



**MATEMATICKO-FYZIKÁLNÍ
FAKULTA**
Univerzita Karlova

DIPLOMOVÁ PRÁCE

Tomáš Masák

Velká data - extrakce klíčových informací pomocí metod matematické statistiky a strojového učení

Katedra pravděpodobnosti a matematické statistiky

Vedoucí diplomové práce: Prof. RNDr. Jaromír Antoch, CSc.

Studijní program: Matematika

Studijní obor: Pravděpodobnost, matematická statistika
a ekonometrie

Praha 2017

Prohlašuji, že jsem tuto diplomovou práci vypracoval(a) samostatně a výhradně s použitím citovaných pramenů, literatury a dalších odborných zdrojů.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona v platném znění, zejména skutečnost, že Univerzita Karlova má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle §60 odst. 1 autorského zákona.

V dne

Podpis autora

Rád bych zde poděkoval Prof. RNDr. Jaromíru Antochovi, CSc., nejen za jeho trefné poznámky a postřehy, které výraznou měrou přispěly ke zkvalitnění této práce, ale hlavně za jeho celkovou trpělivost a nesmírnou ochotu, se kterou mě vedl během mých dosavadních studiích.

Zároveň bych chtěl poděkovat svým nejbližším, bez jejichž podpory by tato práce nemohla vzniknout.

Název práce: Velká data - extrakce klíčových informací pomocí metod matematické statistiky a strojového učení

Autor: Tomáš Masák

Katedra: Katedra pravděpodobnosti a matematické statistiky

Vedoucí diplomové práce: Prof. RNDr. Jaromír Antoch, CSc., Katedra pravděpodobnosti a matematické statistiky

Abstrakt: Tato práce se zabývá metodami zpracování dat, zejména analýzou hlavních komponent a její řídkou modifikací (AŘHK), která je NP-těžkou úlohou. Úlohu AŘHK lze přepsat do regresního kontextu, ve kterém je řídkost typicky vynucována pomocí ℓ_1 -penalizace regresních koeficientů. V této práci navrhuje použít iterativní převažování ℓ_2 -penalizace namísto zmíněného ℓ_1 -přístupu. Výsledný algoritmus porovnáváme s několika známými algoritmy pro AŘHK pomocí simulační studie a také zajímavého praktického příkladu, ve kterém analyzujeme data o hlasování poslanců v Parlamentu České republiky. Experimentálně ukazujeme, že námi navržený algoritmus produkuje lepší výsledky než ostatní uvažované algoritmy. Pro navržený algoritmus uvádíme též důkaz konvergence spolu s důkazem konvergence původního regresního přístupu k AŘHK.

Klíčová slova: penalizace nejmenších čtverců, iterativně vážené nejmenší čtverce, analýza hlavních komponent, řídkost, algoritmy pro AŘHK

Title: Big data - extraction of key information combining methods of mathematical statistics and machine learning

Author: Tomáš Masák

Department: Department of probability and mathematical statistics

Supervisor: Prof. RNDr. Jaromír Antoch, CSc., Department of probability and mathematical statistics

Abstract: This thesis is concerned with data analysis, especially with principal component analysis and its sparse modification (SPCA), which is NP-hard-to-solve. SPCA problem can be recast into the regression framework in which sparsity is usually induced with ℓ_1 -penalty. In the thesis, we propose to use iteratively reweighted ℓ_2 -penalty instead of the aforementioned ℓ_1 -approach. We compare the resulting algorithm with several well-known approaches to SPCA using both simulation study and interesting practical example in which we analyze voting records of the Parliament of the Czech Republic. We show experimentally that the proposed algorithm outperforms the other considered algorithms. We also prove convergence of both the proposed algorithm and the original regression-based approach to PCA.

Keywords: penalized least squares, iteratively reweighted least squares, principal component analysis, sparsity, algorithms for SPCA

Obsah

Seznam zkratk	3
Seznam značení	4
Seznam obrázků	5
Seznam tabulek	7
Úvod	8
1 Analýza hlavních komponent	11
1.1 Populační hlavní komponenty	11
1.2 Výběrové hlavní komponenty	14
1.3 Hlavní komponenty a nejmenší čtverce	15
1.4 Hlavní komponenty a faktorová analýza	17
1.5 Asymptotické vlastnosti	20
1.6 Shrnutí	23
2 Penalizace nejmenších čtverců	24
2.1 Hřebenová regrese	29
2.2 Lasso	31
2.3 Adaptivní lasso	32
2.4 Elastická síť	34
2.5 LARS algoritmus pro lasso	35
2.6 Analýza hlavních komponent pomocí penalizované regrese	37
2.7 Shrnutí	40
3 Analýza řídkých hlavních komponent	42
3.1 Deflační metody	45
3.2 Důležitost opory nábojů	46
3.3 Rozptyl vysvětlený řídkými komponentami	47
3.4 Algoritmy	49
3.4.1 Jednoduché prahování (ST-PCA)	49
3.4.2 Hladový algoritmus (Greedy-PCA)	50
3.4.3 Regresní algoritmus (S-PCA)	51

3.4.4	Adaptivní modifikace (ADA-S-PCA)	53
3.5	Shrnutí	54
4	Nový algoritmus pro analýzu řídkých hlavních komponent	55
4.1	Nekonvexní penalizace nejmenších čtverců	55
4.2	Nová modifikace regresního algoritmu (IRLS-S-PCA)	61
4.3	Spojítost s existující literaturou	62
4.4	Globální konvergence algoritmů	64
4.4.1	Nekonvexní penalizace nejmenších čtverců	64
4.4.2	S-PCA algoritmus	67
4.4.3	IRLS-S-PCA algoritmus	70
4.5	Praktická implementace	71
4.6	Shrnutí	72
5	Simulační studie	74
5.1	Normální špičatý model s jednou komponentou	74
5.2	Normální špičatý model s dvěma komponentami	79
5.3	Přesná kovarianční matice	83
5.4	Shrnutí	85
6	Analýza reálných dat: hlasování poslanců	87
6.1	Popis dat	88
6.2	Volba počtu komponent a kardinalit nábojů	89
6.3	Výsledky	91
6.3.1	Tři hlavní komponenty	94
6.4	Shrnutí	96
	Diskuze	97
	Závěr	99
	Seznam použité literatury	101
	Dodatky	106
	A. Poznámky ke značení	106
	B. Užité maticová algebra	107

Seznam zkratek

ADA-S-PCA	adaptivní modifikace S-PCA ze sekce 3.4.4
AHK	analýza hlavních komponent
AŘHK	analýza řídkých hlavních komponent
FPR	podíl špatně odhadnutých nenulových koeficientů (<i>false negative rate</i>)
Greedy-PCA	hladový algoritmus ze sekce 3.4.2
HK	hlavní komponenty
HP	hlavní podprostor
IRLS	iterativní vážené nejmenší čtverce (<i>iteratively reweighted least squares</i>)
IRLS-S-PCA	algoritmus ze sekce 4.2
KKT	Karushovy-Kuhnovy-Tuckerovy (podmínky optimality)
MNČ	metoda nejmenších čtverců
PCA	analýza hlavních komponent (<i>principal component analysis</i> , použito výhradně v grafických výstupech)
SDP	algoritmus d'Aspremonta a kol. (2007, <i>semidefinite programming</i>)
S-PCA	algoritmus ze sekce 3.4.3
ST-PCA	algoritmus iterativního prahování (<i>simple thresholding</i>)
TNR	podíl správně odhadnutých nulových koeficientů (<i>true negative rate</i>)
TPR	podíl správně odhadnutých nenulových koeficientů (<i>true positive rate</i>)

