldsymbol{\beta}}^{\text{LASSO}}_{\alpha_\lambda}(\lambda)$ a $\hat{\boldsymbol{\beta}}^{\text{LS}}_{\alpha_\lambda}(\lambda)$ môžeme vyjadriť pomocou vzťahov (2.18) a (2.20), z čoho po úprave dostávame

$$\begin{aligned} \hat{\boldsymbol{\beta}}^{\text{relax}}_{\alpha_\lambda}(\lambda, \gamma) &= \left(\mathbf{X}^\top_{\alpha_\lambda}\mathbf{X}_{\alpha_\lambda}\right)^{-1}\mathbf{X}^\top_{\alpha_\lambda}\mathbf{y} - \gamma\lambda\left(\mathbf{X}^\top_{\alpha_\lambda}\mathbf{X}_{\alpha_\lambda}\right)^{-1}\mathbf{s} \\ &= \hat{\boldsymbol{\beta}}^{\text{LS}}_{\alpha_\lambda} - \gamma\lambda\left(\mathbf{X}^\top_{\alpha_\lambda}\mathbf{X}_{\alpha_\lambda}\right)^{-1}\mathbf{s}. \end{aligned} \tag{2.22}$$

Keďže $\hat{\boldsymbol{\beta}}^{\text{LASSO}}_{\overline{\alpha_\lambda}}(\lambda) = \hat{\boldsymbol{\beta}}^{\text{LS}}_{\overline{\alpha_\lambda}}(\lambda) = \mathbf{0}$, triviálne tiež platí $\hat{\boldsymbol{\beta}}^{\text{relax}}_{\overline{\alpha_\lambda}}(\lambda, \gamma) = \mathbf{0}$. Výraz $\lambda\left(\mathbf{X}^\top_{\alpha_\lambda}\mathbf{X}_{\alpha_\lambda}\right)^{-1}\mathbf{s} = \hat{\boldsymbol{\beta}}^{\text{LS}}_{\alpha_\lambda}(\lambda) - \hat{\boldsymbol{\beta}}^{\text{LASSO}}_{\alpha_\lambda}(\lambda)$ vyjadruje stlačenie (shrinkage) koeficientov $\hat{\boldsymbol{\beta}}^{\text{LASSO}}_{\alpha_\lambda}(\lambda)$ smerom k nule oproti $\hat{\boldsymbol{\beta}}^{\text{LS}}_{\alpha_\lambda}(\lambda)$. Parameter γ teda možno interpretovať aj ako koeficient regulujúci úroveň stlačenia.

Optimálne hodnoty parametrov λ, γ sa volia z vopred zvolenej diskrétnej množiny (mriežky) tak, aby sa minimalizovala (cross) validačná chyba. Pre každú hodnotu λ sa riešením úlohy (2.15) získa vektor koeficientov $\hat{\boldsymbol{\beta}}^{\text{LASSO}}(\lambda)$, z ktorého sa identifikuje množina aktívnych indexov α_λ. Následne sa pomocou (2.20) vypočíta $\hat{\boldsymbol{\beta}}^{\text{LS}}_{\alpha_\lambda}$ a pomocou (2.22) skonštruujeme $\hat{\boldsymbol{\beta}}^{\text{relax}}_{\alpha_\lambda}(\lambda, \gamma)$ pre niekoľko vopred zvolených hodnôt γ. Pomocou získaných vektorov $\hat{\boldsymbol{\beta}}^{\text{relax}}(\lambda, \gamma)$ nakoniec vypočítame (cross) validačnú chybu pre každú vopred zvolenú kombináciu hodnôt λ, γ a vyberieme tú s najnižšou chybou.

2.5 Stupne voľnosti selekčných algoritmov

Predchádzajúca rozprava mohla vyvolať dojem, že ideálom vo svete selekčných algoritmov je best subset selekcia a zvyšné metódy sú len istou formou heuristickej náhrady, aproximácie z dôvodu výpočtovej náročnosti či rovno nerealizovateľnosti best subset selekcie. Nie je to však tak. Ani v prípade, že by sa podarilo prekonať problém výpočtovej náročnosti, by best subset selekcia nebola vždy tou najlepšou metódou. Dokonca, nebola by najvhodnejším algoritmom zrejme v prevažnej väčšine prípadov.

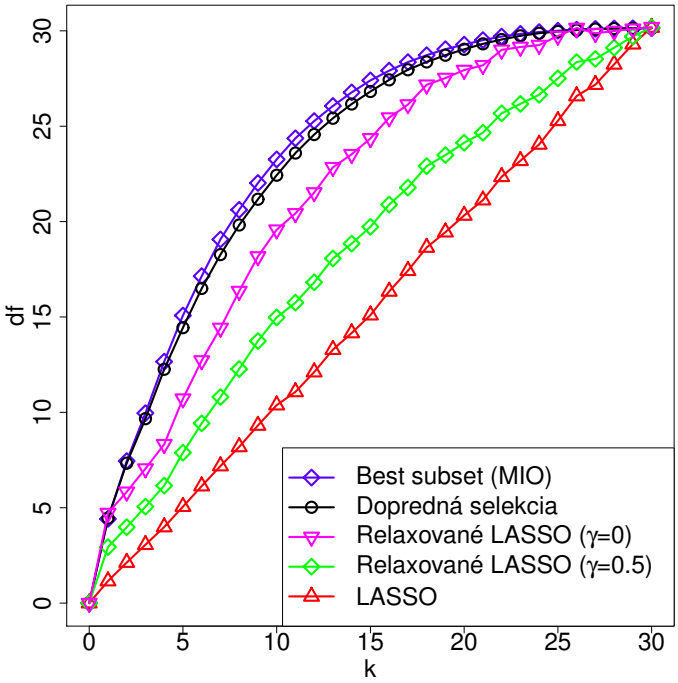
Dôvodom tejto skutočnosti je opäť bias-variance trade-off, konkrétne fenomén preučenia. Pre jeho lepšie vysvetlenie je vhodné definovať počet stupňov voľnosti selekčného algoritmu A (porov. [11], časť 2.5; [10])

$$\mathrm{df}\,(A) = \frac{1}{\sigma^2} \sum_{i=1}^{N} \mathrm{cov}\,(y_i, \hat{y}_i)\,, \tag{2.23}$$

kde $\hat{\mathbf{y}}$ je vektor predikovaných hodnôt na trénovacej množine pre optimálny model vybraný daným selekčným algoritmom. Teraz už máme konkrétnu veličinu, ktorú si môžeme predstaviť pod pojmom *zložitosť* vystupujúcu v trade-off krivke. Veličina $\mathrm{df}\,(A)$ popisuje schopnosť selekčného algoritmu A prispôsobiť (adaptovať) sa trénovacej vzorke dát. Čím lepšie sú totiž predikcie $\hat{\mathbf{y}}$ schopné kopírovať realizáciu $\mathbf{y}$ v trénovacej vzorke, tým vyššie sú kovariancie $\mathrm{cov}\,(y_i, \hat{y}_i)$, a tým vyšší je počet stupňov voľnosti $\mathrm{df}(A)$.

Pre selekčný algoritmus A_k^{LS}, ktorý vždy vráti LS predikcie pre fixný submodel S veľkosti $|S| = k$ možno ukázať, že $\mathrm{df}\left(A_k^{\mathrm{LS}}\right) = k$. Toto je konzistentné s klasickou definíciou počtu stupňov voľnosti, ako počtu nezávislých parametrov definujúcich množinu prípustných riešení. A_k^{LS} hľadá LS odhad $\hat{\boldsymbol{\beta}}_S^{\mathrm{LS}}$ v priestore $\mathbb{R}^k$, preto počet stupňov voľnosti (dimenzia $\mathbb{R}^k$) je k. Pri best subset selekcii či doprednej selekcii je však hodnota $\mathrm{df}\,(A)$ (výrazne) vyššia. Ak aj zafixujeme veľkosť hľadaného submodelu hodnotou k, stále môžeme voliť z $\binom{p}{k}$ rôznych kombinácií premenných v modeli S. Množina prípustných vektorov $\boldsymbol{\beta}_S$ má teda výrazne vyššiu dimenziu, čo sa odrazí na vyššej adaptabilite k trénovacím dátam, a teda aj hodnote $\mathrm{df}\,(A)$.

Pre shrunk LASSO sa dá dokázať [27, 22], že pri fixovanej hodnote λ je počet nenulových koeficientov k_λ vo vektore $\hat{\boldsymbol{\beta}}^{\mathrm{LASSO}}(\lambda)$ nevychýleným odhadom $\mathrm{df}\,(\mathrm{LASSO}(\lambda))$, t. j. $\mathrm{df}\,(\mathrm{LASSO}(\lambda)) = \mathrm{E}\,(k_\lambda)$. Napriek tomu, že LASSO vyberá tiež z veľkého počtu modelov ($\binom{p}{k_\lambda}$), výsledný počet stupňov voľnosti je nízky vďaka ohraničeniu $\sum_{j=1}^{p} |\beta_j| \leq t$.



Obr. 2.4: Príklad závislosti počtu stupňov voľnosti df od použitého selekčného algoritmu a požadovanej veľkosti modelu $k = |S|$, resp. $k = |\alpha_\lambda|$, pre $k = 0, \ldots, p$, kde $p = 30$. Zobrazené sú výsledky pre best subset selekciu (pomocou MIO prístupu), doprednú selekciu, LASSO, relaxované LASSO s $\gamma = 0.5$ a $\gamma = 0$ na základe numerického výpočtu na simulovaných dátach s $N = 70$. Podľa [12], obr. 4. Spracované na základe dát dostupných v [13].

Pri relaxovanom LASSO df narastá s poklesom γ a dosahuje maximum pre unshrunk LASSO ($\gamma = 0$). Aj v tomto maxime je však hodnota df výrazne nižšia než pri best subset selekcii či doprednej selekcii. Dôvodom je, že unshrinkage sa realizuje len na fixovanom submodeli $S = \alpha_\lambda$ danom voľbou λ. Popísané správanie je ilustrované na obrázku 2.4.

Výsledkom uvedeného je, že v situáciách s nízkym počtom trénovacích príkladov N, alebo malou hodnotou signal-to-noise ratio (SNR), sa algoritmy s vysokou adaptabilitou (vysokým df - best subset selekcia, dopredná selekcia) prehnane prispôsobujú konkrétnej realizácii trénovacích dát. Výsledný model je potom zvolený z veľkej časti na základe náhodného šumu, čo vedie k vyššej testovacej chybe než by bolo možné dosiahnuť výberom vhodnejšieho modelu. V týchto situáciách sa preto javí výhodnejšie z hľadiska minimalizácie výslednej testovacej chyby realizovať selekciu premenných pomocou menej adaptívnych algoritmov, ako napr. LASSO, či relaxované LASSO.

Kapitola 3

Numerické výsledky

Prezentované selekčné algoritmy boli implementované v programovacom jazyku R. Branch and bound algoritmus [5] od Furnivala a Wilsona bol realizovaný pomocou funkcie `regsubsets` z balíčka `leaps`. Táto funkcia umožňuje vykonať aj doprednú alebo spätnú selekciu, čo sme však nevyužili[1]. Náhradou bola vlastná funkcia založená na sweepovaní matice $\mathbf{A}$ (1.27) a využití vzťahov (1.65), resp. (1.66). Program selekčného algoritmu založeného na MIO formulácii (1.16) riešenej solverom GUROBI (časť 2.1.2) bol prevzatý z balíčka `bestsubset` [13][2], bol však mierne upravený tak, aby jeho výstupom bol aj dolný odhad účelovej funkcie, t. j. optimálnych hodnôt $RSS_{S^{(k)}}$, $k = 1, \ldots, p$. Z balíčka `bestsubset` bola prevzatá aj funkcia implementujúca relaxované LASSO. Klasický „nerelaxovaný“ LASSO algoritmus bol však realizovaný prostredníctvom funkcie `glmnet` z balíčka `glmnet`, pretože s objektom získaným ako výstup funkcie `glmnet` možno ľahšie vykonať doplnkové analýzy výsledku. Validačná dopredná selekcia, VS.KL a validačný VS.KL algoritmus sú nové selekčné metódy navrhnuté v rámci tejto práce, preto museli byť implementované vlastnými funkciami.

[1] Jedným z dôvodov bolo, že funkcia `regsubsets` vyžaduje ako vstup maticu plánu $\mathbf{X}$ plnej hodnosti. V praxi tak bolo nutné vykonať predselekciu premenných pomocou QR-rozkladu matice $\mathbf{X}$. Okrem toho, submodely $\left\{S^{(k)}\right\}_{k=1}^{p}$ získané ako výstup funkcie `regsubsets` pri doprednej alebo spätnej selekcii neboli vždy do seba vnorené, t. j. nebolo vždy splnené $S^{(k)} \subset S^{(k+1)}$, $k = 1, \ldots, p-1$. Túto charakteristickú vlastnosť sa nepodarilo zabezpečiť ani voľbou parametra *nested* = TRUE, ktorý by mal garantovať, že funkcia `regsubsets` vykonáva doprednú/spätnú salekciu v súlade s popisom algoritmu v časti 2.2.

[2] Balíček [13] je interaktívnym doplnkom článku [12] umožňujúcim jednoducho reprodukovať v článku prezentované výsledky.

Všetky vlastné implementácie algoritmov využívajú formalizmus sweepovania v jeho symetrizovanej forme.

Časovo náročné výpočty (MIO, branch and bound, VS.KL, validačná verzia VS.KL) boli uskutočnené na výkonnom počítači s 24 procesormi Intel Xeon X5670 (2.93 GHz), 128 GB RAM a 64-bitovým OS (macOS 10.13.4). Iba v prípade MIO formulácie bol výpočet paralelizovaný (vďaka vnútornej implementácii solveru GUROBI). Na tomto počítači bol jazyk R nainštalovaný vo verzii 3.4.3 a solver GUROBI vo verzii 7.5.2 (oba 64-bit). Jednoduchšie výpočty trvajúce rádovo sekundy (dopredná selekcia a jej validačná verzia, LASSO a relaxované LASSO) boli vykonané na počítači s dvojjadrovým procesorom Intel Core 2 Duo P8400 (2.26 GHz), 3 GB RAM, 32-bitovým OS (Ubuntu 12.04) a jazykom R nainštalovaným vo verzii 3.2.2 (32-bit). Všetky vlastné funkcie boli pred použitím skompilované pomocou funkcie `cmpfun` z balíčka `compiler`.

V každej úlohe bol najprv každý zo stĺpcov matice plánu $\mathbf{X}$ ako aj výstupný vektor $\mathbf{y}$ štandardizovaný tak, aby v rámci trénovacej množiny mal nulový priemer a jednotkovú štandardnú odchýlku. Vycentrovaním premenných odstránime intercept, ako bolo diskutované v časti 1.1. V časti 2.4 sme stručne zdôvodnili naškálovanie všetkých premenných na (aspoň približne) rovnaký rozsah ako nutnú podmienku úspechu penalizačných metód (ridge regression, LASSO). Rovnaká škála premenných je však dôležitá aj pre numerickú stabilitu procesu sweepovania, a tým aj všetkých na ňom založených algoritmov (VS.KL, dopredná selekcia a ich validačné verzie). Cieľom je dosiahnuť, aby prvky matice $\mathbf{A}$ (1.27) boli rádovo rovnaké, a tým sa potlačili numerické chyby pri aritmetických operáciách s nimi pri operácii PPT. Z rovnakého dôvodu je žiaduce, aby boli rádovo rovnaké aj prvky transformovaných matíc $\mathrm{ppt}\,(\mathbf{A}, \alpha)$. Nech prvky a_{ij} matice $\mathbf{A}$ sú rovnakého rádu d, t. j. $|a_{ij}| \sim 10^d$. Potom v hrubom priblížení platí, že prvky $(\mathbf{A}_{\alpha,\alpha})^{-1}$ sú rádu $-d$ a prvky $\mathbf{A}_{\overline{\alpha},\alpha} (\mathbf{A}_{\alpha,\alpha})^{-1} \mathbf{A}_{\alpha,\overline{\alpha}}$ sú rádu d. Rádovo rovnaké prvky v maticiach $\mathrm{ppt}\,(\mathbf{A}, \alpha)$ teda dosiahneme, ak $-d = d$, resp. $d = 0$. Preto by prvky počiatočnej matice $\mathbf{A}$ mali byť rádovo jednotky. Pri vyššie uvedenej štandardizácii budú prvky $a_{jj} = \mathbf{x}_j^\top \mathbf{x}_j$, $j = 1, \ldots, p$, resp. $a_{p+1,p+1} = \mathbf{y}^\top \mathbf{y}$ na diagonále matice $\mathbf{A}$ nadobúdať hodnoty $N_{\mathrm{Train}} - 1$. V úlohe z časti 3.1 máme $N_{\mathrm{Train}} \sim 10^5 \gg 1$. Vyskúšaním niekoľkých možností sa ukázalo, že rádový rozsah prvkov matíc $\mathrm{ppt}\,(\mathbf{A}, \alpha)$ bol najmenší (v rámci uvažovanej množiny možností), ak počiatočná matica $\mathbf{A}$ bola transformovaná podľa

vzťahu $\mathbf{A} \mapsto \mathbf{A}/\sqrt{N_{\text{Train}}}$. Táto transformácia bola použitá vždy, ak bolo potrebné využiť procedúru sweepovania. Získané výsledky boli však spätne normované[3] tak, aby zodpovedali stĺpcom $\mathbf{x}_j$ a výstupu $\mathbf{y}$ s nulovým priemerom a jednotkovou štandardnou odchýlkou na trénovacích dátach.