Seznam značení

\triangle	konec příkladu
\square	konec důkazu
\diamond	konec poznámky
$:=$	symbol pro lokální ztotožnění
\approx	přibližně rovno
\equiv	je ekvivalentní
$\mathbf{1}_{[i=j]}$	operátor rovný jedné pro $i = j$ a rovný nule jinak
$[p]$	množina indexů $\{1, \dots, p\}$ pro $p \in \mathbb{N}$
$[v_i]_{i=1}^p$	vektor $\mathbf{v} = (v_1, \dots, v_p)$ pro $v_1, \dots, v_p \in \mathbb{R}$
$\nabla_{\mathbf{x}} f(\mathbf{x}, \mathbf{y})$	vektor parciálních derivací funkce f podle složek proměnné \mathbf{x}
$\ \mathbf{v}\ _q$	ℓ_q -norma vektoru \mathbf{v} (viz definice 2.2)
$\ \mathbf{X}\ _F$	Frobeniova norma matice \mathbf{X}
\mathbf{A}_c	ortonormální doplněk matice \mathbf{A} (viz lemma B.2)
\mathbf{a}_i	i -tý populační náboj
$\hat{\mathbf{a}}_i$	i -tý výběrový náboj
$\tilde{\mathbf{a}}_i$	i -tý řídký náboj
β^*	skutečná hodnota regresního parametru β
$\hat{\beta}_{OLS}$	odhad regresního parametru β metodou nejmenších čtverců
$\hat{\beta}_R$	odhad regresního parametru β metodou hřebenové regrese
$\hat{\beta}_L$	odhad regresního parametru β lasso metodou
$\hat{\beta}_{EN}$	odhad regresního parametru β metodou elastické sítě
$\hat{\beta}_{AL}$	odhad regresního parametru β adaptivní lasso metodou
$\hat{\beta}_0$	odhad regresního parametru β s použitím ℓ_0 -penalizace
$\hat{\beta}_{\log}$	odhad regresního parametru β s použitím nekonvexní logaritmické penalizace ρ_{\log}
$\text{diag}(c_i)_{i=1}^r$	diagonální matice s prvky $c_1, \dots, c_r \in \mathbb{R}$ na diagonále
$\text{cov}(X, Y)$	kovariance náhodných veličin X a Y
\mathbf{D}	matice singulárních čísel
d_i	i -té singulární číslo, tj. i -tý diagonální prvek matice \mathbf{D}
$\mathbb{E}X$	střední hodnota náhodné veličiny X
\mathbf{I}_k	matice typu $k \times k$ s jednotkami na diagonále a nulami mimo ni
K	počet hledaných (řídkých) hlavních komponent
l	vlastní čísla, zpravidla kovarianční matice Σ
$\lambda ; \lambda_{\cdot} ; \lambda_{\cdot\cdot}$	ladící parametry různých úloh a algoritmů
n	počet pozorování
\mathbb{N}_0	množina přirozených čísel rozšířená o nulu
$\mathcal{N}(\boldsymbol{\mu}, \Sigma)$	mnohorozměrné normální rozdělení se střední hodnotou $\boldsymbol{\mu} \in \mathbb{R}^p$ a kovarianční maticí $\Sigma \in \mathbb{R}^{p \times p}$
p	počet proměnných
ρ	penalizační funkce
ρ_{\log}	nekonvexní penalizační funkce definovaná v (4.2)
$\hat{\Sigma}$	výběrová kovarianční matice (viz definice 1.6)
$\text{tr}(\mathbf{A})$	stopa čtvercové matice \mathbf{A}
$\text{var}(X)$	rozptyl náhodné veličiny X

Seznam obrázků

1.1	<i>Vlevo:</i> Populační HK v dvourozměrném příkladu 1.3. <i>Vpravo:</i> Výběrové HK v dvourozměrném příkladu 1.8 určené z náhodného výběru rozsahu $n = 100$	13
1.2	Úhly mezi prvními dvěma náboji a jejich odhady metodou AHK (značeny α_1, α_2) a také úhel mezi skutečným a odhadnutým hlavním podprostorem dimenze 2 (značen A). Krabicové grafy jsou spočteny ze 100 nezávislých tahů popsaném v příkladu 1.17. . . .	22
2.1	Řešení penalizované lineární soustavy rovnic (2.8) pro různé ℓ_q -normy užití jako penalizační funkce v situaci z příkladu 2.5. Souřadné osy odpovídají koeficientům β_1 a β_2	27
2.2	Hranice množin $\{\ \beta\ _q^q \leq t\}$ pro hodnoty $t = 1, \frac{5}{4}, 2$ a $q = 1, \frac{1}{2}, \frac{3}{10}$	28
2.3	<i>Vlevo:</i> Odhad metodou hřebenové regrese získaný řešením úlohy (2.14) s $t = 1$. <i>Vpravo:</i> Lasso odhad získaný řešením úlohy (2.23) s $t = 1$. Řešením je v obou případech bod ležící v přípustné oblasti dotýkající se vrstevnice reziduálního součtu čtverců ležící co nejbližší k MNČ odhadu.	31
2.4	Ilustrace k příkladu 2.11, situace je obdobná jako na obr. 2.3. <i>Vlevo:</i> řešení lasso úlohy pro volbu $t = 1$ a optimální volbu $t = 1.5$. <i>Vpravo:</i> adaptivní lasso s počátečním lasso odhadem s $t = \frac{5}{2}$	34
2.5	Řešení penalizované úlohy nejmenších čtverců (2.11). <i>Vlevo:</i> $\ell_{\frac{3}{2}}$ -penalizace. <i>Vpravo:</i> Penalizace elastické sítě $\rho(\cdot) := a \ \beta\ _1 + (1 - a)\ \beta\ _2^2$ pro $a = \frac{1}{2}$	35
4.1	<i>Vlevo:</i> Ilustrace k příkladu 4.1 se dvěma regresory, kde $\delta = 10^{-2}$ a $t = \log(\delta)/2$. <i>Vpravo:</i> Hranice množin $\{\rho_{\log}(\beta) \leq t\}$ pro $\delta = 10^{-1}, 10^{-2}, 10^{-3}$ a $t = \log(\delta)/2$	56
4.2	Ilustrace iteračního schématu (4.6)-(4.7) na jednoduchém příkladu s dvěma korelovanými regresory.	59
4.3	Ilustrace k příkladu s dvěma regresory. <i>Vlevo:</i> Řešení jednoduchého lineárního modelu přímky s pouze prvním (řešení $\hat{\beta}_0$) respektive druhým (řešení $\tilde{\beta}_0$) regresorem. Přerušovaná přímka je parametrizována pomocí $\eta \in \mathbb{R}$ implicitním předpisem $\eta\hat{\beta}_0 + (1 - \eta)\tilde{\beta}_0 = 0$. <i>Vpravo:</i> Na této přímce je vykreslena účelová funkce úlohy (4.3) pro $\lambda = 10$ pro různé hodnoty δ	60
5.1	Křivky znázorňující počet správně odhadnutých nenulových koeficientů (TPR) proti počtu špatně odhadnutých nenulových koeficientů na základě jednoho tahu z modelu (5.1) pro volbu $\theta = 5$. Křivka ve vyšší úrovni znamená lepší chování algoritmu.	76