Ukázalo sa tiež, že hoci pri PPT transformácii $\text{ppt}(\mathbf{B}, \alpha)$ matice $\mathbf{B} = \text{ppt}(\mathbf{A}, S)$, $S \in \langle p \rangle$ explicitne konštruujeme inverziu $(\mathbf{B}_{\alpha,\alpha})^{-1}$, nie je vhodné ju priamo použiť na výpočet zvyšných blokov matice $\text{ppt}(\mathbf{B}, \alpha)$. Vo všeobecnosti výrazne presnejší výsledok získame, ak napríklad blok $(\mathbf{B}_{\alpha,\alpha})^{-1}\mathbf{B}_{\alpha,\overline{\alpha}}$ nebudeme počítať vynásobením matice $\mathbf{B}_{\alpha,\overline{\alpha}}$ inverziou $(\mathbf{B}_{\alpha,\alpha})^{-1}$ zľava, ale riešením sústavy

$$\mathbf{B}_{\alpha,\alpha}[\mathbf{C}, \mathbf{D}] = \left[\mathbf{I}_{|\alpha|}, \mathbf{B}_{\alpha,\overline{\alpha}}\right]. \tag{3.1}$$

Ako výsledok dostaneme $\mathbf{C} = (\mathbf{B}_{\alpha,\alpha})^{-1}$ a $\mathbf{D} = (\mathbf{B}_{\alpha,\alpha})^{-1}\mathbf{B}_{\alpha,\overline{\alpha}}$. Dôvod výrazného zlepšenia presnosti výsledku spočíva zrejme v tom, že pri riešení sústavy (3.1) sa vytvára vhodná faktorizácia matice $(\mathbf{B}_{\alpha,\alpha})^{-1}$, pričom postupné násobenie matice $\mathbf{B}_{\alpha,\overline{\alpha}}$ vzniknutými faktormi je numericky presnejšie než priame vynásobenie maticou $(\mathbf{B}_{\alpha,\alpha})^{-1}$.

Správnosť výsledkov získaných sweepovaním bola po vyššie uvedených úpravách skontrolovaná porovnaním s výstupom funkcie `lm`, ktorá je štandardom pre výpočet LS odhadov veličín v LRM (napr. $\hat{\boldsymbol{\beta}}^{\text{LS}}$, RSS). Pre dáta z časti 3.1 boli relatívne odchýlky medzi výsledkami funkcie `lm` a algoritmov založených na procedúre sweepovania nasledovné[4]: v prípade prvkov vektora $\hat{\boldsymbol{\beta}}_S^{\text{LS}}$ na úrovni $\sim 10^{-6} - 10^{-8}$, v prípade $RSS_S = \text{ppt}[(\mathbf{A}, S)]_{p+1,p+1}$ na úrovni $\sim 10^{-4} - 10^{-5}$ pre takmer všetky submodely $S \in \mathcal{S}$. Pokiaľ sa však RSS_S vypočítalo dosadením $\hat{\boldsymbol{\beta}}_S^{\text{LS}}$ do kvadratickej formy $||\mathbf{y} - \mathbf{X}_S\boldsymbol{\beta}_S||_2^2$, relatívna odchýlka klesla na úroveň $\sim 10^{-7} - 10^{-8}$. Túto skutočnosť bude v budúcnosti potrebné reflektovať vhodnou úpravou numerickej realizácie sweepovacej procedúry. Hodnoty RSS_S prezentované v tejto práci boli získané presnejším variantom výpočtu pomocou dosadenia $\hat{\boldsymbol{\beta}}_S^{\text{LS}}$ do kvadratickej formy $||\mathbf{y} - \mathbf{X}_S\boldsymbol{\beta}_S||_2^2$.

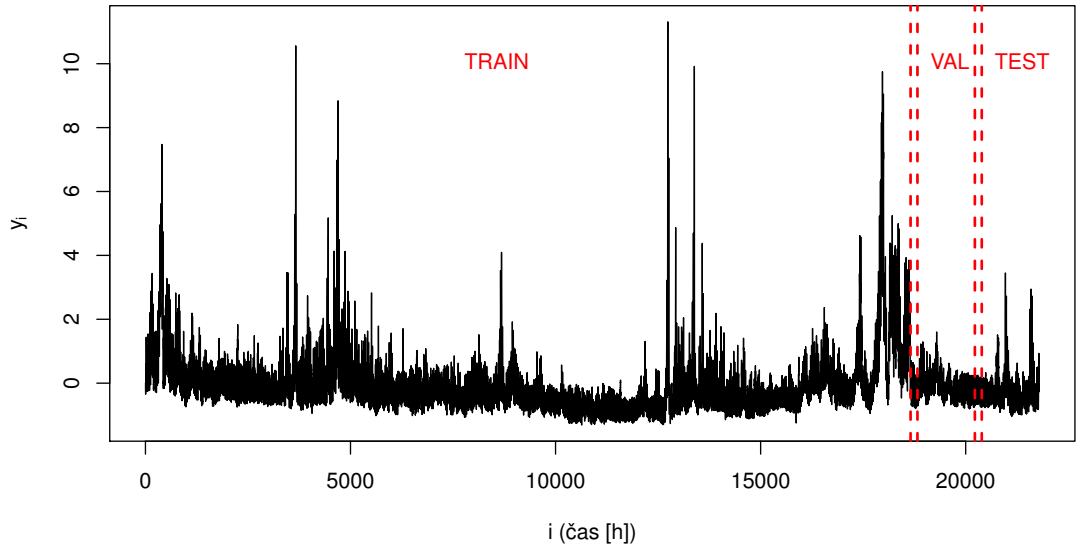
[3]Ľahko sa ukáže, že pri spätnej transformácii sa regresné koeficienty $\hat{\boldsymbol{\beta}}$ nemenia a $RSS_S = \sqrt{N_{\text{Train}}}\,[\text{ppt}(\mathbf{A}, S)]_{p+1,p+1}$.

[4]Prezentované sú výsledky pre prípad, kedy neplná hodnosť matice plánu $\mathbf{X}$ bola naprávaná predselekciou premenných pomocou pivotizácie z QR-rozkladu a pripočítaním $\lambda = 10^{-6}$ k diagonále počiatočnej matice $\mathbf{A}$. Pritom hodnota diagonálnych prvkov bola $N_{\text{Train}} \approx 10^5$. Pripočítanie $\lambda = 10^{-3}$ zhoršilo relatívne odchýlky o $3 - 4$ rády.

Branch and bound algoritmus je pre modely S_{FULL} s relatívne malým počtom prediktorov ($p < 50$, za istých podmienok $p < 100$) schopný počas rádovo hodín nájsť globálne optimálne submodely $S_k = \underset{S \in \mathcal{S}_k}{\operatorname{argmin}} RSS_S$, $k = 1, \ldots, p$. Túto vlastnosť sme využili pre ohodnotenie schopnosti VS.KL algoritmu nájsť globálne optimálne submodely S_k, alebo ich aspoň dostatočne dobre aproximovať z hľadiska hodnoty RSS_S. Vytvárali sme umelo simulované dáta s $p \approx 50$ prediktormi a $N \approx 10^3$ pozorovaniami a porovnávali sme výsledky VS.KL a branch and bound. VS.KL takmer vždy našiel optimálne submodely S_k získané branch and bound algoritmom, alebo sa RSS_S ním zvolených modelov od RSS_{S_k} líšil zanedbateľne málo. Pritom výpočet pomocou VS.KL stačilo mať spustený len niekoľko minút až desiatok minút. Avšak, narozdiel od branch and bound, VS.KL nevie garantovať optimalitu svojich výsledkov. Diskusia z časti 2.5 ako aj praktické výsledky v 3.1 ozrejmujú, prečo „posadnutosť“ hľadaním submodelov S_k minimalizujúcich RSS_S v rámci množiny $\mathcal{S}_k$ nie je z hľadiska minimalizácie predikčnej chyby modelu relevantná. Vzhľadom na to, ako už aj tak značnú rozsiahlosť predkladanej práce, neuvádzame detailné výsledky porovnania VS.KL a branch and bound algoritmu.

3.1 Dáta a použité parametre algoritmov

Reálne dáta pre účely porovnania selekčných algoritmov boli poskytnuté firmou Tangent works. Predstavujú časový rad vývoja ceny elektrickej energie s diskretizačným krokom 1 hodina. Okrem dátumu, času a ceny energie v tomto čase nemáme k dispozícii žiadne iné údaje. Cieľom je predikovať cenu energie na deň dopredu (t. j. o 24 hodín). Pre tento účel bol vytvorený autoregresný model z predchádzajúcich cien energie až o 168 hodín (týždeň) dozadu, ich rôznych transformácií, diferencií, kombinácií a dummy premenných označujúcich napr. víkendy a sviatky. S bližšími detailami procesu generovania sady vysvetľujúcich premenných v autoregresnom modeli sme, bohužiaľ, kvôli firemnému tajomstvu neboli oboznámení. Celkovo bolo takto vytvorených $p = 431$ premenných + intercept, pričom však hodnosť matice plánu $\mathbf{X}$ bola len 383. Počet pozorovaní bol $N = 21792$. Každý zo stĺpcov $\mathbf{X}$ ako aj vektor $\mathbf{y}$ boli štandardizované tak, aby v rámci trénovacej množiny mali nulový priemer a jednotkovú štandardnú



Obr. 3.1: Časový rad $\{y_i\}_{i=1}^N$ štandardizovanej výstupnej premennej. Červenými zvislými čiarkovaným čiarami je znázornené rozdelenie dát do trénovacej (train), validačnej (val) a testovacej (test) množiny. Tieto množiny sú súvislé bloky dát dĺžky $N_{\text{Train}} = 18656$ a $N_{\text{Val}} = N_{\text{Test}} = 1400$, pričom medzi blokmi sú medzery zodpovedajúce 168 pozorovaniam.

odchýlku, čím sme eliminovali intercept a zostala matica $\mathbf{X}$ rozmeru $N \times p$ a hodnosti 382. Priemerná cena energie pred štandardizáciou bola 48.50 a jej štandardná odchýlka 26.25. Časový rad štandardizovanej premennej $\{y_i\}_{i=1}^N$ je znázornený na obr. 3.1.

Cena energie y_i v časovom okamihu i je modelovaná autoregresne pomocou predchádzajúcich hodnôt až do rádu y_{i-168}. Aby nevznikol informačný prekryv medzi trénovacou, validačnou a testovacou množinou, nie je vhodné rozdeliť pozorovania do týchto množín náhodným výberom. Preto tieto množiny boli zvolené ako súvislé, po sebe idúce bloky dát, ale s minimálnym odstupom 168 pozorovaní medzi po sebe nasledujúcimi blokmi. Do trénovacej množiny bolo zaradených prvých cca 85% ($N_{\text{Train}} = 18656$) dát a do validačnej a testovacej množiny zhodne po $N_{\text{Val}} = N_{\text{Test}} = 1400$ pozorovaniach v súlade s obr. 3.1. Na validáciu a testovanie tak bolo vyčlenených celkovo menej než 15% z celkového objemu dát. Hlavným dôvodom tohto spôsobu rozdelenia bola snaha vyhnúť sa v rámci validácie aj testovania úsekom s prudkými zmenami ceny energie y_i. Kvalita predikcií natrénovaného modelu je v týchto úsekoch výrazne znížená, preto by výber optimálneho submodelu na základe validačnej chyby, či odhad predikčnej chyby na základe testovacej množiny mohol byť výrazne vychýlený.

Neúplná hodnosť matice plánu $\mathbf{X}$ bola riešená výberom pivotizačných premenných z QR rozkladu a pripočítaním $\lambda = 10^{-6}$ k diagonále matice $\mathbf{A}$. Uvažované boli submodely veľkosti $k \in \{4, \ldots, 350\}$ úplného modelu S_{FULL} veľkosti $|S_{\text{FULL}}| = 382$. Solver GUROBI použitý na riešenie selekčnej úlohy v tvare MIO formulácie mal nastavený časový limit $t_{\max} = 500s$ pre každú veľkosť submodelu k. Pre submodely s veľkosťami $k \in \{4, \ldots, 200\}$ prebiehal výpočet paralelizovane priemerne na $10 - 12$ procesoroch, pri $k \in \{201, \ldots, 350\}$ priemerne na 20 procesoroch. Pri oboch verziách algoritmu VS.KL boli nastavené parametre $K_1 = 5, K_2 = 20, L_1 = 10, L_2 = 20$ ako konštanty a $m_{11} = m_{12} = \min(\lceil k/3 \rceil, \lfloor 0.39k \rfloor)$, $m_{13} = \min(\max(k - m_{11} - m_{12}, 0), \max(\lceil k/8 \rceil, 5))$, $m_{21} = m_{22} = \min(\lceil 0.43k \rceil, \lfloor 0.499k \rfloor)$ sa menili v závislosti od veľkosti submodelu k. Blok dvoch volaní funkcie VS.KL s náhodným počiatočným modelom S_0 sa spúšťal pre každé k. Pri štandardnom VS.KL algoritme bolo pre $k \in \{4, \ldots, 200\}$ použité $t^1_{\max} = t^2_{\max} = 150s$, $t^3_{\max} = t^4_{\max} = 100s$ a pre $k \in \{201, \ldots, 350\}$ bolo nastavené $t^1_{\max} = 200s$, $t^2_{\max} = 250s$, $t^3_{\max} = t^4_{\max} = 150s$. Celková dĺžka výpočtu $\sum_{i=1}^{4} t^i_{\max}$ VS.KL algoritmu pre fixovanú hodnotu k je teda len mierne vyššia než limit $t_{\max} = 500s$ pri riešení MIO úlohy. Vzhľadom na minimálne 10-násobnú paralelizáciu MIO výpočtov, boli ale výsledky VS.KL algoritmu získané pri približne desatinovej výpočtovej záťaži oproti MIO úlohe. Pri validačnom VS.KL algoritme boli časové limity rovnaké pre celý uvažovaný rozsah k, konkrétne $t^1_{\max} = t^2_{\max} = 200s$, $t^3_{\max} = t^4_{\max} = 150s$. Doba behu zvyšných algoritmov bola rádovo sekundy, maximálne desiatky sekúnd.

3.1.1 Nepenalizačné algoritmy

Priebeh trénovacej chyby RSS_S/N_{Train} pre postupnosť submodelov $\{S^{(k)}\}_{k=4}^{350}$ zvolených jednotlivými algoritmami ako optimálne v rámci množiny $\mathcal{S}_k$ submodelov veľkosti k je zobrazený na obr. 3.2. V celom rozsahu $k = 30, \ldots, 350$ boli najnižšie hodnoty trénovacej chyby dosiahnuté pomocou VS.KL algoritmu, mierne horšie výsledky priniesol MIO prístup nasledovaný s väčším odstupom doprednou selekciou. Dolný odhad skutočného globálneho minima chýb $\{\min RSS_S/N_{\text{Train}} | S \in \mathcal{S}_k\}_{k=4}^{350}$ získaný pomocou solvera GUROBI je takmer v celom rozsahu k prakticky nevyužiteľný, lebo výrazne podhodnocuje polohu minima. Krivka globálnych miním trénovacej chyby bude totiž zrejme ležať tesne pod krivkou získanou pomocou VS.KL. S rastom veľkosti submodelu k sa krivky

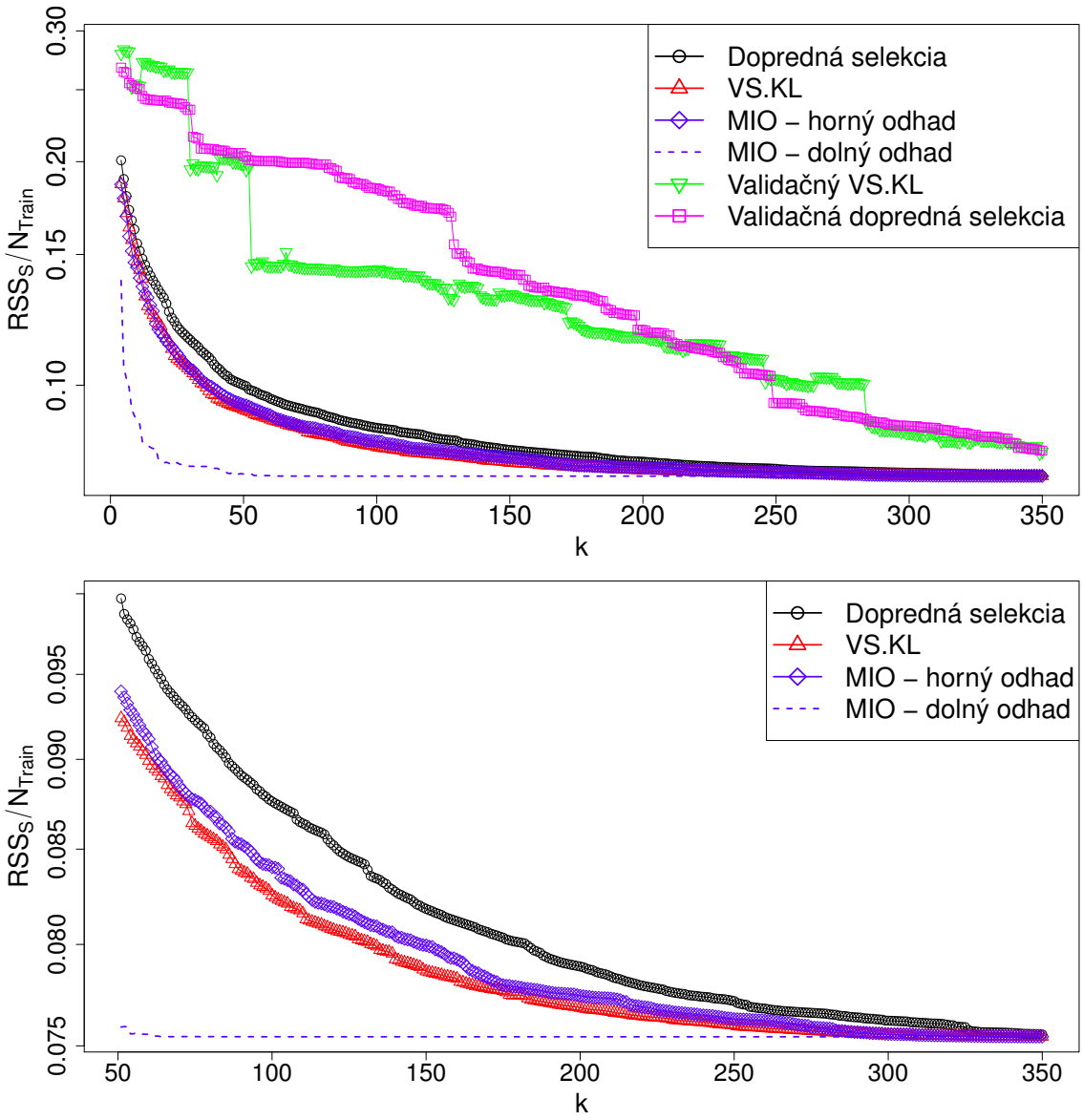
pre VS.KL, MIO a doprednú selekciu (DS) postupne približujú (obr. 3.3), usporiadanie $RSS^{VS.KL}_{S^{(k)}} < RSS^{MIO}_{S^{(k)}} < RSS^{DS}_{S^{(k)}}$ sa však zachováva. Hodnoty $\{RSS_{S^{(k)}}\}_{k=4}^{350}$ sú v prípade validačných verzií algoritmov doprednej selekcie a VS.KL výrazne vyššie a vo všeobecnosti nie monotónne nerastúce, pretože v týchto prípadoch sa submodely $S^{(k)}$ vyberajú minimalizáciou validačnej a nie trénovacej chyby.

Obr. 3.4 a 3.5 ilustrujú priebeh informačných kritérií BIC a AIC vypočítaných na základe trénovacej chyby $RSS_{S^{(k)}}$ submodelov $\left\{S^{(k)}\right\}_{k=4}^{350}$ zvolených ako optimálne algoritmami VS.KL, MIO a doprednou selekciou. Jasne je vidno, že veľkosť k modelu $S^{(k)}$ minimalizujúceho AIC je pre všetky tri selekčné algoritmy výrazne posunutá smerom k vyšším k. To ilustruje, ako AIC bežne nadhodnocuje veľkosť modelu. BIC vyberá výrazne menšie modely, s počtom premenných menším zhruba o 100 než pri AIC. Ako sa ukáže nižšie z analýzy validačnej a testovacej chyby submodelov, menšie modely vyberané pomocou BIC sú z hľadiska predikcie lepšie, hoci sú stále veľkostne nadhodnotené oproti modelom s najnižším odhadom predikčnej chyby. Pre AIC aj BIC platí, že veľkosti optimálnych modelov k^* sú v závislosti od použitého selekčného algoritmu usporiadané v poradí $k^*_{VS.KL} < k^*_{MIO} < k^*_{DS}$. Vyššie uvedené platí, aj ak uvážime varianciu polohy minima krivky BIC/AIC z dôvodu, že hodnota IC je náhodná veličina, ako aj kvôli tomu, že $RSS_{S^{(k)}}$ získaná selekčným algoritmom je len horným odhadom skutočného minima trénovacej chyby $\{\min RSS_S/N_{\text{Train}} | S \in \mathcal{S}_k\}$, $k = 4, \ldots, 350$.

Priebeh validačnej chyby $MSE_{\text{Val}} = ||\mathbf{y}^{\text{Val}} - \mathbf{X}^{\text{Val}}\hat{\boldsymbol{\beta}}^{\text{LS}}_{S^{(k)}}||_2^2/N_{\text{Val}}$ pre postupnosti modelov $\left\{S^{(k)}\right\}_{k=4}^{350}$ získané pomocou nepenalizačných selekčných algoritmov je znázornený na obr. 3.6. Validačná množina bola rozdelená na 10 rovnako veľkých častí (so 140 pozorovaniami), pričom pre každú z týchto častí bola vypočítaná hodnota validačnej chyby s použitím LS odhadov regresných koeficientov $\hat{\boldsymbol{\beta}}^{\text{LS}}_{S^{(k)}}$[5] priradených submodelom $S^{(k)}$. Priemer získaných 10 hodnôt (hodnota $MSE_{\text{Val}}(S^{(k)})$) a ich štandardná odchýlka $\Delta MSE_{\text{Val}}(S^{(k)})$ sú prezentované vo forme krivky $\left\{MSE_{\text{Val}}(S^{(k)})\right\}_{k=4}^{350}$ a pásu spoľahlivosti s hranicami $MSE_{\text{Val}}(S^{(k)}) \pm \Delta MSE_{\text{Val}}(S^{(k)})$.

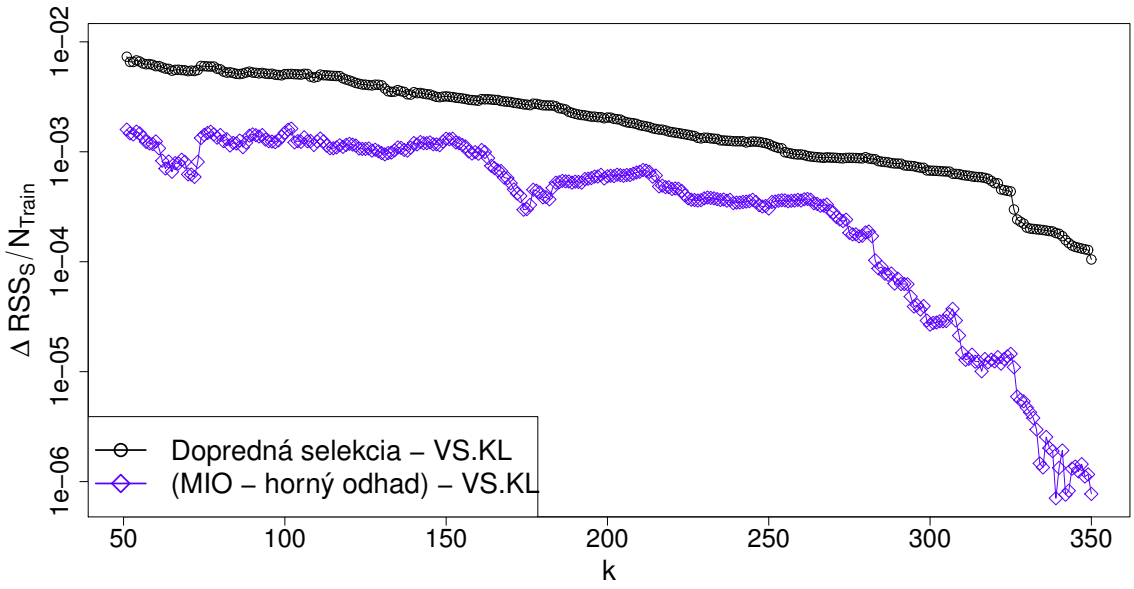
Vzhľadom na veľmi široké pásy spoľahlivosti nepozorujeme signifikantný rozdiel medzi krivkami validačnej chyby v prípade doprednej selekcie, VS.KL a MIO. Značná

[5]Pri všetkých uvažovaných nepenalizačných selekčných algoritmoch (dopredná selekcia, VS.KL, ich validačné verzie a MIO) je submodelu S priradený LS-odhad vektora $\hat{\boldsymbol{\beta}}^{\text{LS}}_S$ v LRM s maticou plánu $\mathbf{X}^{\text{Train}}_S$ a výstupom $\mathbf{y}^{\text{Train}}$, čo zdôrazňujeme znakom $\hat{}$ a označením LS v symbole $\hat{\boldsymbol{\beta}}^{\text{LS}}_S$.

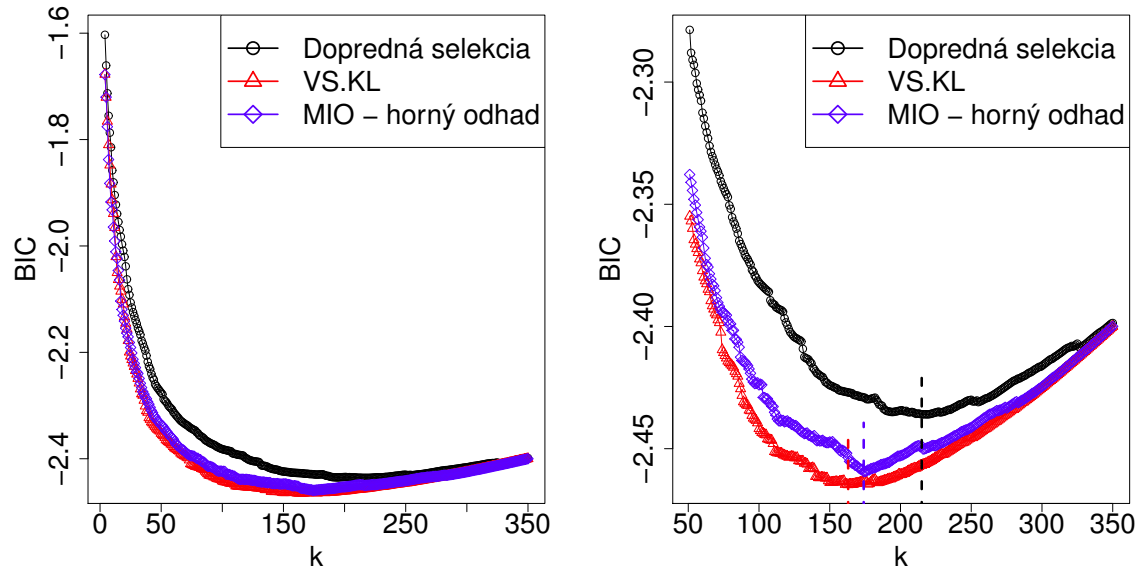


Obr. 3.2: Horný graf: priebeh trénovacej chyby $RSS_{S^{(k)}}/N_{\text{Train}}$ pre postupnosti optimálnych modelov $\left\{S^{(k)}\right\}_{k=4}^{350}$ získané algoritmami VS.KL, doprednou selekciou, ich validačnými verziami a riešením MIO úlohy (1.16) pomocou solvera GUROBI (*MIO - horný odhad*). Krivka *MIO-dolný odhad* reprezentuje spodné ohraničenie pre skutočne optimálne hodnoty $\{\min RSS_S/N_{\text{Train}} | S \in \mathcal{S}_k\}_{k=4}^{350}$ poskytnuté solverom GUROBI pri riešení MIO úlohy. Dolný graf: detail priebehu trénovacej chyby pre $k \in \{51, \ldots, 350\}$.

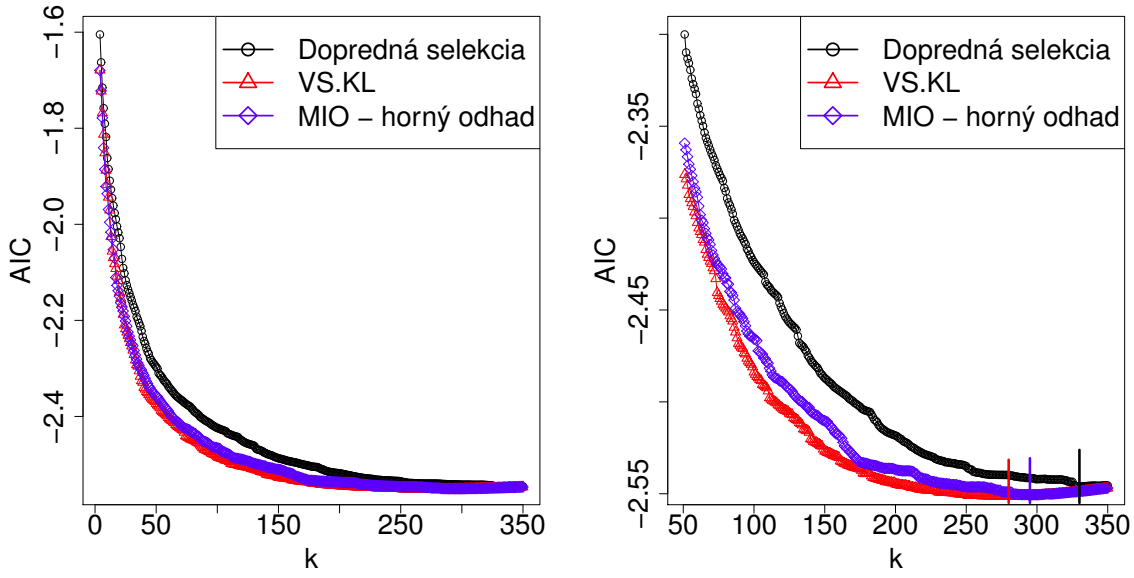
neistota určenia validačnej chyby vychádzajúca jednak z „roztraseného“ priebehu jej krivky, jednak z veľkej šírky pásov spoľahlivosti, nás vedie k opatrnosti pri výbere optimálneho submodelu $S^{(k^*)}$ z postupnosti $\left\{S^{(k)}\right\}_{k=4}^{350}$ len na základe minimalizácie validačnej chyby. Na druhej strane, takto získané výsledky sa pri pohľade na tvar kriviek



Obr. 3.3: Rozdiel $\Delta RSS_{S^{(k)}}/N_{Train} = \left(RSS_{S^{(k)}} - RSS_{S^{(k)}}^{VS.KL}\right)/N_{Train}$ minimálnych hodnôt trénovacej chyby získaných pri MIO formulácii a doprednej selekcii oproti hodnotám vypočítaným pomocou algoritmu VS.KL.