5.2	Průměrná plocha pod křivkou pro křivky znázorněné na obrázku 5.1. Větší plocha pod křivkou znamená lepší chování algoritmu. Pro každé $\theta = 3, 4, 5, 6, 7$ jsou dané průměry spočteny ze 100 nezávislých tahů z modelu 5.1.	77
5.3	Krabicové diagramy pro úhly mezi skutečným nábojem \mathbf{v} a jeho odhady spočtené různými algoritmy. Nižší úhly znamenají lepší chování algoritmu. Každý krabicový graf je vytvořen na základě 100 nezávislých tahů.	78
5.4	Rozptyl vysvětlený řídkými HK dělený rozptylem vysvětleným obyčejných HK. Vyšší podíl vysvětleného rozptylu znamená lepší chování algoritmu. Pro $n = 100$, $\theta = 5$ a všechny hodnoty kardinality $s = 1, \dots, 200$ je udán relativní vysvětlený rozptyl jako průměr přes 100 nezávislých tahů z modelu (5.1).	79
5.5	Krabicové diagramy úhlu mezi skutečným a odhadnutým hlavním podprostorem dimenze 2. Menší úhel znamená lepší chování daného algoritmu. Každý diagram je tvořen 100 nezávislými tahy z modelu (5.2). Na rozdíl od tabulky 5.2 je zde uveden i úhel pro obyčejnou AHK (PCA).	81
5.6	Krabicové diagramy relativního vysvětleného rozptylu pomocí prvních dvou řídkých HK, tj. součet rozptylů vysvětlených prvními dvěma řídkými HK dělený součtem rozptylů vysvětlených pomocí prvních dvou obyčejných HK. Vyšší hodnoty znamenají lepší chování daného algoritmu.	82
5.7	Chování různých algoritmů v závislosti na parametru γ . Pruhy vyznačují, pro jaké hodnoty γ se dané algoritmy chovají „optimálně“, tj. dávají odhady nábojů přesně tak, jak jsou udány v posledních dvou sloupcích tabulky 5.3.	85
6.1	Podíl celkového rozptylu vysvětlený každou z prvních deseti obyčejných HK pro data hlasování poslanců.	90
6.2	Projekce hlasování poslanců do hlavního podprostoru dimenze 2 pomocí AHK a různých algoritmů pro AŘHK.	92
6.3	Projekce hlasování poslanců do hlavního podprostoru dimenze 2 pomocí S-PCA a IRLS-S-PCA, avšak první a druhý náboj jsou nalezeny sekvenčně s využitím Hotellingovy deflace mezi hledáním prvního a druhého řídkého náboje.	93
6.4	Projekce hlasování poslanců do hlavního podprostoru dimenze 3 pomocí IRLS-S-PCA algoritmu. Pro lepší přehlednost je graf rozšířen a jsou dány také dvou-dimenzionální projekce do rovin, které jsou tvořeny první a druhou a také druhou a třetí řádkou HK. . .	95

Seznam tabulek

5.1	Směrodatné odchylky pro různé veličiny a různé algoritmy na základě 100 nezávislých tahů o 100 pozorováních z modelu (5.1) pro $\theta = 5$. <i>Plocha pod křivkou</i> je spočtena integrací přes všechny hodnoty kardinality s . Směrodatné odchylky pro <i>množství vysvětleného rozptylu</i> a <i>úhel mezi nábojem a jeho odhadem</i> jsou udány pro správnou volbu kardinality $s = 50$	77
5.2	Výběrové mediány spočteny ze 100 nezávislých tahů. Mediány jsou udány pro úhly mezi náboji \mathbf{v}_1 a \mathbf{v}_2 a jejich řídkými odhady $\tilde{\mathbf{v}}_1$ a $\tilde{\mathbf{v}}_2$ a také pro úhel mezi skutečným a odhadnutým hlavním podprostorem (HP), dále pro procentuální podíl správně odhadnutých nulových koeficientů (TNR) a pro relativní procentuální podíl vysvětleného rozptylu jednotlivými řídkými náboji a oběma současně (tedy $\Sigma_{1,2}$ je součet rozptylů vysvětlených pomocí $\tilde{\mathbf{v}}_1$ a $\tilde{\mathbf{v}}_2$ dělený součtem rozptylů vysvětlených pomocí prvních dvou obyčejných HK).	80
5.3	První dva (populační) náboje modelu 5.3 odhadnuté různými metodami: obyčejnou AHK (značeno PCA), jednoduchým prahováním (ST-PCA) a ostatními metodami. Ostatní metody jsou: S-PCA, ADA-S-PCA, IRLS-S-PCA a Greedy-PCA.	84
6.1	Politická situace v České republice v roce 2015.	89

Úvod

V této práci se soustředíme na modelování velkých dat a s tím spojené výzvy pro mnohorozměrnou statistickou analýzu. Vzhledem k rapidnímu růstu datových skladů a rozmachu informačních technologií se oblast zpracování a analýzy dat v současnosti těší značné pozornosti. V praxi se setkáváme s tím, že stále více a více znaků je měřeno na každém jednotlivci a ve výsledných datech je nakonec výrazně více proměnných (měřených znaků) než pozorování (jednotlivců). Typickým příkladem je genetika nebo molekulární biologie, kde jsou pro každého jednotlivce měřeny miliony genů. Dalším příkladem je zpracování signálu nebo konkrétněji zpracování obrazu, ve kterém každý snímek obsahuje miliony pixelů. V této mnohorozměrné situaci vyvstává řada problémů se zpracováním dat. Problémy se navíc stále prohlubují, jelikož podíl genomu měřeného pomocí genetických čipů stejně jako rozlišení obrazu se nepřestávají dále zvyšovat.

Mnohorozměrná statistika tudíž prochází dramatickým vývojem, přizpůsobující se velmi vysokému počtu parametrů. Je-li počet proměnných vyšší než počet pozorování, naráží klasické statistické metody na problém takzvaného prokletí vysoké dimenze. Pokud je dimenze dat vysoká, konstantní množství šumu v jednotlivých dimenzích se z hlediska uvažované úlohy typicky kumuluje. Standardním příkladem je odhad kovarianční matice. Pokoušíme-li se odhadnout kovarianční matici na základě n pozorování z p -rozměrného rozdělení (například normálního), potřebujeme odhadnout řádově p^2 parametrů. Je-li p vyšší než n , může být výběrová kovarianční matice špatným odhadem skutečné kovarianční matice. Dalším příkladem je úloha odhadu regresních parametrů ve standardním lineárním modelu. Z pohledu malého fixního p a relativně vyššího n je tento model v mnohých aplikacích bráný za příliš restriktivní. V mnohorozměrné situaci při $n > p$ je naopak příliš flexibilní a tato flexibilita může vést k velmi špatnému odhadu regresních koeficientů. Tento jev nazýváme *overfitting*.

Klasická statistická inference se tedy v případě $p \gg n$ neobejde bez dodatečných předpokladů. Nejčastějším předpokladem je *řídkost* uvažovaných modelů, která přirozeně vyvstává v mnoha aplikacích. V této práci pod pojmem řídkost rozumíme, že mnoho z odhadovaných parametrů je ve skutečnosti nulových, což je nejčastější, nikoliv však jediný pohled na řídkost. Z pohledu regresního modelu předpoklad řídkosti vyjadřuje očekávání, že mnoho proměnných nemá přímý vliv na odezvu. V případě odhadu kovarianční matice předpoklad řídkosti znamená, že mnoho proměnných je vzájemně nekorelovaných. Řídkost je vcelku přirozeným předpokladem v mnoha aplikacích. Zkoumáme-li například genetický původ některých chorob, přítomnost dané nemoci jistě nezávisí na celém měřeném genomu. Přestože ani s předpokladem řídkosti dopředu nevíme, které parametry jsou nulové, řídkost se ukazuje být „statistikovou spásou v jinak beznadějných úlohách hledání mnohorozměrného odhadu“ (Berthett a Rigollet, 2013).

Bohužel, mnohé výpočetně jednoduché procedury se za dodatečného předpokladu řídkosti stávají výpočetně náročnými. Zatímco odhad parametrů v normálním lineárním modelu lze snadno nalézt v polynomiálním čase, při omezení počtu nenulových koeficientů se úloha stává NP-těžkou. V takovém případě se jedná

o známý problém současného výběru a odhadu proměnných v regresním modelu, jehož přesné řešení vyžaduje kombinatorické hledání přes všechny příslušné velké podmnožiny proměnných. Odhad parametrů spojený s výběrem relevantních proměnných v regresním modelu hraje ve statistickém modelování ústřední roli, jelikož mnoho úloh na něj lze převést.

My se v této práci věnujeme analýze hlavních komponent (AHK) za dodatečného předpokladu řídkosti, kterou lze také v jistém smyslu převést na regresní úlohu současného výběru a odhadu proměnných. Soustředíme se tedy na propojení AHK s penalizačními metodami pro lineární model, které běžně slouží k řešení úlohy současného výběru a odhadu regresních parametrů.