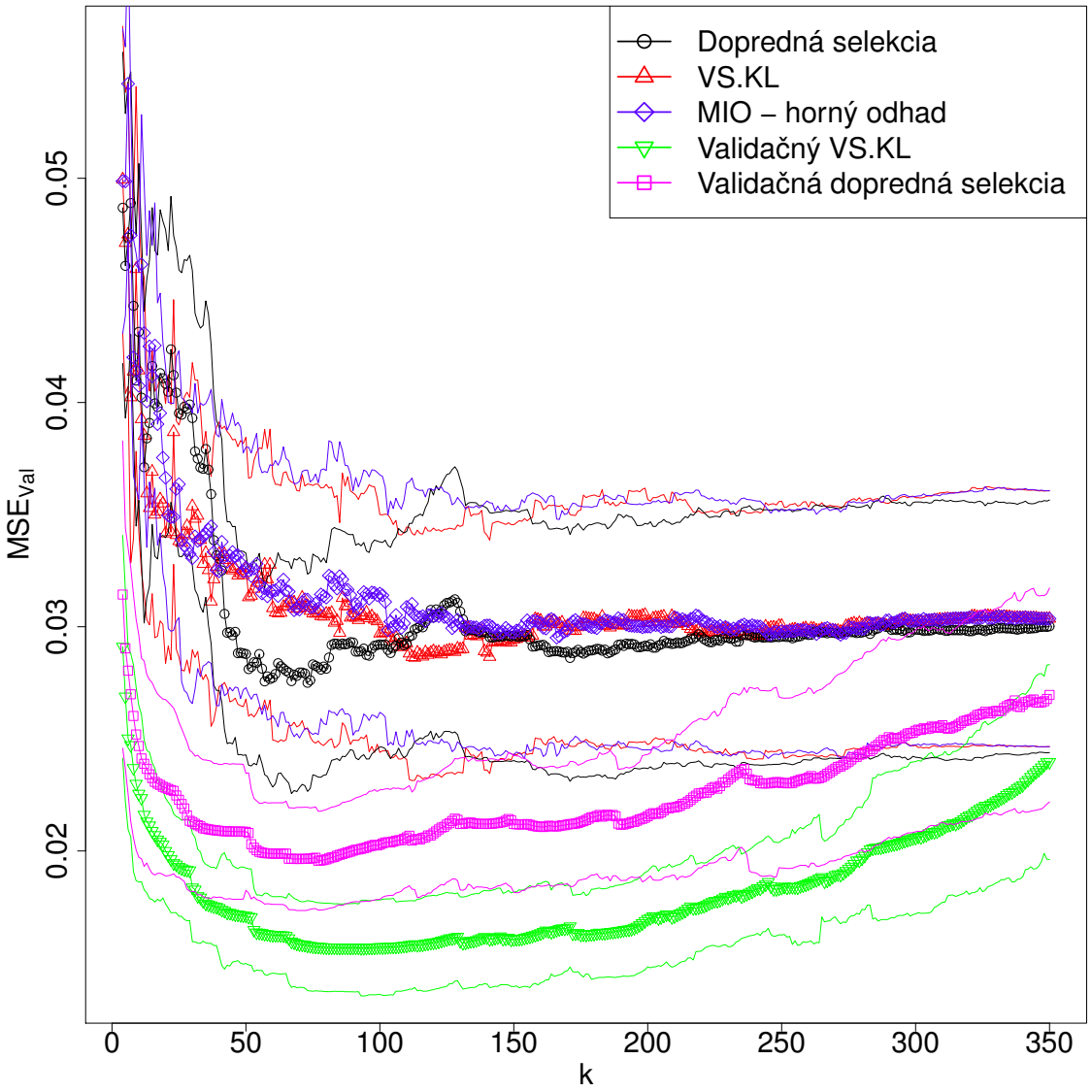
Obr. 3.4: Priebeh BIC na základe trénovacej chyby $RSS_{S^{(k)}}$ pre submodely $\left\{S^{(k)}\right\}_{k=4}^{350}$ zvolené ako optimálne algoritmami VS.KL, MIO a doprednou selekciou. Detail pre $k \in \{51, \ldots, 350\}$ je vpravo. Poloha minima k^* je vyznačená čiarkovanou čiarou, $k^* = 163$ pre VS.KL, $k^* = 174$ pre MIO a $k^* = 215$ pre doprednú selekciu.



Obr. 3.5: Priebeh AIC na základe trénovacej chyby $RSS_{S^{(k)}}$ pre submodely $\left\{S^{(k)}\right\}_{k=4}^{350}$ zvolené ako optimálne algoritmami VS.KL, MIO a doprednou selekciou. Detail pre $k \in \{51, \ldots, 350\}$ je vpravo. Poloha minima k^* je vyznačená zvislou čiarou, $k^* = 280$ pre VS.KL, $k^* = 295$ pre MIO a $k^* = 330$ pre doprednú selekciu.

validačnej chyby zdajú byť v porovnaní s inými alternatívami rozumné. Preto pri doprednej selekcii, VS.KL a MIO volíme $S^{(k^*)}$ z postupnosti $\left\{S^{(k)}\right\}_{k=4}^{350}$ tak, aby minimalizoval chybu $MSE_{\text{Val}}(S^{(k)})$. Počet premenných k^* ako aj validačná chyba $MSE_{\text{Val}}(S^{(k^*)})$ takto vybraných submodelov $S^{(k^*)}$ sú uvedené v tabuľke 3.1.

Je mierne prekvapujúce, že v prípade doprednej selekcie, VS.KL a MIO sa validačná chyba počínajúc zhruba veľkosťou submodelu $k = 100$ stabilizuje. Výrazný pokles chyby v oblasti $k < 50 - 100$ možno prisúdiť postupnému eliminovaniu bias-u tým, ako sa postupným pridávaním premenných zvyšuje komplexnosť modelov $S^{(k)}$. S ďalším pridávaním premenných (rastom k) by validačná chyba v dôsledku nárastu variancie mala opäť začať rásť, hoci len pozvoľne, podobne ako vidno pri validačných verziách doprednej selekcie a VS.KL algoritmu. Pozorovaná stabilizácia (absencia nárastu variančnej zložky chyby) by mohla mať pôvod v relatívne veľkom počte pozorovaní N_{Train} v trénovacej množine, ktorá navyše obsahuje viacero oblastí s výrazne odlišným charakterom vývoja výstupu y (viď obr. 3.1). Submodely $\left\{S^{(k)}\right\}_{k=100}^{350}$ vybrané optimalizáciou trénovacej chyby (RSS) sú pravdepodobne ekvivalentné z hľadiska schopnosti simulovať vplyv nenáhodnej zložky zdrojového modelu (potlačiť bias), no ich relatívne nízka



Obr. 3.6: Validačná chyba $MSE_{\text{Val}}(S^{(k)})$ a jej pás spoľahlivosti $MSE_{\text{Val}}(S^{(k)}) \pm \Delta MSE_{\text{Val}}(S^{(k)})$ pre postupnosti submodelov $\left\{S^{(k)}\right\}_{k=4}^{350}$ získané pomocou nepenalizačných selekčných algoritmov. Polohy a hodnoty miním kriviek validačnej chyby sú uvedené v tabuľke 3.1.

zložitosť im zrejme nedovoľuje výraznejšie sa prispôsobiť konkrétnej realizácii náhodnej zložky (šumu) v dátach. V širokom rozsahu veľkostí submodelov $k = 100, \ldots, 350$ tak zrejme nedochádza k výraznejšiemu preučeniu. Výsledkom je široké minimum v priebehu validačnej chyby, ktoré sa javí ako stabilizované plató, pretože k opätovnému nárastu chyby dochádza až pre veľkosti k mimo zobrazeného rozsahu $k \leq 350$. Minimum validačnej chyby (~ 0.03) je výrazne nižšie než sme pozorovali pri trénovacej

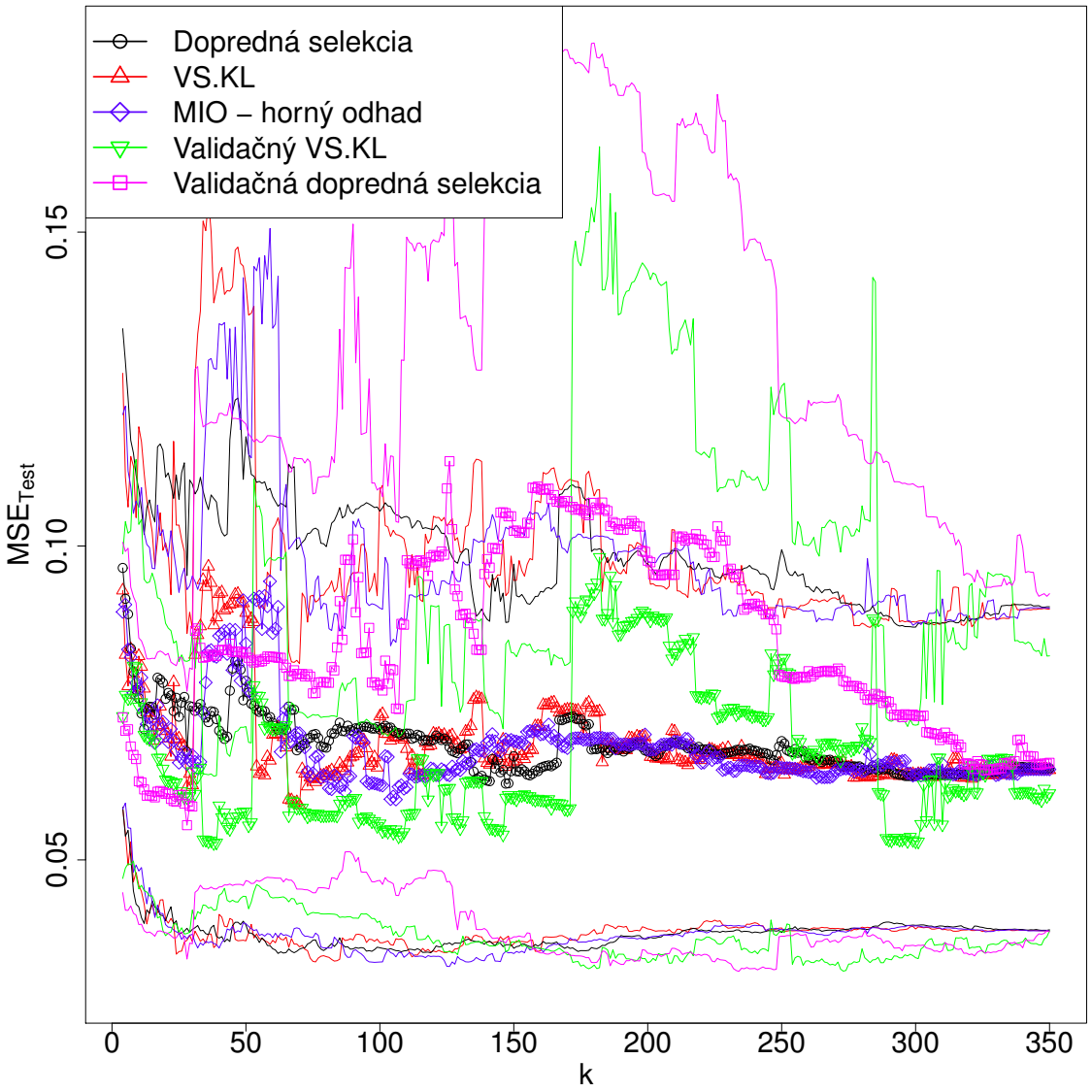
Tabuľka 3.1: Počet premenných k^*_{Val} v submodeli $S^{(k^*_{\text{Val}})}$ minimalizujúcom validačnú chybu, zodpovedajúce minimum $MSE_{\text{Val}}(S^{(k^*_{\text{Val}})})$ a testovacia chyba $MSE_{\text{Test}}(S^{(k^*_{\text{Val}})})$ pre tento submodel, počet premenných k^*_{Test} v submodeli $S^{(k^*_{\text{Test}})}$ minimalizujúcom testovaciu chybu a zodpovedajúce minimum $MSE_{\text{Test}}(S^{(k^*_{\text{Test}})})$ pre nepenalizačné selekčné algoritmy v súlade s obr. 3.6 a 3.7 (DS - dopredná selekcia, val. - validačný).

algoritmus	k^*_{Val}	$MSE_{\text{Val}}(S^{(k^*_{\text{Val}})})$	$MSE_{\text{Test}}(S^{(k^*_{\text{Val}})})$	k^*_{Test}	$MSE_{\text{Test}}(S^{(k^*_{\text{Test}})})$
DS	73	0.027500	0.069292	147	0.062148
VS.KL	113	0.028619	0.064328	70	0.058707
MIO	166	0.029514	0.068952	106	0.059616
val. VS.KL	93	0.015667	0.056707	38	0.052657
val. DS	77	0.019565	0.078271	28	0.055535

chybe (~ 0.075). Dôvodom je, že vývoj výstupu y je vo validačnej oblasti značne stabilizovaný, bez výraznejších výkyvov v porovnaní s trénovacou oblasťou.

Pri validačných verziách VS.KL a doprednej selekcie sa dosahuje výrazne nižšia validačná chyba v celom rozsahu $k = 4, \ldots, 350$, pričom aj medzi výsledkami oboch algoritmov je štatisticky významný rozdiel. Toto je zrejme dôsledok nedostatočnej veľkosti validačnej množiny ($N_{\text{Val}} = 1400$), následkom čoho sa mohli validačné algoritmy príliš prispôsobiť konkrétnej realizácii dát z validačnej množiny, teda overfitovať ich. Oba validačné algoritmy poskytujú teoreticky očakávaný priebeh validačnej chyby s pozvoľným nárastom napravo od minima v dôsledku nárastu variančnej zložky chyby. Validačná chyba $MSE_{\text{Val}}(S^{(k)})$ získaná algoritmom VS.KL by mala aproximovať minimum kvadratickej formy $||\mathbf{y}^{\text{Val}} - \mathbf{X}^{\text{Val}}\boldsymbol{\beta}||_2^2$ v rámci množiny LS-odhadov $\left\{\hat{\boldsymbol{\beta}}^{\text{LS}}_S | S \in \mathcal{S}_k\right\}$ pre LRM konštruovaný na trénovacej množine.

Krivky testovacej chyby $MSE_{\text{Test}} = ||\mathbf{y}^{\text{Test}} - \mathbf{X}^{\text{Test}}\hat{\boldsymbol{\beta}}^{\text{LS}}_{S^{(k)}}||_2^2/N_{\text{Test}}$ pre postupnosti modelov $\left\{S^{(k)}\right\}_{k=4}^{350}$ získané pomocou nepenalizačných selekčných algoritmov sú znázornené na obr. 3.7. Analogicky ako pri validácii bola rozdelením trénovacej množiny na 10 rovnako veľkých častí (so 140 pozorovaniami) vypočítaná štandardná odchýlka hodnôt testovacej chyby $\Delta MSE_{\text{Test}}(S^{(k)})$. Pomocou nej je určený pás spoľahlivosti pre testovaciu chybu s hranicami $MSE_{\text{Test}}(S^{(k)}) \pm \Delta MSE_{\text{Test}}(S^{(k)})$. Je zaujímavé, že spodná



Obr. 3.7: Testovacia chyba $MSE_{\text{Test}}(S^{(k)})$ a jej pás spoľahlivosti $MSE_{\text{Test}}(S^{(k)}) \pm \Delta MSE_{\text{Test}}(S^{(k)})$ pre postupnosti submodelov $\left\{S^{(k)}\right\}_{k=4}^{350}$ získané pomocou nepenalizačných selekčných algoritmov. Polohy a hodnoty miním kriviek validačnej chyby sú uvedené v tabuľke 3.1.

hranica tohto pásu osciluje pri všetkých algoritmoch okolo rovnakej úrovne 0.04.