Cílem AHK je komprese dat neboli redukce dimenze. Mnohdy se pokoušíme o výrazné snížení dimenze z důvodu vizualizace. Tato redukce dat spočívá v nalezení několika nekorelovaných lineárních kombinací proměnných (takzvaných hlavních komponent, HK), které zachycují co nejvíce rozptylu v datech. AHK je velmi populární a důležitá metoda pro zpracování dat. Nachází uplatnění prakticky ve všech vědních odvětvích, ve kterých se vyskytují mnohorozměrná data. AHK je běžně používána při zpracování signálu, v meteorologii, politologii, financích a mnoha dalších odvětvích. Příklady použití této metody zahrnují nejružnější aplikace strojového učení, jakými jsou systémy pro rozpoznávání tváře, klasifikace textu, odstranění šumu ve zvukovém záznamu a mnoho dalších. Z pohledu statistiky slouží AHK často jako metoda předzpracování dat pro následnou shlukovou analýzu, klasifikaci nebo regresi. Vyčerpávající přehled využitelnosti AHK není možné poskytnout.

V mnohorozměrné situaci vyvstávají dva zásadní problémy AHK. Zaprvé, každá HK je typicky lineární kombinací všech původních proměnných, což komplikuje interpretaci samotných výstupů i případných datových analýz, které mohou následovat. Tento interpretační problém AHK je relevantní i z pohledu středně velkého počtu proměnných. Zadruhé, z mnohorozměrného asymptotického pohledu, kdy počet proměnných p jde do nekonečna rychleji než počet pozorování n , je AHK takzvaně silně nekonzistentní. Praktická implikace tohoto teoretického poznatku je taková, že v případě $p > n$ nelze považovat výsledky získané AHK za příliš spolehlivé.

Předpoklad řídkosti může zlepšit situaci i v případě AHK. Analýza řídkých hlavních komponent (AŘHK) předpokládá, že každá HK je řídká, tj. je kombinací pouze malého počtu proměnných. Předpoklad řídkosti rozšiřuje případy, kdy je metoda konzistentní, a z praktického pohledu významně zlepšuje interpretovatelnost AHK. Nicméně, zatímco obyčejná AHK je jednoduchý problém řešitelný v polynomiálním čase, AŘHK je NP-těžký problém. Proto bylo v uplynulých letech vyvinuto velké množství algoritmů, které se snaží najít sub-optimální řešení této těžké úlohy. V této práci se věnujeme právě hledání řešení AŘHK pomocí penalizačních metod pro regresní model. Stavíme přitom na faktu, že AHK lze převést (ve smyslu, který bude specifikován později) na regresní problém, ve kterém lze užít pestré škály penalizačních metod k vynucení řídkosti.

Naše práce je koncipována následovně. V první kapitole připomínáme metodu AHK a ve druhé kapitole uvádíme přehled často používaných penalizačních metod. Ve třetí kapitole diskutujeme předpoklad řídkosti a jeho implikace pro AHK

a také představujeme několik algoritmů pro AŘHK. Zatímco první tři kapitoly jsou kompilační, vlastní přínos autora je obsažen ve zbylých třech kapitolách. Ve čtvrté kapitole představujeme nekonvexní penalizační funkci a tuto penalizaci využíváme v regresním schématu pro AHK. Výsledkem je nový algoritmus pro AŘHK, jehož konvergenci dokazujeme v téže kapitole. V páté kapitole provádíme simulační studii, ve které porovnáváme námi navržený algoritmus s existujícími algoritmy diskutovanými ve třetí kapitole. Konečně, šestá kapitola sestává z aplikace studovaných algoritmů na reálná data, konkrétně na hlasování poslanců Parlamentu České republiky.

1. Analýza hlavních komponent

Analýza hlavních komponent (AHK) je známá metoda změny proměnných, kterou nejčastěji užíváme za účelem snížení celkového počtu proměnných za cenu co nejmenší ztráty informace. Reprezentace dat v prostoru nižší dimenze je klíčová nejen pro vizualizaci, ale také pro zefektivnění případné následující datové analýzy.

Cílem AHK je nalézt lineární kombinace původních proměnných (tzv. *hlavní komponenty*, HK), které jsou nekorelované a zároveň vysvětlují co největší podíl rozptylu v datech. Pokud si ponecháme pouze prvních několik HK a vyřadíme zbylá data, projektujeme tím původní data do podprostoru nízké dimenze, který aproximuje původní data (tzv. *hlavní podprostor*). Jelikož se AHK soustředí na rozptyl v datech, úzce souvisí s (výběrovou) kovarianční maticí; koeficienty lineárních kombinací, které určují HK, jsou vlastní vektory (výběrové) kovarianční matice. Tyto vektory tvoří ortonormální bázi hlavního podprostoru.

Přestože původ AHK sahá až na počátek dvacátého století ke Karlu Pearsonovi (1901), tato metoda nabývá s rozmachem strojového učení na stále větší důležitosti, a různá její rozšíření jako analýza funkcionálních hlavních komponent nebo analýza řídkých hlavních komponent jsou dosud rozvíjena.

Jak uvidíme v této kapitole, existuje několik ekvivalentních způsobů zavedení a hledání hlavních komponent. V případě HK budeme rozlišovat mezi *populačními* HK a *výběrovými* HK, na které lze v jistém smyslu nahlížet jako na odhady těch populačních. Přídomky *populační* a *výběrové* přitom vypouštíme, pokud je z kontextu jasné, které z HK máme na mysli.

Ohledně přijímaných konvencí upozorňujeme čtenáře na sekci A dodatků, kde je vysvětleno používané značení jako doplněk k seznamu použitých zkratk.

1.1 Populační hlavní komponenty

Z populačního hlediska jsou HK standardizované nekorelované lineární kombinace původních proměnných, které mají po řadě maximální možný rozptyl.

Definice 1.1. Necht X_1, \dots, X_p jsou náhodné veličiny, a $\Sigma \in \mathbb{R}^{p \times p}$ je kovarianční matice náhodného vektoru $\mathcal{X} = (X_1, \dots, X_p)^\top$. *Populační hlavní komponenty* jsou lineární kombinace proměnných $Z_1 = \mathbf{a}_1^\top \mathcal{X}, \dots, Z_p = \mathbf{a}_p^\top \mathcal{X}$, pro které platí

$$\mathbf{a}_1 = \arg \max_{\mathbf{w} \in \mathbb{R}^p} \mathbf{w}^\top \Sigma \mathbf{w} \quad \text{s.t.} \quad \|\mathbf{w}\|_2 = 1$$

a

$$\mathbf{a}_k = \arg \max_{\mathbf{w} \in \mathbb{R}^p} \mathbf{w}^\top \Sigma \mathbf{w} \quad \text{s.t.} \quad \|\mathbf{w}\|_2 = 1, \\ \mathbf{w}^\top \mathbf{a}_j = 0, \quad j = 1, \dots, k-1,$$

pro $k = 2, \dots, p$. Vektory $\mathbf{a}_1, \dots, \mathbf{a}_p$ se nazývají *populační náboje*.

Nyní si ukážeme, že HK lze hledat pomocí *spektrálního rozkladu* kovarianční matice (viz věta B.3 v dodatcích).

Věta 1.2. Nechť \mathcal{X} je náhodný vektor s kovarianční maticí $\Sigma \in \mathbb{R}^{p \times p}$. Nechť $\Sigma = \mathbf{A}\mathbf{L}\mathbf{A}^\top$ je spektrální rozklad kovarianční matice. Pak *vlastní vektory* $\mathbf{a}_1, \dots, \mathbf{a}_p$ jsou hledané populační náboje.