Medzi krivkami testovacej chyby pre doprednú selekciu, VS.KL a MIO nevidno štatisticky významný rozdiel. Rovnako nie je veľmi významný ani rozdiel medzi testovacími chybami $MSE_{\text{Test}}(S^{(k^*_{\text{Val}})})$ submodelov $S^{(k^*_{\text{Val}})}$ zvolených týmito algoritmami na základe minimalizácie validačnej chyby (viď tabuľka 3.1). Ešte menšie rozdiely by boli pozorované, ak by sa submodely $S^{(k)}$ podarilo zvoliť tak, aby zodpovedali minimu

kriviek $MSE_{\text{Test}}(S^{(k)})$ pre jednotlivé algoritmy. Pre $k \gtrsim 150$ sa testovacie chyby pre doprednú selekciu, VS.KL a MIO stabilizujú v okolí úrovne 0.07, čo možno zdôvodniť rovnako ako v prípade analogického správania pozorovaného pri validácii. Výrazne vyššia úroveň testovacej chyby (0.07) v porovnaní s validačnou chybou (0.03) je spôsobená dynamickejším charakterom správania výstupu y v testovacej oblasti.

Oproti validácii sa výrazne zmenil charakter kriviek pre validačné verzie doprednej selekcie a VS.KL, ktoré teraz namiesto ukážkového „U“ tvaru vykazujú výrazne chaotickejší priebeh. Toto správanie je spôsobené zrejme relatívne malým počtom pozorovaní N_{Val} vo validačnej množine, čo vedie k overfitingu validačných dát. Napriek tomu, pri $k \lesssim 170$ je až na niekoľko krátkych úsekov testovacia chyba pri validačnom VS.KL algoritme značne nižšia v porovnaní s trojicou nevalidačných algoritmov a dokonca je výrazne užší aj pás spoľahlivosti. Toto je veľmi sľubný výsledok, ktorý umožňuje označiť validačný VS.KL algoritmus za perspektívneho kandidáta pre testovanie jeho selekčných charakteristík na iných sadách dát. Výrazné zhoršenie testovacej chyby pre $k \gtrsim 170$ má zrejme pôvod v tom, že submodely s takýmto veľkým počtom premenných sú už dostatočne komplexné na to, aby nastalo preučenie na validačnej množine s nie veľkým počtom pozorovaní $N_{\text{Val}} = 1400$.

Kvalitu validačného VS.KL algoritmu v porovnaní s inými nepenalizačnými algoritmami podporuje aj s výrazným odstupom najnižšia testovacia chyba $MSE_{\text{Test}}(S^{(k^*_{\text{Val}})})$ submodelu $S^{(k^*_{\text{Val}})}$ vybraného minimalizáciou validačnej chyby (viď tabuľka 3.1). V ďalšej časti ukážeme, že pomocou algoritmov typu LASSO možno získať modely s ešte nižšími hodnotami testovacej chyby a to navyše pri zanedbateľných výpočtových nárokoch. Napriek tomu, vychádzajúc zo záverov článku [12], sa môže validačný VS.KL algoritmus uplatniť pri dátach s vysokým *signal to noise ratio* (SNR), kedy je pri selekcii vplyv variancie marginálny oproti bias-u. Výsledky validačného VS.KL algoritmu sa môžu tiež zlepšiť v prípade, že okrem trénovacej množiny bude veľký počet pozorovaní aj vo validačnej množine. Výpočtové nároky sa s nárastom počtu pozorovaní nezmenia, pretože pri validačnom VS.KL algoritme pracujeme len s maticami $\mathbf{A} = \left(\mathbf{X}^{\text{Train}}\,\mathbf{y}^{\text{Train}}\right)^\top \left(\mathbf{X}^{\text{Train}}\,\mathbf{y}^{\text{Train}}\right)$, $\left(\mathbf{X}^{\text{Val}}\right)^\top \mathbf{X}^{\text{Val}}$ a vektorom $\left(\mathbf{X}^{\text{Val}}\right)^\top \mathbf{y}^{\text{Val}}$, ktorých rozmery závisia len od počtu prediktorov p.

3.1.2 Penalizačné algoritmy

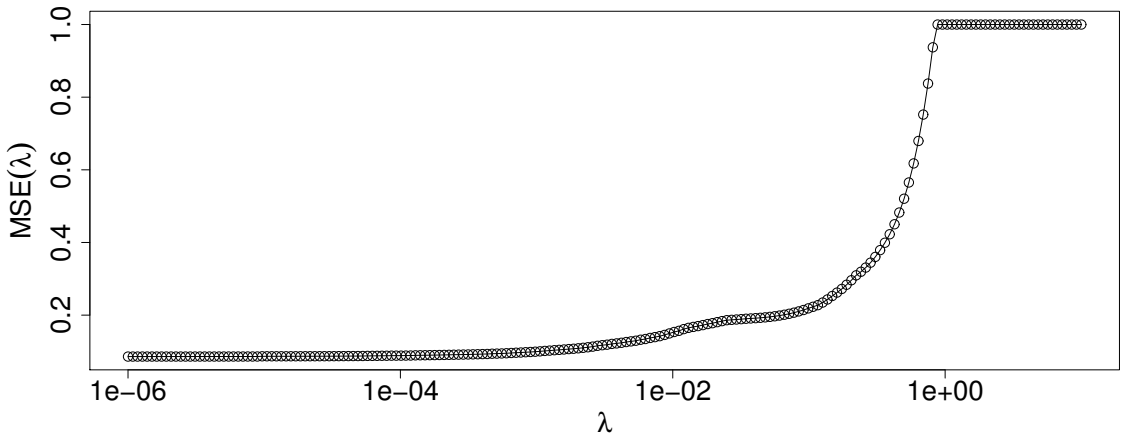
LASSO

Výhodou algoritmu LASSO je, že vďaka prítomnosti regularizačného člena v LASSO úlohe (2.15) ovládaného parametrom λ môže mať vstupná matica plánu $\mathbf{X}$ aj neplnú hodnosť. Narozdiel od všetkých ostatných selekčných metód sme preto nemuseli vykonať predselekciu premenných pomocou QR-rozkladu matice $\mathbf{X}$. Naopak, pomocou LASSO sme prediktory vyberali z pôvodnej sady veľkosti $p = 431$. Navyše bolo potrebné pridať aj intercept β_0, pretože vycentrovaním premenných sa z LASSO úlohy (2.15) vo všeobecnosti neodstráni[6].

Hodnoty penalizačného parametra boli uvažované v rozsahu $\lambda \in [10^{-6}, 10^{1}]$. Konkrétne, bola vytvorená mriežka 200 hodnôt $\{\lambda_i\}_{i=1}^{200}$ tvaru $\lambda_i = 10^{u_i}$, kde hodnoty exponentu u_i boli rovnomerne rozdelené v intervale $[-6, 1]$, t. j. $u_i = -6+(7/199)(i-1)$, $i = 1, \ldots, 200$. Obr. 3.8 znázorňuje priebeh trénovacej chyby $MSE_{\text{Train}}(\lambda) = ||\mathbf{y}^{\text{Train}} - \mathbf{X}^{\text{Train}}\hat{\boldsymbol{\beta}}^{\text{LASSO}}(\lambda)||_2^2/N_{\text{Train}}$ LASSO optimálnych modelov s regresnými koeficientami $\hat{\boldsymbol{\beta}}^{\text{LASSO}}(\lambda)$ pre takto zvolené hodnoty λ. Horné ohraničenie $\lambda \leq 10$ je správne, pretože pri $\lambda \to 10$ je LASSO penalizácia už dosť silná na to, aby optimálnym riešením LASSO úlohy (2.15) bol prázdny model, čo sa ďalším rastom λ už nemôže zmeniť. Toto je v obr. 3.8 reprezentované konštantnou úrovňou trénovacej chyby $MSE_{\text{Train}}(\lambda) = \left(\mathbf{y}^{\text{Train}}\right)^{\top}\mathbf{y}^{\text{Train}}/N_{\text{Train}} = (N_{\text{Train}} - 1)/N_{\text{Train}}$, zodpovedajúcou prázdnemu modelu, pre $\lambda \to 10$. Nižšie ukážeme, že dolné ohraničenie $\lambda \geq 10^{-6}$ je tiež dostatočne nízke, pretože leží naľavo od hodnoty minimalizujúcej validačnú a testovaciu chybu.

Pre postupnosť 200 LASSO optimálnych modelov reprezentovaných regresnými koeficientami $\left\{\hat{\boldsymbol{\beta}}^{\text{LASSO}}(\lambda_i)\right\}_{i=1}^{200}$ bola vypočítaná validačná chyba $MSE_{\text{Val}}(\lambda) = ||\mathbf{y}^{\text{Val}} - \mathbf{X}^{\text{Val}}\hat{\boldsymbol{\beta}}^{\text{LASSO}}(\lambda)||_2^2/N_{\text{Val}}$. Rozdelením validačnej množiny na 10 rovnako veľkých častí bola rovnako ako pri nepenalizačných metódach stanovená aj štandardná odchýlka $\Delta MSE_{\text{Val}}(\lambda)$. Priebeh $MSE_{\text{Val}}(\lambda)$ spolu s pásom spoľahlivosti $MSE_{\text{Val}}(\lambda) \pm \Delta MSE_{\text{Val}}(\lambda)$ je zobrazený na obr. 3.9. Validačná krivka má veľmi podobný tvar ako sme pozorovali pri trénovacej chybe, odlišuje sa len prítomnosťou nevýrazného minima pre $\lambda_{\min} \cong 5.111 \times 10^{-4}$. Rozdiel $MSE_{\text{Val}}(\lambda) - MSE_{\text{Val}}(\lambda_{\min})$ je veľmi malý v po-

[6]Pre všetky uvažované hodnoty λ však optimálna hodnota interceptu $\hat{\beta}_0^{\text{LASSO}}(\lambda) \approx 10^{-18}$, čím sa ukázalo, že jeho pridanie nebolo potrebné.



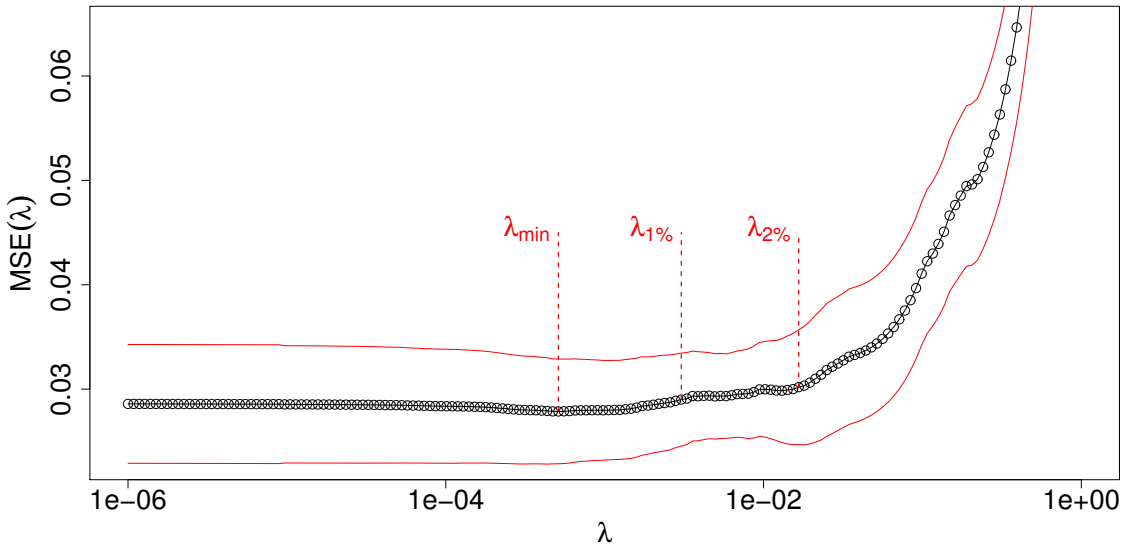
Obr. 3.8: Trénovacia chyba $MSE_{\text{Train}}(\lambda) = ||\mathbf{y}^{\text{Train}} - \mathbf{X}^{\text{Train}}\hat{\boldsymbol{\beta}}^{\text{LASSO}}(\lambda)||_2^2/N_{\text{Train}}$ pre $\lambda \in [10^{-6}, 10^{1}]$. Plató pri $\lambda \to 10$ zodpovedá prázdnemu modelu s $MSE_{\text{Train}}(\lambda) = \left(\mathbf{y}^{\text{Train}}\right)^{\top}\mathbf{y}^{\text{Train}}/N_{\text{Train}} = (N_{\text{Train}} - 1)/N_{\text{Train}}$. Na opačnej strane $MSE_{\text{Train}}(\lambda = 10^{-6}) \approx 0.086$.

rovnaní s $\Delta MSE_{\text{Val}}(\lambda_{\min})$ pre λ zo širokého okolia bodu $\lambda_{\min}$. Pre ďalšiu analýzu bol preto vybraný LRM určený koeficientami $\hat{\boldsymbol{\beta}}^{\text{LASSO}}(\lambda)$ zodpovedajúcimi nielen $\lambda_{\min}$, ale aj hodnotám

$$\lambda_{j\%} = \max\left\{\lambda | MSE_{\text{Val}}(\lambda) \leq MSE_{\text{Val}}(\lambda_{\min}) + j\, 0.01\, \delta MSE_{\text{Val}}\right\}, \tag{3.2}$$

kde $j = 1, 2$ a $\delta MSE_{\text{Val}} = MSE_{\text{Val}}(\lambda = 10) - MSE_{\text{Val}}(\lambda_{\min})$ je rozpätie medzi minimom validačnej chyby a jej maximom $MSE_{\text{Val}}(\lambda \to \infty) = MSE_{\text{Val}}(\lambda = 10)$. Hodnoty $\lambda_{\min}$, $\lambda_{1\%}$, $\lambda_{2\%}$ a im zodpovedajúce chyby sú uvedené v tabuľke 3.2. LASSO modely určené koeficientami $\hat{\boldsymbol{\beta}}^{\text{LASSO}}(\lambda)$ pre $\lambda = \lambda_{\min}, \lambda_{1\%}, \lambda_{2\%}$ sa len málo líšia hodnotou validačnej chyby, sú však priepastne rozdielne z hľadiska počtu $|\alpha_\lambda|$ nenulových zložiek vektora $\hat{\boldsymbol{\beta}}^{\text{LASSO}}(\lambda)$, t.j. počtu premenných v modeli. Hodnoty $|\alpha_\lambda|$ sú pre $\lambda_{\min}$, $\lambda_{1\%}$ a $\lambda_{2\%}$ uvedené v tabuľke 3.2 a pre celú uvažovanú postupnosť $\{\lambda_i\}_{i=1}^{200}$ sú znázornené na obr. 3.10.