Důkaz. Nechť $Z_1 = \mathbf{a}_1^\top \mathcal{X}, \dots, Z_p = \mathbf{a}_p^\top \mathcal{X}$. Z definice vlastních vektorů platí $\mathbf{a}_j^\top \mathbf{a}_k = \mathbb{1}_{[j=k]}$. Dále, pro $j \in [p]$ platí

$$\text{var}(Z_j) = \text{var}(\mathbf{a}_j^\top \mathcal{X}) = \mathbf{a}_j^\top \text{var}(\mathcal{X}) \mathbf{a}_j = \mathbf{a}_j^\top \Sigma \mathbf{a}_j = l_j \mathbf{a}_j^\top \mathbf{a}_j = l_j. \quad (1.1)$$

Tím máme nekorelovanost HK a správné pořadí nábojů (po řadě se snižující rozptyly). Zbývá ověřit maximalitu.

Nechť $k \in [p]$ a $\mathbf{c} \in \mathbb{R}^p$ je takové, že $\mathbf{c}^\top \mathbf{c} = 1$, a $\mathbf{c}^\top \mathcal{X}$ je nekorelované s $\mathbf{a}_1^\top \mathcal{X}, \dots, \mathbf{a}_{k-1}^\top \mathcal{X}$. Chceme ukázat, že $\text{var}(\mathbf{c}^\top \mathcal{X}) = \mathbf{c}^\top \Sigma \mathbf{c}$ je maximalizován pro $\mathbf{c} = \mathbf{a}_k$.

Vyjádříme \mathbf{c} vzhledem k ortogonální bázi \mathbf{A} jako

$$\mathbf{c} = \alpha_1 \mathbf{a}_1 + \dots + \alpha_p \mathbf{a}_p = \mathbf{A} \boldsymbol{\alpha},$$

kde $\boldsymbol{\alpha} \in \mathbb{R}^p$ a $\sum_{i=1}^p \alpha_i^2 = 1$, takže $\mathbf{c}^\top \mathbf{c} = 1$.

Potom

$$\text{var}(\mathbf{c}^\top \mathcal{X}) = \mathbf{c}^\top \Sigma \mathbf{c} = \mathbf{c}^\top \mathbf{A} \mathbf{L} \mathbf{A}^\top \mathbf{c}.$$

Jelikož $\mathbf{c}^\top \mathbf{A} = \boldsymbol{\alpha}^\top \mathbf{A}^\top \mathbf{A} = \boldsymbol{\alpha}^\top$, dostáváme $\text{var}(\mathbf{c}^\top \mathcal{X}) = \sum_{i=1}^p l_i \alpha_i^2$.

Aby bylo $\mathbf{c}^\top \mathcal{X}$ nekorelované s $\mathbf{a}_1^\top \mathcal{X}, \dots, \mathbf{a}_{k-1}^\top \mathcal{X}$, musí být $\alpha_i = 0$ pro $i \in [k-1]$.

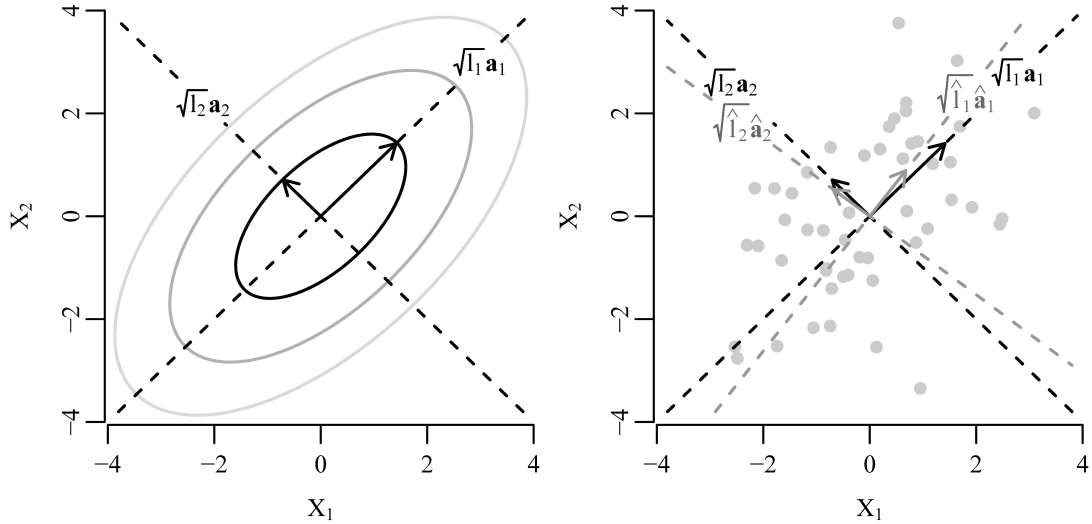
Navíc, $\sum_{i=1}^p \alpha_i^2 = 1$ a $l_1 \geq \dots \geq l_p$, tudíž rozptyl $\text{var}(\mathbf{c}^\top \mathcal{X}) = \sum_{i=1}^p l_i \alpha_i^2 = \sum_{i=k}^p l_i \alpha_i^2$ je maximalizován pro volbu $\alpha_k = 1$ a $\alpha_i = 0$ pro $i \neq k$. Tím pádem $\mathbf{c} = \mathbf{a}_k$. \square

Příklad 1.3. Nechť $\mathcal{X} \sim \mathcal{N}(0, \Sigma)$ s kovarianční maticí

$$\Sigma = \begin{pmatrix} 5/2 & 3/2 \\ 3/2 & 5/2 \end{pmatrix}. \quad (1.2)$$

Vlastní vektory Σ a tudíž i náboje jsou $\mathbf{a}_1 = (\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}})^\top$ a $\mathbf{a}_2 = (-\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}})^\top$, vlastní čísla jsou $l_1 = 4$ a $l_2 = 1$. Na obrázku 1.1 vlevo jsou vyznačeny vrstevnice (ve tvaru elips) hustoty daného dvourozměrného normálního rozdělení. Náboje tvoří osy těchto elips. První náboj je hlavní osa elipsy. Reprezentuje směr největšího rozptylu, tj. směr, ve kterém hustota normálního rozdělení klesá nejpomaleji. Jak je vyznačeno na obrázku, relativní délka hlavní a vedlejší osy každé elipsy je dána odmocninami vlastních čísel $\sqrt{l_1}$ a $\sqrt{l_2}$. \triangle

Poznámka 1.4. Jak je vidět již z definice 1.1, u nábojů nezáleží na znaménku. Jsou-li vlastní čísla kovarianční matice různá, jsou vlastní vektory určeny jednoznačně až na znaménko, a to samé tedy platí i pro náboje a HK. Pokud vlastní čísla nejsou různá, náboje přirozeně dědí nejednoznačnost vlastních vektorů; ta



Obrázek 1.1: *Vlevo:* Populační HK v dvourozměrném příkladu 1.3. *Vpravo:* Výběrové HK v dvourozměrném příkladu 1.8 určené z náhodného výběru rozsahu $n = 100$.

se pak přenáší i na HK. Dále, náboje ani HK nejsou invariantní vzhledem k lineárním transformacím dat. Konkrétně, HK vypočtené z kovarianční a korelační matice se obecně liší. Toto je známá nepříjemná vlastnost AHK, jelikož interpretace AHK se může značně lišit například v závislosti na použitém měřítku. My však budeme vždy pracovat s kovarianční maticí a pro detailní diskuzi problému odkážeme na Jolliffe (2002). \diamond

Poznámka 1.5. Když nyní víme, že náboje jsou vlastní vektory kovarianční matice, lze snadno ukázat, že hlavní komponenty jsou nekorelované:

$$\text{cov}(Z_i, Z_j) = \text{cov}(\mathbf{a}_i^\top \mathcal{X}, \mathbf{a}_j^\top \mathcal{X}) = \mathbf{a}_i^\top \text{var}(\mathcal{X}) \mathbf{a}_j = \mathbf{a}_i^\top \Sigma \mathbf{a}_j = l_j \mathbf{1}_{[i=j]}.$$

\diamond

Důvodem hledání populačních HK je typicky snaha o reprezentaci mnoho-rozměrné náhodné veličiny v prostoru nižší dimenze. V situaci z definice 1.1 se přirozeně nabízí reprezentovat náhodný vektor $\mathcal{X} \in \mathbb{R}^p$ pomocí náhodného vektoru $\mathcal{Z} = (Z_1, \dots, Z_K)^\top \in \mathbb{R}^K$ pro nějaké $K < p$. Otázkou je, jak volit K .

V praxi se často setkáváme s volbami $K = 1$ (jednorozměrný signál pozorovaný v jisté bázi), $K = 2$ nebo $K = 3$ z důvodu vykreslení dat. Z rovností (1.1) vyplývá, že rozptyl vysvětlený j -tou hlavní komponentou je roven j -tému vlastnímu číslu kovarianční matice a snižuje se se zvyšujícím se j . Pokud tedy nemáme nějaké zvláštní požadavky na volbu K , můžeme volit K jedním z následujících způsobů.

1. Postupně zvyšuj K dokud prvních K hlavních komponent nevysvětluje dostatečný podíl celkového rozptylu, tj. dokud $(l_1 + \dots + l_K)/(l_1 + \dots + l_p)$ není blízké jedné.
2. Postupně zvyšuj K dokud rozptyl vysvětlený $(K + 1)$ -ní HK není zanedbatelný, tj. dokud $l_{K+1}/(l_1 + \dots + l_p)$ není blízké nule. S touto strategií souvisí slavný *elbow* graf, viz Jolliffe (2002), sekce 6.1.3.

1.2 Výběrové hlavní komponenty

Nyní mějme matici pozorovaných dat $\mathbf{X} \in \mathbb{R}^{n \times p}$. Výběrové hlavní komponenty jsou definovány analogicky jako jejich populační verze. Jsou to nekorelované lineární kombinace sloupců matice \mathbf{X} , které mají po řadě maximální možný rozptyl. Formálně lze výběrové HK definovat jako populační HK s tím, že skutečná kovarianční matice je nahrazena za výběrovou.

Definice 1.6. Necht $\mathbf{X} = (\mathbf{x}^{(1)} | \dots | \mathbf{x}^{(n)})^\top$ je centrovaná matice dat a $\widehat{\Sigma} = \frac{1}{n} \mathbf{X}^\top \mathbf{X}$ je výběrová kovarianční matice. *Výběrové hlavní komponenty* jsou lineární kombinace pozorovaných vektorů $\mathbf{z}_1 = \mathbf{X}\widehat{\mathbf{a}}_1, \dots, \mathbf{z}_p = \mathbf{X}\widehat{\mathbf{a}}_p$, pro které platí

$$\widehat{\mathbf{a}}_1 = \arg \max_{\mathbf{w} \in \mathbb{R}^p} \mathbf{w}^\top \widehat{\Sigma} \mathbf{w} \quad \text{s.t.} \quad \|\mathbf{w}\|_2 = 1$$

a

$$\begin{aligned} \widehat{\mathbf{a}}_k &= \arg \max_{\mathbf{w} \in \mathbb{R}^p} \mathbf{w}^\top \widehat{\Sigma} \mathbf{w} \quad \text{s.t.} \quad \|\mathbf{w}\|_2 = 1, \\ &\quad \mathbf{w}^\top \widehat{\mathbf{a}}_j = 0, \quad j = 1, \dots, k-1, \end{aligned}$$

pro $k = 2, \dots, p$. Vektory $\widehat{\mathbf{a}}_1, \dots, \widehat{\mathbf{a}}_p$ se nazývají *výběrové náboje*.

Poznámka 1.7. Náboje jsou normované a tedy můžeme v optimalizačních úlohách v předchozí definici psát $\mathbf{X}^\top \mathbf{X}$ místo $\widehat{\Sigma}$. Ze stejného důvodu nezáleží na tom, zda výběrovou kovarianční matici $\widehat{\Sigma}$ definujeme jako $\frac{1}{n} \mathbf{X}^\top \mathbf{X}$ nebo $\frac{1}{n-1} \mathbf{X}^\top \mathbf{X}$. \diamond

Označení výběrových nábojů stejnými písmeny, jakými jsme značili v předchozí sekci populační náboje, pouze se stříškou navíc, vyjadřuje, že na výběrové náboje se díváme jako na odhady nábojů populačních. Ve stejném duchu budeme značit spektrální rozklad výběrové kovarianční matice jako $\widehat{\Sigma} = \widehat{\mathbf{A}} \widehat{\mathbf{L}} \widehat{\mathbf{A}}^\top$. Věta 1.2 zůstává platná i pro výběrové HK, pouze se spektrálním rozkladem výběrové kovarianční matice. Podobně poznámka 1.4 je platná i pro výběrové HK a výběrové náboje.

Příklad 1.8. Necht $\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(n)}$ je náhodný výběr o rozsahu $n = 100$ z rozdělení $\mathcal{N}(0, \Sigma)$, kde Σ je jako v příkladu 1.3. Spektrálním rozkladem výběrové kovarianční matice $\widehat{\Sigma} = \frac{1}{n} \mathbf{X}^\top \mathbf{X}$ lze opět určit náboje. Viz obrázek 1.1 pro konkrétní realizaci. \triangle

Výběrové HK jsou opět nekorelované. Důvodem hledání HK je snaha o reprezentaci mnohorozměrných dat z \mathbb{R}^p v prostoru nižší dimenze K , na což lze nahlížet také jako na kompresi dat. Nejčastějším motivem je přitom vizualizace dat v prostoru dimenze $K = 2$ nebo $K = 3$. Pro jiné než vizualizační účely probíhá volba K analogicky jako pro populační HK podle vysvětleného rozptylu.

Čím se však výběrové HK liší od populačních, je to, že není třeba vůbec pracovat s kovarianční maticí, lze si vystačit pouze s maticí dat \mathbf{X} . Toto je výhodné zejména v případě $p > n$, kdy může být kovarianční matice obrovská. Namísto spektrálního rozkladu kovarianční matice lze pracovat se singulárním rozkladem matice dat (viz věta B.4 v dodatcích). Singulární rozklad nabízí (ve výběrovém nastavení) efektivní způsob, jak nalézt HK, jejich rozptyly i náboje v jednom.

Lemma 1.9. Necht $\mathbf{X} \in \mathbb{R}^{n \times p}$, $\text{rank}(\mathbf{X}) = r \in \mathbb{N}$ a necht $\mathbf{X} = \mathbf{U}\mathbf{D}\mathbf{V}^\top$, $\mathbf{U} \in \mathbb{R}^{n \times r}$, $\mathbf{D} = \text{diag}(d_i)_{i=1}^r \in \mathbb{R}^{r \times r}$, $\mathbf{V} \in \mathbb{R}^{p \times r}$ je singulární rozklad matice \mathbf{X} . Pak pro prvních r výběrových hlavních komponent platí $\mathbf{z}_i = d_i \mathbf{u}_i$ a $\text{var}(\mathbf{z}_i) = d_i^2/n$ pro $i = 1, \dots, r$, a $\mathbf{v}_1, \dots, \mathbf{v}_r$ jsou příslušné náboje. Navíc, je-li $r < p$, jsou zbylé hlavní komponenty nulové.