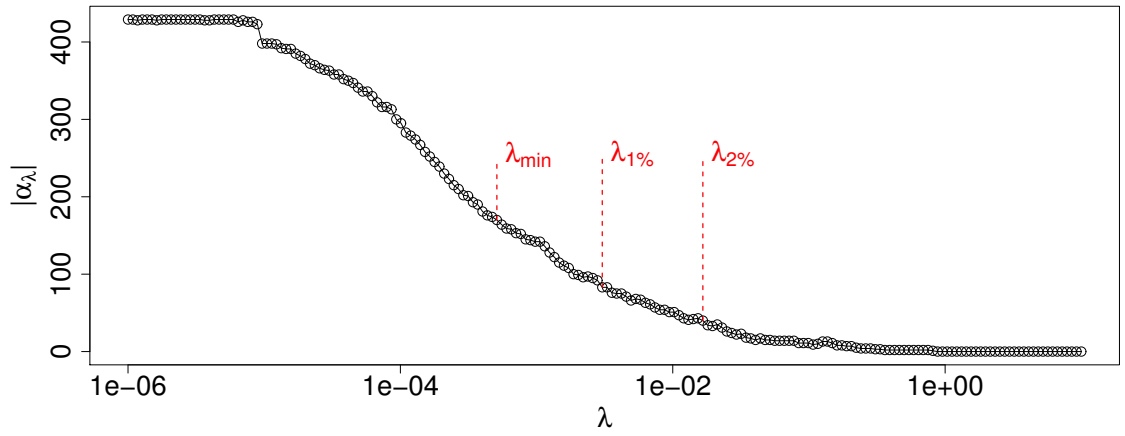
Testovacia chyba $MSE_{\text{Test}}(\lambda) = ||\mathbf{y}^{\text{Test}} - \mathbf{X}^{\text{Test}}\hat{\boldsymbol{\beta}}^{\text{LASSO}}(\lambda)||_2^2/N_{\text{Test}}$, znázornená na obr. 3.11, má kvalitatívne prakticky identický priebeh ako sme pozorovali pri validácii. Významnejší rozdiel je len v navýšení celkovej úrovne testovacej chyby oproti validácii, čo je spôsobené prudšími zmenami výstupu y v testovacej oblasti, ako už bolo diskutované pri nepenalizačných algoritmoch. Minimum testovacej chyby ako aj jej hodnoty



Obr. 3.9: Validačná chyba $MSE_{\text{Val}}(\lambda) = ||\mathbf{y}^{\text{Val}} - \mathbf{X}^{\text{Val}}\hat{\boldsymbol{\beta}}^{\text{LASSO}}(\lambda)||_2^2/N_{\text{Val}}$ a jej pás spoľahlivosti $MSE_{\text{Val}}(\lambda) \pm \Delta MSE_{\text{Val}}(\lambda)$ pre $\lambda \in \{\lambda_i\}_{i=1}^{200}$ a zodpovedajúce koeficienty $\left\{\hat{\boldsymbol{\beta}}^{\text{LASSO}}(\lambda_i)\right\}_{i=1}^{200}$. Monotónny nárast $MSE_{\text{Val}}(\lambda)$ je zakončený plató pri $\lambda \to 10$ zodpovedajúcim prázdnemu modelu, podobne ako v priebehu trénovacej chyby. Táto časť však nie je zobrazená, aby bolo možné detailnejšie zobraziť okolie minima krivky. Červené zvislé čiarkované čiary vyznačujú hodnoty λ_{min}, $\lambda_{1\%}$ a $\lambda_{2\%}$, ktoré sú spolu s im zodpovedajúcimi chybami $MSE_{\text{Val}}(\lambda)$ uvedené v tabuľke 3.2.

Tabuľka 3.2: Hodnoty λ_{min}, $\lambda_{1\%}$, $\lambda_{2\%}$, im zodpovedajúca validačná chyba $MSE_{\text{Val}}(\lambda)$ a testovacia chyba $MSE_{\text{Test}}(\lambda)$ pre LASSO s použitím všetkých $p = 431$ prediktorov. $|\alpha_\lambda|$ označuje počet nenulových zložiek vektora $\hat{\boldsymbol{\beta}}^{\text{LASSO}}(\lambda)$, t.j. počet premenných v LASSO optimálnom modeli pri hodnote λ. Uvedená je aj poloha minima testovacej chyby $\lambda_{\text{min}}^{\text{Test}}$ a jej veľkosť $MSE_{\text{Test}}(\lambda_{\text{min}}^{\text{Test}})$.

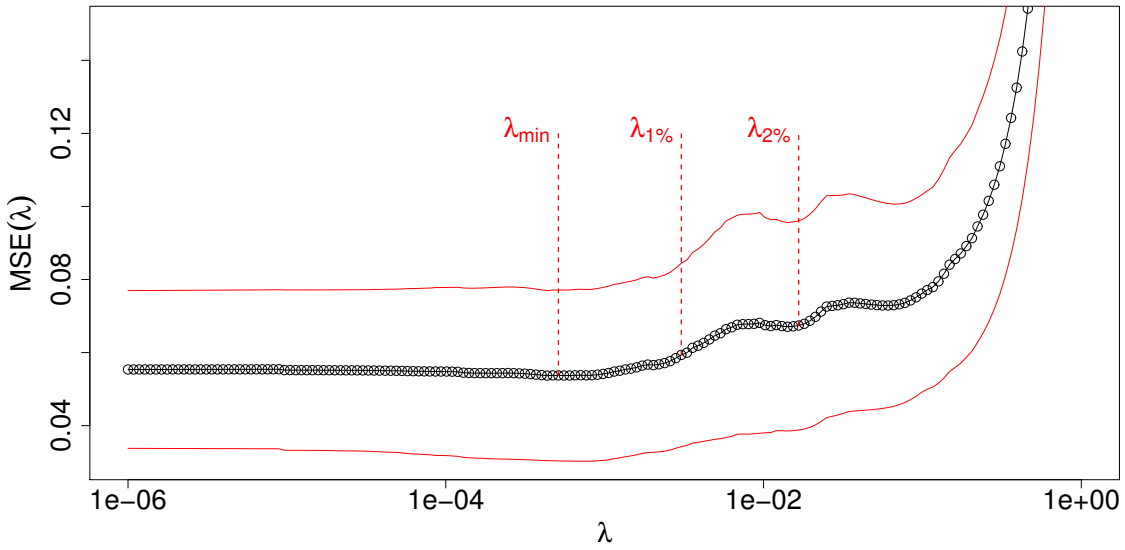
	λ	$\lvert\alpha_\lambda\rvert$	$MSE_{\text{Val}}(\lambda)$	$MSE_{\text{Test}}(\lambda)$
λ_{min}	5.111×10^{-4}	170	0.027863	0.053739
$\lambda_{1\%}$	3.037×10^{-3}	83	0.028998	0.059360
$\lambda_{2\%}$	1.664×10^{-2}	40	0.030187	0.067424
$\lambda_{\text{min}}^{\text{Test}}$	4.347×10^{-4}	176	0.027913	0.053698



Obr. 3.10: Počty premenných $|\alpha_\lambda|$ v LASSO optimálnych modeloch pre $\lambda \in \{\lambda_i\}_{i=1}^{200}$. Zvislými červenými prerušovanými čiarami sú vyznačené $\lambda_{\min}$, $\lambda_{1\%}$, $\lambda_{2\%}$, ktorým zodpovedajúce hodnoty $|\alpha_\lambda|$ sú uvedené v tabuľke 3.2.

pre $\lambda = \lambda_{\min}, \lambda_{1\%}, \lambda_{2\%}$ sú uvedené v tabuľke 3.2. Pomocou $\lambda_{\min}$ sa podarilo polohu minima testovacej chyby odhadnúť veľmi presne. Aj napriek relatívne plytkému minimu validačnej chyby malo teda zmysel presne sa riadiť jeho polohou pri výbere optimálneho modelu. Optimálny model obsahuje relatívne veľa premenných ($|\alpha_{\lambda_{\min}}| = 170$). Toto pre LASSO charakteristické správanie, pozorované aj v článkoch [2, 12], má pôvod vo vychýlení koeficientov $\hat{\boldsymbol{\beta}}^{\text{LASSO}}(\lambda)$ smerom k nulovej hodnote, pretože penalizáciou v tvare ℓ_1-normy $||\boldsymbol{\beta}||_1$ sú „diskriminované“ vektory koeficientov $\boldsymbol{\beta}$ s väčšími hodnotami zložiek β_j. Porovnaním výsledkov LASSO a validačného VS.KL algoritmu dostávame $MSE_{\text{Test}}(\lambda_{\min}) < MSE_{\text{Test}}(S^{(k^*_{\text{Val}})}) < MSE_{\text{Test}}(\lambda_{1\%})$. Validačný VS.KL je teda z hľadiska odhadu predikčnej chyby schopný konkurovať selekcii pomocou LASSO, je však značne výpočtovo náročnejší. Dokonca, môže byť vhodnejšou alternatívou v prípade, ak z nejakého dôvodu[7] sa chceme obmedziť na submodely s relatívne malým počtom prediktorov. Toto zodpovedá vysokým hodnotám λ pri LASSO, kedy však veľká vychýlenosť odhadov $\hat{\boldsymbol{\beta}}^{\text{LASSO}}_{\lambda}$ výrazne zvyšuje odhad predikčnej chyby. Napríklad, ak porovnáme model $S^{(k^*_{\text{Val}})}$ zvolený validačným VS.KL algoritmom a LASSO model pre $\lambda = \lambda_{1\%}$ s podobnými počtami parametrov $k^*_{\text{Val}} = 93$ a $|\alpha_{\lambda_{1\%}}| = 83$, nižšiu testovaciu chybu (odhad predikčnej chyby) sme získali pri validačnom VS.KL algoritme.

[7]Napr. minimalizácia nákladov na meranie vstupných premenných, či interpretovateľnosť výsledného LRM.



Obr. 3.11: Testovacia chyba $MSE_{\text{Test}}(\lambda) = ||\mathbf{y}^{\text{Test}} - \mathbf{X}^{\text{Test}}\hat{\boldsymbol{\beta}}^{\text{LASSO}}(\lambda)||_2^2/N_{\text{Test}}$ a jej pás spoľahlivosti $MSE_{\text{Test}}(\lambda) \pm \Delta MSE_{\text{Test}}(\lambda)$ pre $\lambda \in \{\lambda_i\}_{i=1}^{200}$ a zodpovedajúce koeficienty $\left\{\hat{\boldsymbol{\beta}}^{\text{LASSO}}(\lambda_i)\right\}_{i=1}^{200}$. Monotónny nárast $MSE_{\text{Test}}(\lambda)$ je zakončený plató pri $\lambda \to 10$ zodpovedajúcim prázdnemu modelu. Táto časť však nie je zobrazená, aby bolo možné detailnejšie zobraziť okolie minima krivky. Červené zvislé čiarkované čiary vyznačujú hodnoty $\lambda_{\min}$, $\lambda_{1\%}$ a $\lambda_{2\%}$, ktoré sú spolu s im zodpovedajúcimi chybami $MSE_{\text{Test}}(\lambda)$ a minimom testovacej chyby uvedené v tabuľke 3.2.

Napriek tomu, že LASSO nevyžaduje plnú hodnosť matice plánu $\mathbf{X}$, je relevantné porovnať nepenalizačné algoritmy a LASSO za rovnakých podmienok, t. j. pri selekcii z množiny $p = 382$ prediktorov vytvorenej predvýberom z pôvodnej množiny 431 premenných pomocou pivotizácie z QR-rozkladu matice $\mathbf{X}$. Aplikáciou LASSO na takéto dáta (s maticou $\mathbf{X}$ plnej hodnosti 382), dostávame veľmi podobné priebehy validačnej a testovacej krivky ($MSE_{\text{Val}}(\lambda)$ a $MSE_{\text{Test}}(\lambda)$) ako v prípade $p = 431$ prezentovanom vyššie. Minimálne sú aj rozdiely v hodnotách $\lambda_{\min}$, $\lambda_{1\%}$, $\lambda_{2\%}$, im zodpovedajúcich chýb $MSE_{\text{Val}}(\lambda)$, $MSE_{\text{Test}}(\lambda)$, či počtoch premenných v modeli $|\alpha_\lambda|$, ktoré uvádzame v tabuľke 3.3.

Relaxované LASSO

Relaxované LASSO, narozdiel od klasického LASSO, potrebuje ako vstup maticu plánu $\mathbf{X}$ plnej hodnosti. Súčasťou výpočtu koeficientov $\hat{\boldsymbol{\beta}}^{\text{relax}}(\lambda, \gamma)$ je totiž aj výpočet LS

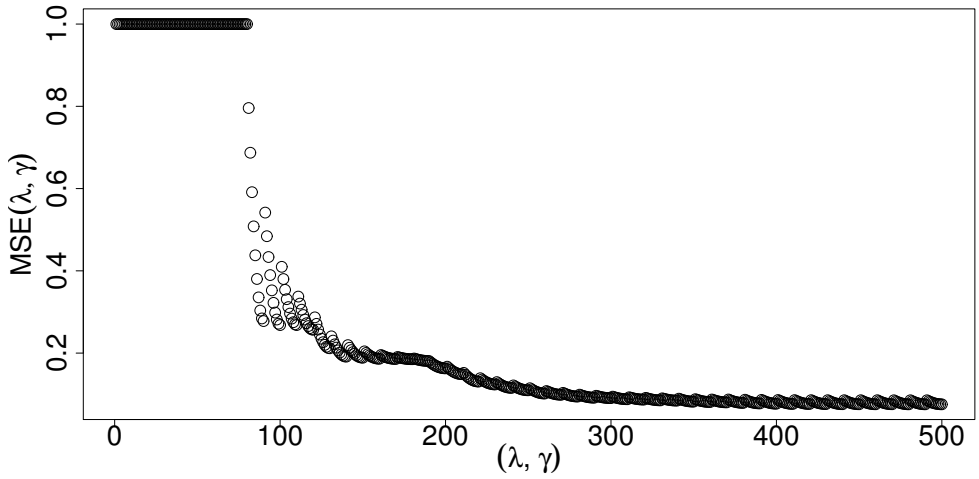
Tabuľka 3.3: Hodnoty $\lambda_{\min}$, $\lambda_{1\%}$, $\lambda_{2\%}$, im zodpovedajúca validačná chyba $MSE_{\text{Val}}(\lambda)$, testovacia chyba $MSE_{\text{Test}}(\lambda)$ a počet premenných v modeli $|\alpha_\lambda|$ pre LASSO pri použití $p = 382$ prediktorov zvolených predvýberom pomocou QR-rozkladu matice plánu $\mathbf{X}$. Doplnená je aj poloha minima testovacej chyby $\lambda_{\min}^{\text{Test}}$ a jej veľkosť $MSE_{\text{Test}}(\lambda_{\min}^{\text{Test}})$.

	λ	$\lvert\alpha_\lambda\rvert$	$MSE_{\text{Val}}(\lambda)$	$MSE_{\text{Test}}(\lambda)$
$\lambda_{\min}$	5.543×10^{-4}	157	0.027980	0.053524
$\lambda_{1\%}$	2.801×10^{-3}	89	0.029135	0.057792
$\lambda_{2\%}$	1.804×10^{-2}	34	0.030316	0.067739
$\lambda_{\min}^{\text{Test}}$	6.010×10^{-4}	150	0.027986	0.053510

odhadu $\hat{\boldsymbol{\beta}}_{\alpha_\lambda}^{\text{LS}}$ pre submodel daný množinou indexov α_λ nenulových zložiek vektora $\hat{\boldsymbol{\beta}}^{\text{LASSO}}(\lambda)$. Relaxované LASSO sme preto aplikovali na množinu $p = 382$ prediktorov vytvorenú predvýberom premenných pomocou pivotizácie z QR-rozkladu matice plánu $\mathbf{X}$. Uvažovali sme mriežku 50 hodnôt $\{\lambda_i\}_{i=1}^{50}$ rozmiestnených v intervale $\lambda \in [10^{-6}, 10^{1}]$ podľa vzorca $\lambda_i = 10^{u_i}$, kde hodnoty exponentu u_i boli rovnomerne rozdelené v intervale $[-6, 1]$, t. j. $u_i = -6 + (7/49)(i-1)$, $i = 1, \ldots, 50$. Mriežka hodnôt $\{\gamma_i\}_{i=1}^{10}$ parametra γ bola tvorená 10 bodmi rovnomerne rozmiestnenými v intervale $[0, 1]$.