Důkaz. Necht $\widehat{\Sigma} = \widehat{\mathbf{A}}\widehat{\mathbf{L}}\widehat{\mathbf{A}}^\top$ je spektrální rozklad matice $\widehat{\Sigma}$. Ze singulárního rozkladu matice \mathbf{X} dostáváme zároveň

$$\widehat{\Sigma} = \frac{1}{n} \mathbf{X}^\top \mathbf{X} = \frac{1}{n} \mathbf{V}\mathbf{D}\mathbf{U}^\top \mathbf{U}\mathbf{D}\mathbf{V}^\top = \mathbf{V} \text{diag} \left(\frac{1}{n} d_i^2 \right)_{i=1}^r \mathbf{V}^\top.$$

Tedy je-li $r = p$, musí být přímo $\widehat{\mathbf{A}} = \mathbf{V}$ a $\widehat{\mathbf{L}} = \mathbf{D}$. Je-li $r < p$, musí platit $\widehat{l}_{r+1} = \dots = \widehat{l}_p = 0$, tudíž musí být alespoň $\widehat{\mathbf{a}}_i = \mathbf{v}_i$ a $\widehat{l}_i = d_i^2/n$ pro $i = 1, \dots, r$. Zbývá tedy ověřit nulovost zbylých HK. Pro $i = 1, \dots, r$ máme

$$\mathbf{z}_i = \mathbf{X}\widehat{\mathbf{a}}_i = \mathbf{U}\mathbf{D}\mathbf{V}^\top \mathbf{v}_i = \mathbf{U}\mathbf{D}\mathbf{e}_i = d_i \mathbf{U}\mathbf{e}_i = d_i \mathbf{u}_i.$$

Navíc, je-li $r < p$, pak pro $i = r+1, \dots, p$ platí $\widehat{\mathbf{a}}_i \perp \mathbf{v}_j$ pro všechna $j < i$, a tudíž

$$\mathbf{z}_i = \mathbf{X}\widehat{\mathbf{a}}_i = \mathbf{U}\mathbf{D}\mathbf{V}^\top \widehat{\mathbf{a}}_i = \mathbf{U}\mathbf{D}\mathbf{0} = \mathbf{0}.$$

Tím je důkaz hotov. □

1.3 Hlavní komponenty a nejmenší čtverce

Pro data $\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(n)} \in \mathbb{R}^p$ uvažujme aproximaci

$$\mathbf{x}^{(i)} \approx \boldsymbol{\mu} + \mathbf{Q}\boldsymbol{\nu}_i, \quad i = 1, \dots, n,$$

kde $\boldsymbol{\mu} \in \mathbb{R}^p$, $\mathbf{Q} \in \mathbb{R}^{p \times k}$ je *ortonormální* matice, a $\boldsymbol{\nu}_i \in \mathbb{R}^k$ pro $i = 1, \dots, n$. Cílem je tedy nalézt k -dimenzionální aproximaci našich p -dimenzionálních dat. Jako míru kvality aproximace zvolíme reziduální součet čtverců.

Naším cílem je najít vektor $\boldsymbol{\mu}$, ortonormální matici \mathbf{Q} , a koeficienty $\boldsymbol{\nu}_i$ takové, aby reziduální součet čtverců

$$\sum_{i=1}^n \|\mathbf{x}^{(i)} - (\boldsymbol{\mu} + \mathbf{Q}\boldsymbol{\nu}_i)\|_2^2 \tag{1.3}$$

byl co nejmenší. Bez újmy na obecnosti předpokládejme, že data jsou vycentrována a tudíž $\boldsymbol{\mu} = \mathbf{0}$. Potom chceme nalézt \mathbf{Q} jako řešení úlohy

$$\arg \min_{\mathbf{Q} \in \mathbb{R}^{p \times k}, \boldsymbol{\nu}_i \in \mathbb{R}^k} \sum_{i=1}^n \|\mathbf{x}^{(i)} - \mathbf{Q}\boldsymbol{\nu}_i\|_2^2 \quad \text{s.t.} \quad \mathbf{Q}^\top \mathbf{Q} = \mathbf{I}. \tag{1.4}$$

Derivace účelové funkce (1.4) podle $\boldsymbol{\nu}_i$ je

$$\mathbf{Q}^\top (\mathbf{x}^{(i)} - \mathbf{Q}\boldsymbol{\nu}_i).$$

Položíme-li tuto derivaci rovnou nule, dostáváme s využitím ortonormality \mathbf{Q} , že

$$\mathbf{Q}^\top \mathbf{x}^{(i)} = \mathbf{Q}^\top \mathbf{Q} \boldsymbol{\nu}_i = \boldsymbol{\nu}_i. \quad (1.5)$$

Dosazením rovnosti $\boldsymbol{\nu}_i = \mathbf{Q}^\top \mathbf{x}^{(i)}$ zpět do účelové funkce z (1.4) dostáváme úlohu pouze pro \mathbf{Q} :

$$\arg \min_{\mathbf{Q} \in \mathbb{R}^{p \times k}, \boldsymbol{\nu}_i \in \mathbb{R}^k} \sum_{i=1}^n \|\mathbf{x}^{(i)} - \mathbf{Q} \mathbf{Q}^\top \mathbf{x}^{(i)}\|_2^2 \quad \text{s.t.} \quad \mathbf{Q}^\top \mathbf{Q} = \mathbf{I}. \quad (1.6)$$

Matice $\mathbf{Q} \mathbf{Q}^\top$ je projekční matice do prostoru s bází tvořenou sloupci \mathbf{Q} . Následující lemma říká, že tyto sloupce jsou rovny nábojům.

Lemma 1.10. Necht $\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(n)} \in \mathbb{R}^p$, necht $K \leq \text{rank}(\mathbf{X})$, a $\mathbf{X} = \mathbf{U} \mathbf{D} \mathbf{V}^\top$ je singulární rozklad matice $\mathbf{X} = (\mathbf{x}^{(1)} | \dots | \mathbf{x}^{(n)})^\top$. Pak řešením úlohy (1.6) je $\widehat{\mathbf{Q}} = (\mathbf{v}_1 | \dots | \mathbf{v}_K)^\top$.

Důkaz. S využitím *Frobeniovovy normy* a operátoru *stopy matice* (viz definice B.6 a poznámka B.7) můžeme psát

$$\begin{aligned} \sum_{i=1}^n \|\mathbf{x}^{(i)} - \mathbf{Q} \mathbf{Q}^\top \mathbf{x}^{(i)}\|_2^2 &= \|\mathbf{X} - \mathbf{X} \mathbf{Q} \mathbf{Q}^\top\|_F^2 \\ &= \text{tr} \left((\mathbf{X} - \mathbf{X} \mathbf{Q} \mathbf{Q}^\top)^\top (\mathbf{X} - \mathbf{X} \mathbf{Q} \mathbf{Q}^\top) \right) \\ &= \text{tr}(\mathbf{X}^\top \mathbf{X}) - 2 \text{tr}(\mathbf{X}^\top \mathbf{X} \mathbf{Q} \mathbf{Q}^\top) + \text{tr}(\mathbf{Q} \mathbf{Q}^\top \mathbf{X}^\top \mathbf{X} \mathbf{Q} \mathbf{Q}^\top) \\ &= \text{tr}(\mathbf{X}^\top \mathbf{X}) - \text{tr}(\mathbf{Q}^\top \mathbf{X}^\top \mathbf{X} \mathbf{Q}). \end{aligned}$$

Namísto minimalizace účelové funkce z (1.6) tedy můžeme maximalizovat funkci $\text{tr}(\mathbf{Q}^\top \mathbf{X}^\top \mathbf{X} \mathbf{Q})$. S využitím singulárního rozkladu navíc dostáváme

$$\text{tr}(\mathbf{Q}^\top \mathbf{X}^\top \mathbf{X} \mathbf{Q}) = \text{tr}(\mathbf{Q}^\top \mathbf{V} \mathbf{D} \mathbf{U}^\top \mathbf{U} \mathbf{D} \mathbf{V}^\top \mathbf{Q}) = \text{tr}(\mathbf{Q}^\top \mathbf{V} \mathbf{D}^2 \mathbf{V}^\top \mathbf{Q}). \quad (1.7)$$

Nyní by mělo být vidno, že \mathbf{Q} maximalizující poslední člen v (1.7) je právě $\widehat{\mathbf{Q}} = (\mathbf{v}_1 | \dots | \mathbf{v}_K)^\top$. Pro tuto volbu \mathbf{Q} je hodnota posledního členu v (1.7) rovna $\sum_{i=1}^K d_i^2$. Vzhledem k uspořádání singulárních čísel $d_1 > d_2 > \dots$ a ortonormalitě \mathbf{Q} přitom zřejmě vyšší hodnoty dosáhnout nelze. \square

Všimněme si, že úloha (1.6) je též ekvivalentní následující úloze aproximace matice maticí nízké hodnosti:

$$\arg \min_{\mathbf{X}_* \in \mathbb{R}^{n \times p}} \|\mathbf{X} - \mathbf{X}_*\|_F^2 \quad \text{s.t.} \quad \text{rank}(\mathbf{X}_*) = K. \quad (1.8)$$

Dle Eckartovy-Youngovy-Minského věty (viz věta B.8 v dodatcích) je řešení úlohy (1.8) dáno oříznutím singulárního rozkladu matice \mathbf{X} , tj. $\widehat{\mathbf{X}}_* = \sum_{i=1}^K d_i \mathbf{u}_i \mathbf{v}_i^\top$.