Priebeh trénovacej chyby $MSE_{\text{Train}}(\lambda, \gamma) = ||\mathbf{y}^{\text{Train}} - \mathbf{X}^{\text{Train}}\hat{\boldsymbol{\beta}}^{\text{relax}}(\lambda, \gamma)||_2^2 / N_{\text{Train}}$ modelov zvolených relaxovaným LASSO pre kombinácie parametrov $\{(\lambda_i, \gamma_j)\}_{i=1,j=1}^{50,10}$ je znázornený na obr. 3.12. Túto závislosť je potrebné interpretovať dvojúrovňovo, pretože hodnoty (λ_i, γ_j) dvojrozmernej mriežky sú na jednorozmernej x-ovej osi usporiadané v blokoch $(\lambda_i, \gamma_1 = 1), \ldots, (\lambda_i, \gamma_{10} = 0)$ s fixovaným λ_i a postupne klesajúcimi hodnotami γ[8]. Hlavný trend závislosti $MSE_{\text{Train}}(\lambda, \gamma)$ je tak prioritne daný závislosťou od parametra λ. Táto má charakter postupného poklesu s klesajúcim λ, analogického ako v závislosti $MSE_{\text{Train}}(\lambda)$ pre klasické LASSO. Nie je to prekvapivé, pretože každý z 10-prvkových blokov $(\lambda_i, \{\gamma_j\}_{j=1}^{10})$ na x-ovej osi začína kombináciou $(\lambda_i, \gamma_1 = 1)$, ktorej zodpovedá LASSO odhad regresných koeficientov $\hat{\boldsymbol{\beta}}^{\text{relax}}(\lambda_i, \gamma = 1) = \hat{\boldsymbol{\beta}}^{\text{LASSO}}(\lambda_i)$. V rámci blokov $(\lambda_i, \{\gamma_j\}_{j=1}^{10})$ trénovacia chyba $MSE_{\text{Train}}(\lambda_i, \gamma)$ klesá s poklesom γ viac,

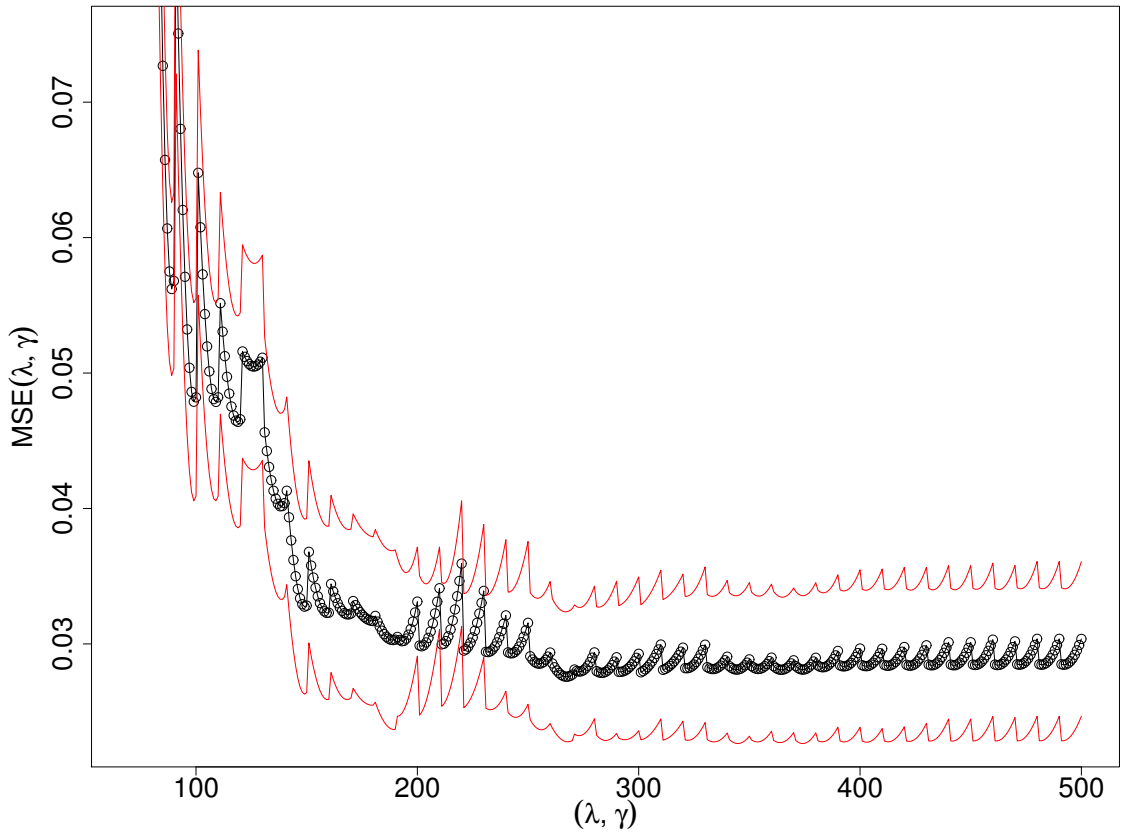
[8]Dvojrozmernému grafu sme sa snažili vyhnúť napríklad aj preto, že pri vykresľovaní validačnej a testovacej chyby by bolo komplikované zobraziť intervaly spoľahlivosti pre $MSE(\lambda, \gamma)$.



Obr. 3.12: Trénovacia chyba $MSE_{\text{Train}}(\lambda,\gamma) = ||\mathbf{y}^{\text{Train}} - \mathbf{X}^{\text{Train}}\hat{\boldsymbol{\beta}}^{\text{relax}}(\lambda,\gamma)||_2^2/N_{\text{Train}}$ pre $\lambda \in [10^{-6}, 10^{1}]$. Parametre $\{\lambda_i\}_{i=1}^{50}, \{\gamma\}_{i=1}^{10}$ sú usporiadané po blokoch $(\lambda_i, \{\gamma_j\}_{j=1}^{10})$ s fixovanou hodnotou $\lambda = \lambda_i$. Hodnoty λ klesajú v smere zľava doprava, γ klesá od $\gamma_1 = 1$ do $\gamma_{10} = 0$ smerom doprava v rámci každého z blokov.

či menej výrazne, v závislosti od toho, ako silne LASSO penalizácia vychýlila odhad koeficientov $\hat{\boldsymbol{\beta}}^{\text{LASSO}}(\lambda_i)$ smerom k nule oproti LS odhadu $\hat{\boldsymbol{\beta}}^{\text{LS}}_{\alpha_{\lambda_i}}$. Inak povedané, nakoľko penalizácia s parametrom λ_i znížila počet stupňov voľnosti množiny prípustných LRM (zložitosť triedy hypotéz) pre regresiu trénovacích dát oproti LS úlohe. Pre $\lambda \to 10$ opäť kvôli silnej penalizácii dostávame prázdny model.

Obr. 3.13 zobrazuje validačnú chybu $MSE_{\text{Val}}(\lambda,\gamma) = ||\mathbf{y}^{\text{Val}} - \mathbf{X}^{\text{Val}}\hat{\boldsymbol{\beta}}^{\text{relax}}(\lambda,\gamma)||_2^2/N_{\text{Val}}$ a jej pás spoľahlivosti $MSE_{\text{Val}}(\lambda,\gamma) \pm \Delta MSE_{\text{Val}}(\lambda,\gamma)$ získaný obvyklým spôsobom. Hlavný trend (závislosť od λ) je veľmi podobný ako pri trénovacej chybe. Výrazne odlišný je však priebeh $MSE_{\text{Val}}(\lambda,\gamma)$ v závislosti od γ v rámci blokov $(\lambda_i, \{\gamma_j\}_{j=1}^{10})$ s fixovanou hodnotou $\lambda = \lambda_i$. V ľavej časti (veľké λ) chyba $MSE_{\text{Val}}(\lambda,\gamma)$ prudko klesá s tým, ako sa poklesom γ menia koeficienty $\hat{\boldsymbol{\beta}}^{\text{relax}}(\lambda,\gamma)$ z LASSO odhadu $\hat{\boldsymbol{\beta}}^{\text{LASSO}}(\lambda)$ na LS odhad $\hat{\boldsymbol{\beta}}^{\text{LS}}(\lambda)$. Vzhľadom na silnú penalizáciu je počet stupňov voľnosti pri LASSO v tejto oblasti nedostatočný na kvalitný popis dát, hlavným príspevkom k chybe v prípade LASSO je preto vychýlenie odhadu $\hat{\boldsymbol{\beta}}^{\text{LASSO}}(\lambda)$ smerom k nule (bias). LS odhad $\hat{\boldsymbol{\beta}}^{\text{LS}}(\lambda)$ konštruovaný na množine premenných α_λ je nevychýlený, vďaka malému počtu $|\alpha_\lambda|$ premenných v modeli súčasne nedochádza k preučeniu, variančná zložka chyby pre LS je preto veľmi nízka. Prechodom od LASSO odhadu smerom k LS odhadu sa preto



Obr. 3.13: Validačná chyba $MSE_{\text{Val}}(\lambda,\gamma) = ||\mathbf{y}^{\text{Val}} - \mathbf{X}^{\text{Val}}\hat{\boldsymbol{\beta}}^{\text{relax}}(\lambda,\gamma)||_2^2/N_{\text{Val}}$ a jej pás spoľahlivosti $MSE_{\text{Val}}(\lambda,\gamma) \pm \Delta MSE_{\text{Val}}(\lambda,\gamma)$ pre relaxované LASSO a $\lambda \in [10^{-6}, 10^{1}]$. Minimum $MSE_{\text{Val}}\left(\lambda_{\text{min}}^{\text{Val}}, \gamma_{\text{min}}^{\text{Val}}\right) = 0.027578$ sa nadobúda pre $\left(\lambda_{\text{min}}^{\text{Val}}, \gamma_{\text{min}}^{\text{Val}}\right) = (1.9307 \times 10^{-3}, 1/3)$. Počet premenných v zodpovedajúcom modeli je $|\alpha_{\lambda_{\text{min}}^{\text{Val}}}| = 106$.

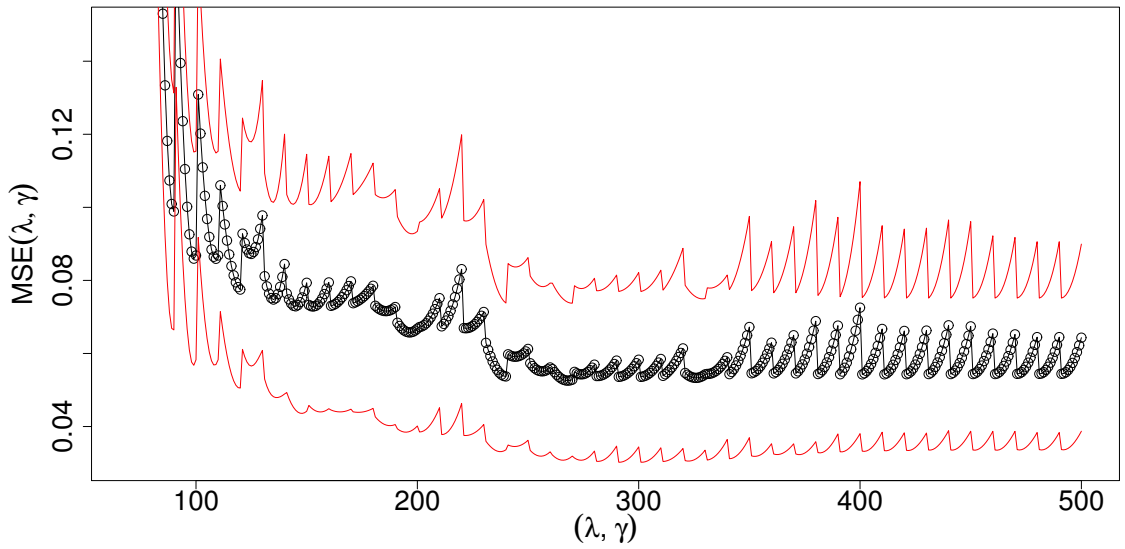
znižuje časť chyby spôsobená bias-om bez toho, aby to vyvolalo významnejší nárast variančnej zložky.

V pravej časti priebehu pozorujeme presne opačné správanie. Čím viac poklesom γ vplýva LS odhad $\hat{\boldsymbol{\beta}}^{\text{LS}}(\lambda)$ na hodnotu koeficientov $\hat{\boldsymbol{\beta}}^{\text{relax}}(\lambda,\gamma)$, tým viac narastá chyba $MSE_{\text{Val}}(\lambda,\gamma)$. V tejto oblasti je penalizácia veľmi slabá, preto je počet premenných $|\alpha_\lambda|$ v modeli veľký. Počet stupňov voľnosti pri LASSO je už postačujúci na kvalitný popis dát, preto zložka chyby spôsobená bias-om klesá k svojmu minimu v rámci uvažovanej množiny LRM. Ešte väčší počet stupňov voľnosti získame pri konštrukcii LS odhadu, čo už ale vedie k preučeniu a nafúknutiu variančnej zložky chyby. Prechodom od LASSO odhadu smerom k LS odhadu sa preto príspevok chyby spôsobeniej bias-om prakticky nemení, kým variančná zložka významne narastie.

Pre hodnoty λ medzi týmito dvoma extrémami pozorujeme bias-variance trade-off medzi podučením pri LASSO a preučením pri LS. Výsledkom je, že závislosť $MSE_{\text{Val}}(\lambda, \gamma)$ ako funkcia γ má tvar „U“, ktoré je vychýlené na jednu, či druhú stranu, podľa toho, či prevažuje variančná zložka chyby alebo príspevok od bias-u. Minimum chyby sa dosahuje pre $0 < \gamma < 1$, teda pre model, ktorý je kompromisom medzi LASSO a LS.

Pri klasickom LASSO je vo validačnej krivke často pozorované veľmi plytké minimum, čo výrazne zvyšuje neistotu pri výbere modelu minimalizujúceho predikčnú chybu na základe validácie. Relaxované LASSO nám výber značne uľahčuje, lebo pre každé λ ihneď vidíme, ktorá zložka chyby prevažuje, resp. aká je veľká. Aj napriek relatívne širokému intervalu spoľahlivosti validačnej chyby $MSE_{\text{Val}}(\lambda, \gamma)$ sa týmto výrazne zvyšuje miera nášho presvedčenia o vhodnosti výberu konkrétneho LRM na základe priebehu validačnej krivky. V tomto konkrétnom prípade je zrejmé, že naľavo od polohy minima validačnej chyby $\left(\lambda_{\min}^{\text{Val}}, \gamma_{\min}^{\text{Val}}\right)$ je prevažujúcim efektom podučenie, napravo preučenie. V bloku priamo obsahujúcom minimum sú tieto efekty vyvážené, pretože závislosť $MSE_{\text{Val}}(\lambda_{\min}^{\text{Val}}, \gamma)$ má tvar „U“ s rovnako vysokými stranami. Potlačenie bias-u čiastočným prechodom od LASSO odhadu $\hat{\boldsymbol{\beta}}^{\text{LASSO}}(\lambda)$ k LS odhadu $\hat{\boldsymbol{\beta}}^{\text{LS}}(\lambda)$ má za následok, že v optimálnom modeli je len $|\alpha_{\lambda_{\min}^{\text{Val}}}| = 106$ premenných, čo je výrazne menej (zhruba 2/3) než pri klasickom LASSO.

Priebeh testovacej chyby $MSE_{\text{Test}}(\lambda, \gamma) = ||\mathbf{y}^{\text{Test}} - \mathbf{X}^{\text{Test}}\hat{\boldsymbol{\beta}}^{\text{relax}}(\lambda, \gamma)||_2^2 / N_{\text{Test}}$ (viď obr. 3.14) veľmi detailne kopíruje tvar validačnej krivky. Preto aj poloha minima validačnej chyby $\left(\lambda_{\min}^{\text{Val}}, \gamma_{\min}^{\text{Val}}\right)$ je dobrým odhadom polohy minima testovacej chyby $\left(\lambda_{\min}^{\text{Test}}, \gamma_{\min}^{\text{Test}}\right)$, sú to susedné mriežkové body. Získaná hodnota odhadu predikčnej chyby $MSE_{\text{Test}}\left(\lambda_{\min}^{\text{Val}}, \gamma_{\min}^{\text{Val}}\right)$ je mierne nižšia oproti klasickému LASSO, a tým aj oproti všetkým nepenalizačným metódam. Vo všeobecnosti by relaxované LASSO malo vždy priniesť aspoň tak dobré výsledky ako klasické LASSO, lebo je jeho zovšeobecnením. Vzhľadom na vyššie uvedené sa relaxované LASSO zdá byť univerzálne odporučiteľným selekčným algoritmom v prípadoch, kedy matica plánu $\mathbf{X}$ má plnú hodnosť. Ak táto podmienka nie je splnená, možno pred použitím relaxovaného LASSO vykonať predselekciu napr. pomocou QR-rozkladu matice $\mathbf{X}$ alebo klasickým LASSO.



Obr. 3.14: Testovacia chyba $MSE_{\text{Test}}(\lambda,\gamma) = ||\mathbf{y}^{\text{Test}} - \mathbf{X}^{\text{Test}}\hat{\boldsymbol{\beta}}^{\text{relax}}(\lambda,\gamma)||_2^2/N_{\text{Test}}$ a jej pás spoľahlivosti $MSE_{\text{Test}}(\lambda,\gamma) \pm \Delta MSE_{\text{Test}}(\lambda,\gamma)$ pre relaxované LASSO a $\lambda \in [10^{-6}, 10^{1}]$. Minimum $MSE_{\text{Test}}\left(\lambda_{\text{min}}^{\text{Test}}, \gamma_{\text{min}}^{\text{Test}}\right) = 0.052557$ sa nadobúda pre $\left(\lambda_{\text{min}}^{\text{Test}}, \gamma_{\text{min}}^{\text{Test}}\right) = (1.9307 \times 10^{-3}, 2/9)$. Počet premenných v zodpovedajúcom modeli je $|\alpha_{\lambda_{\text{min}}^{\text{Test}}}| = |\alpha_{\lambda_{\text{min}}^{\text{Val}}}| = 106$, keďže $\lambda_{\text{min}}^{\text{Test}} = \lambda_{\text{min}}^{\text{Val}}$. Chyba pre parametre $\left(\lambda_{\text{min}}^{\text{Val}}, \gamma_{\text{min}}^{\text{Val}}\right)$ stanovené na základe validácie je $MSE_{\text{Test}}\left(\lambda_{\text{min}}^{\text{Val}}, \gamma_{\text{min}}^{\text{Val}}\right) = 0.052617$.

Záver

Pôvodným zámerom predkladanej práce bolo porovnať štandardné metódy selekcie premených v lineárnom regresnom modeli, ako napr. dopredná a spätná selekcia či LASSO, z hľadiska minimalizácie predikčnej chyby nimi zvoleného modelu. Prirodzenou súčasťou tejto úlohy je aj snaha o odhad minima očakávanej predikčnej chyby dosiahnuteľnej akýmsi teoreticky ideálnym selekčným algoritmom. Otázka vhodného prístupu k hľadaniu globálneho minima odhadu predikčnej chyby však dosiaľ nie je uspokojivo vyriešená. Viacerí autori hľadajú odpoveď v tzv. *best subset* prístupe, pri ktorom je úplným prehľadaním množiny submodelov vytvorená postupnosť LRM, ktoré minimalizujú trénovaciu chybu pre každý prípustný počet premenných v submodeli. Z tejto postupnosti je následne ako optimálny submodel zvolený ten, ktorý minimalizuje (cross)validačnú chybu alebo niektoré z informačných kritérií. Iné metódy selekcie (dopredná selekcia, LASSO, ...) sú považované len za heuristické aproximácie best subset prístupu v prípade, keď úplné, alebo aspoň dostatočne dôkladné, prehľadanie množiny všetkých prípustných submodelov nie je z dôvodu výpočtovej náročnosti prakticky realizovateľné. Ako je však teoreticky a aj pomocou numerických simulácií zdôvodnené v článku [12], tento pohľad vo všeobecnosti nie je správny. Má svoje odpodstatnenie iba v prípade vysokého *signal to noise ratio* (SNR), t. j. v situácii, kedy náhodná zložka v dátach (šum) je výrazne nižšia než deterministický príspevok modelu popisujúceho závislosť medzi vstupmi a výstupom.

V článku [12] bolo ukázané, že univerzálnym selekčným algoritmom, ktorý, v porovnaní s best subset, doprednou selekciou a LASSO, vedie k najnižším hodnotám odhadovanej predikčnej chyby, je tzv. relaxované LASSO. Avšak ešte predtým, než sme sa s citovaným článkom oboznámili (január 2018), zamerali sme sa na best subset prístup. Snahou bolo vytvoriť selekčný algoritmus, ktorý by síce negarantoval nájdenie

submodelov danej veľkosti minimalizujúcich trénovaciu chybu ako v prípade best subset, tieto výsledky by však veľmi dobre aproximoval, a pritom by bol dostatočne rýchly pre praktické výpočty, a to aj pre modely s rádovo stovkami premenných. Výsledkom je tzv. VS.KL algoritmus. Vychádzajúc z podstaty delenia dát na trénovaciu a validačnú množinu, ako aj v snahe o implementáciu záverov z článku [12], boli následne modifikáciou algoritmov VS.KL a doprednej selekcie vytvorené ich validačné verzie.

Význam tejto práce teda treba chápať v dvoch rovinách - teoretickej a praktickej. Hlavné teoretické výsledky sú nasledovné:

Čitateľ je oboznámený s výhodami a eleganciou formalizmu sweepovania, ako špeciálneho prípadu tzv. principal pivot transform (PPT). Vlastnosti podstatné pre aplikáciu sweepovania pri selekcii premenných v LRM sú detailne popísané vo forme viet a liem, vo väčšine prípadov aj s potrebnými dôkazmi. Táto časť práce významne čerpá z prehľadového článku o PPT [24] a krátkych kapitol v knihách [20, 15], pritom však nejde len o kompilát už skôr publikovaných výsledkov. Niektoré vety a dôkazy (napr. lema 1.4.6) boli totiž významne preformulované, aby lepšie vyhovovali potrebám tejto práce. Okrem toho, pokiaľ je nám známe, dosiaľ nepublikovaným výsledkom je formulácia a dôkaz viet 1.4.12 a 1.4.20 pre elegantný výpočet hodnoty RSS modelu vygenerovaného ľubovoľnou kombináciou pridania a odobrania premenných z aktuálneho modelu. Špeciálne pedagogické úsilie je venované intuitívnemu odvodeniu formalizmu sweepovania a jeho vlastností na základe súvisu so sústavou rovníc a ohraničení (1.21) - (1.23). V dostupnej literatúre je totiž sweepovanie zavádzané definitoricky, bez toho, aby bol bližšie objasnený súvis s pôvodnou úlohou selekcie premenných v LRM. Z vlastných pozorovaní vychádzajú aj uvedené odporúčania podstatné pre dosiahnutie uspokojivej numerickej presnosti pri praktickej realizácii výpočtov na báze swepovania.

Druhým hlavným teoretickým výstupom je návrh algoritmu VS.KL, ktorý je heuristickou náhradou best subset prístupu s cieľom výrazne znížiť výpočtovú náročnosť pri zanedbateľnom zhoršení kvality výsledkov. VS.KL bol vytvorený spojením hlavnej myšlienky výmenného KL algoritmu používaného v oblasti optimálneho návrhu štatistického experimentu ([19, 1]) a formalizmu sweepovania. Podobná metóda, hoci zrejme inak motivovaná, bola už v minulosti navrhnutá Millerom [18] pod názvom sequential replacement. Nami navrhnutý algoritmus sa však výrazne líši detailami implementá-

cie, najmä mechanizmom reštartovania, ktorý výrazne zlepšuje rýchlosť konvergencie a stabilitu výsledku. Ako naznačujú doterajšie numerické výsledky, VS.KL dokáže pri malom počte premenných (maximálne 50-100) a vhodnom nastavení parametrov dosiahnuť prakticky identické výsledky ako referenčný branch and bound best subset algoritmus od Furnivala a Wilsona [5], a to za výrazne kratší čas výpočtu. Pri väčšom počte premenných, kedy už branch and bound prístup nie je prakticky realizovateľný, VS.KL dosahuje nižšie RSS pri výrazne nižšej výpočtovej náročnosti než alternatívna metóda založená na zmiešanom celočíselnom programovaní (MIO).

Ďalšími vlastnými návrhmi selekčných algoritmov sú validačné verzie VS.KL a doprednej selekcie. Tieto metódy sa snažia vytvoriť postupnosť submodelov minimalizujúcich validačnú (a nie trénovaciu) chybu pre každý prípustný počet premenných, pritom však regresné koeficienty sú dané ako LS odhady $\hat{\boldsymbol{\beta}}$ vypočítané z trénovacej množiny. Na základe doterajších výsledkov možno usudzovať, že validačný VS.KL algoritmus má potenciál byť za istých okolností konkurenciou pre relaxované LASSO ako referenčnú metódu, pokiaľ nebude kladený dôraz na dobu výpočtu.

Medzi teoretické výstupy možno zaradiť aj detailný popis prehľadávacích selekčných algoritmov (best subset, dopredná a spätná selekcia), pri ktorom je každý krok jasne matematicky formulovaný. V tejto časti nadväzujeme na výborný a najmä výnimočne veľmi podrobný výklad v knihe [20], ktorý v prípade potreby rozvíjame o ďalšie detaily.

Praktické výstupy sú reprezentované porovnaním numerických výsledkov aplikácie uvedených selekčných metód na sadu reálnych dát s viac ako 400 premennými a 20000 pozorovaniami. Pri komparatívnej analýze sme uprednostnili reálne dáta pred simulovanými, pretože v tomto prípade sú selekčné algoritmy vystavené oveľa náročnejšej úlohe. Musia si totiž poradiť s tým, že zdrojový model sa vo všeobecnosti v čase postupne vyvíja.

Potvrdila sa nízka výpočtová náročnosť penalizačných metód (LASSO a relaxované LASSO) a doprednej selekcie, v prípade ktorých výpočet trval len niekoľko sekúnd, maximálne desiatok sekúnd, a to na počítači s relatívne slabým výpočtovým výkonom. LASSO metódy nielenže viedli k najnižším odhadom predikčnej chyby, ale umožňovali aj optimálny model veľmi presne odhadnúť na základe validačnej krivky. Relaxované LASSO vytvorilo model len s mierne nižším odhadom predikčnej chyby než klasické

LASSO, avšak tento model obsahoval výrazne nižší počet premenných (106 oproti 157). Navyše, vďaka tvaru validačnej krivky bol optimálny model pri relaxovanom LASSO zvolený s výrazne vyššou mierou presvedčenia o vhodnosti vykonanej voľby. Jediným pozorovaným nedostatkom relaxovaného LASSO je skutočnosť, že, narozdiel od klasického LASSO, nemôže byť priamo aplikované na dáta s lineárne závislými prediktormi. Ukázali sme však postup, ako tento problém obísť predvýberom množiny lineárne nezávislých prediktorov.

V kategórii prehľadávacích metód sa model s najnižším odhadom predikčnej chyby podarilo zvoliť pri validačnom VS.KL algoritme. Získaný odhad je však mierne horší než v prípade LASSO metód. Výsledky validačného VS.KL algoritmu by sa mohli zlepšiť, ak by bola zvolená väčšia validačná množina. V našom prípade obsahovala len zhruba 7% dát. S výrazným odstupom za validačným VS.KL nasledovali výsledky VS.KL, MIO a doprednej selekcie, medzi ktorými, s prihliadnutím na výrazne roztrasený charakter validačnej a testovacej krivky a šírku pásu spoľahlivosti, nie je z pohľadu predikcie významný rozdiel. VS.KL algoritmus však vynikal pri hľadaní modelov minimalizujúcich trénovaciu chybu, tesne nasledovaný MIO prístupom, ktorý je však výrazne výpočtovo náročnejší. Dopredná selekcia zvolila najmä pri nižších veľkostiach modely s výrazne horšími trénovacími chybami. Najhoršie modely z hľadiska odhadu predikčnej chyby boli zvolené validačnou doprednou selekciou.

Na základe uvedeného a s prihliadnutím na predbežné výsledky na iných sadách dát možno súhlasiť so závermi článku [12], že najuniverzálnejším algoritmom pre selekciu premenných v LRM s cieľom minimalizácie odhadu predikčnej chyby je relaxované LASSO. Bolo by však vhodné ďalšími analýzami preveriť potenciál validačného VS.KL algoritmu, najmä v prípade dostatočne veľkej validačnej množiny a vysokého SNR. Pre aproximatívne riešenie best subset úlohy pri rádovo stovkách prediktorov možno odporúčať použitie VS.KL algoritmu.

Zoznam použitej literatúry

[1] Atkinson, A., C., Donev, A., N., Tobias, R., D.: *Optimum experimental designs, with SAS*, 1. vyd., Oxford University Press, Oxford, 2007

[2] Bertsimas, D., King, A., Mazumder, R.: *Best subset selection via a modern optimization lens.* The Annals of Statistics 44 (2016), 813–852

[3] Cottle, R. W.: *The principal pivoting method revisited.* Mathematical Programming 48 (1990), 369–385

[4] Efroymson, M. A.: *Multiple regression analysis.* Mathematical Methods for Digital Computers, vol. I (1960), 191–203

[5] Furnival, G. M., Wilson R. W.: *Regressions by leaps and bounds.* Technometrics 16 (1974), 499-511

[6] Gareth, J., Witten, D., Hastie, T., Tibshirani, R.: *An introduction to statistical learning*, 1. vyd., Springer, New York, 2013

[7] Geyer, Ch. J.: *Stat 5101 lecture notes*, 2001, dostupné na internete (21. 9. 2017): http://www.stat.umn.edu/geyer/5102/notes/n2.pdf

[8] Geyer, Ch. J.: *Stat 5102 lecture slides: Deck 7 model selection*, 2016, dostupné na internete (21. 9. 2017): www.stat.umn.edu/geyer/5102/slides/s7.pdf

[9] Goodnight, J.: *A tutorial on the SWEEP operator.* The American Statistician 33 (1979), 149-158

[10] Hastie, T., Tibshirani, R., Friedman, J.: *The elements of statistical learning*, 2. vyd., Springer, New York, 2009

[11] Hastie, T. and Tibshirani, R. and Wainwright, M. : *Statistical learning with sparsity: the LASSO and generalizations*, 1. vyd., CRC press, Boca Raton, 2015

[12] Hastie, T., Tibshirani, R., Tibshirani, R. J.: *Extended comparisons of best subset selection, forward stepwise selection, and the LASSO.* arXiv preprint arXiv:1707.08692 (2017)

[13] Hastie, T., Tibshirani, R., Tibshirani, R. J.: *Best subset selection and related tools*, 2017, dostupné na internete (1. 5. 2018): https://github.com/ryantibs/best-subset/

[14] Konishi, S., Kitagawa, G.: *Information criteria and statistical modeling*, 1. vyd., Springer, New York, 2008

[15] Lange, K.: *Numerical analysis for statisticians*, 2. vyd., Springer, New York, 2010

[16] Jennrich, R. I. 1977. *Stepwise regression.* In Statistical Methods for Digital Computers. New York: Wiley - Interscience, 1977. s. 58–75

[17] Meinshausen, N.: *Relaxed LASSO.* Computational Statistics & Data Analysis 52 (2007), 374–393

[18] Miller, A.: *Subset selection in regression*, 2. vyd., Chapman and Hall/CRC Press, Boca Raton, 2002

[19] Pázman, A., Lacko, V.: *Prednášky z regresných modelov*, 2. vyd., Univerzita Komenského, Bratislava, 2015

[20] Seber, G. A. F., Lee, A. J.: *Linear regression analysis*, 2. vyd., John Wiley & Sons, Hoboken, 2003

[21] Tibshirani, R.: *Regression shrinkage and selection via the LASSO.* Journal of the Royal Statistical Society. Series B (methodological) 58 (1996), 267–288

[22] Tibshirani, R., Taylor, J.: *Degrees of freedom in LASSO problems.* Annals of Statistics 40 (2012), 1198–1232

[23] Tibshirani, R. J.: *The LASSO problem and uniqueness.* Electronic Journal of Statistics 7 (2013), 1456–1490

[24] Tsatsomeros, M. J.: *Principal pivot transforms: properties and applications.* Linear Algebra and its Applications 307 (2000), 151–165

[25] Tucker, A. W.: *A combinatorial equivalence of matrices.* Proceedings of symposia in applied mathematics 10 (1960), 129–140

[26] Tucker, A. W.: *Principal pivotal transforms of square matrices.* SIAM Review 5 (1963), 305

[27] Zou, H., Hastie, T., Tibshirani, R.: *On the degrees of freedom of the LASSO.* Annals of Statistics 35 (2007), 2173–2192