Naopak pro úlohu (1.6) víme již z lemmatu 1.10, že $\widehat{\mathbf{Q}} = (\mathbf{v}_1 | \dots | \mathbf{v}_K)^\top$. Hlavní komponenty pak snadno obdržíme přenásobením nábojů maticí \mathbf{X} tak jako v definici 1.6.

Poznámka 1.11. Uvažujme nyní speciální případ úlohy (1.8) pro $K = 1$, který lze zapsat jako

$$\min_{\mathbf{w} \in \mathbb{R}^p} \|\mathbf{X}^\top \mathbf{X} - \mathbf{w}\mathbf{w}^\top\|_F^2. \quad (1.9)$$

Jelikož

$$\begin{aligned} \|\mathbf{X}^\top \mathbf{X} - \mathbf{w}\mathbf{w}^\top\|_F^2 &= \text{tr} \left((\mathbf{X}^\top \mathbf{X} - \mathbf{w}\mathbf{w}^\top)^\top (\mathbf{X}^\top \mathbf{X} - \mathbf{w}\mathbf{w}^\top) \right) \\ &= \text{tr}(\mathbf{X}^\top \mathbf{X} \mathbf{X}^\top \mathbf{X}) - 2 \text{tr}(\mathbf{X}^\top \mathbf{X} \mathbf{w}\mathbf{w}^\top) + \text{tr}(\mathbf{w}\mathbf{w}^\top \mathbf{w}\mathbf{w}^\top) \\ &= \text{tr}(\mathbf{X}^\top \mathbf{X} \mathbf{X}^\top \mathbf{X}) - 2\mathbf{w}^\top \mathbf{X}^\top \mathbf{X} \mathbf{w} + \|\mathbf{w}\|_2^4, \end{aligned}$$

úloha (1.9) je ekvivalentní úloze

$$\min_{\mathbf{w} \in \mathbb{R}^p} -\mathbf{w}^\top \mathbf{X}^\top \mathbf{X} \mathbf{w} + \frac{1}{2} \|\mathbf{w}\|_2^4. \quad (1.10)$$

Všimněme si, že kdybychom nyní vynásobili účelovou funkci (1.10) členem $1/n$ a přešli z neomezené optimalizační formulace k omezené zavedením omezení $\|\mathbf{w}\|_2 = t$, dostali bychom přesně úlohu hledání první výběrové HK (1.6) z definice 1.6. Přitom z úlohy (1.10) je nejsnáze vidět, že všechny tyto vzájemně ekvivalentní úlohy jsou nekonvexní. Singulární rozklad však umožňuje řešit tyto nekonvexní úlohy v čase $O(np^2)$ (viz např. Golub a Van Loan, 2012). \diamond

Úlohu hledání hlavních komponent lze tedy motivovat hledáním podprostoru nižší dimenze, který co nejlépe aproximuje data ve smyslu součtu čtverců, a také hledáním aproximace matice dat maticí nízké hodnoti. V následující sekci si ukážeme spojitost analýzy hlavních komponent s jinou velmi populární metodou mnohorozměrné statistiky, *faktorovou analýzou*.

1.4 Hlavní komponenty a faktorová analýza

Na AHK je často nahlíženo jako na speciální případ faktorové analýzy. Takový náhled je však dle Jolliffa (2002) zavádějící. V této sekci budeme diskutovat tento kontroverzní vztah mezi AHK a faktorovou analýzou.

Cílem faktorové analýzy je vyjádřit pozorovaný náhodný vektor $\mathcal{X} \in \mathbb{R}^p$ jako lineární kombinaci nepozorovaných (*latentních*) náhodných veličin f_1, \dots, f_K , tzv. *faktorů*. Model lze psát jako

$$\mathcal{X} - \boldsymbol{\mu} = \mathbf{W}\mathcal{F} + \boldsymbol{\epsilon}, \quad (1.11)$$

kde $\mathcal{X} \in \mathbb{R}^p$, $\mathbf{W} \in \mathbb{R}^{p \times K}$ je matice pevných koeficientů, $\mathcal{F} = (f_1, \dots, f_K)^\top \in \mathbb{R}^K$, $\boldsymbol{\epsilon} \in \mathbb{R}^p$ je chybový člen, $K \leq p$, $\mathbb{E}\mathcal{X} = \boldsymbol{\mu}$ a $\mathbb{E}\mathcal{F} = 0 = \mathbb{E}\boldsymbol{\epsilon} = 0$. S těmito předpoklady je model (1.11) obecnou verzí *modelu latentních proměnných*, jehož je faktorová analýza speciálním případem. Další předpoklady faktorové analýzy jsou: $\text{var}(\mathcal{F}) = \mathbf{I}_K$, \mathcal{F} a $\boldsymbol{\epsilon}$ jsou nezávislé, a $\text{var}(\boldsymbol{\epsilon}) = \boldsymbol{\Psi}$ je diagonální matice. S těmito předpoklady je

$$\boldsymbol{\Sigma} \equiv \text{var}(\mathcal{X}) = \mathbf{W}\mathbf{W}^\top + \boldsymbol{\Psi}. \quad (1.12)$$

V modelu (1.11), \mathbf{W} spojuje latentní proměnné \mathcal{F} s pozorovanými proměnnými \mathcal{X} . Přitom pozorované proměnné jsou podmíněně nekorelované při daných latentních proměnných. Smysl latentních proměnných je tedy vysvětlení kovarianční struktury pozorovaných proměnných. Tento záměr je vidět i z rovnosti (1.12), v níž je hlavní význam přiřkládán matici \mathbf{W} , zatímco Ψ je libovolná diagonální matice. Tedy faktorová analýza se přirozeně zaměřuje na reprezentaci prvků kovarianční matice Σ ležících mimo diagonálu, zatímco objektem zájmu AHK jsou rozptyly, tj. diagonální prvky Σ (Jolliffe, 2002).

Na druhou stranu, pokud odhlédneme od rozdílnosti záměrů AHK a faktorové analýzy, AHK lze chápat jako speciální případ faktorové analýzy. Zesilme nyní dosavadní předpoklad na varianční matici ϵ a požadujme $\text{var}(\epsilon) = \sigma^2 \mathbf{I}_p$, kde $\sigma^2 > 0$ je známé, a navíc předpokládejme, že $\boldsymbol{\mu} = 0$ a rozdělení \mathcal{F} a ϵ jsou normální. Tím pádem dostáváme

$$\mathcal{X} \sim \mathcal{N}(0, \mathbf{W}\mathbf{W}^\top + \sigma^2 \mathbf{I}_p), \quad (1.13)$$

kde matice \mathbf{W} je jedinou neznámou. Pro přehlednost shrňme v následující definici všechny předpoklady našeho „faktorového modelu pro AHK“, který formálněji pojmenujme *izotropní normální faktorový model*.

Model 1.12. (*Izotropní normální faktorový model.*)

Nechť $\mathbf{W} \in \mathbb{R}^{p \times K}$ je matice pevných koeficientů. Nechť pro náhodné vektory $\mathcal{X} \in \mathbb{R}^p$, $\mathcal{F} \in \mathbb{R}^K$ a $\epsilon \in \mathbb{R}^p$, $K \leq p$, platí

$$\mathcal{X} = \mathbf{W}\mathcal{F} + \epsilon, \quad (1.14)$$

kde

- $\mathcal{F} \sim \mathcal{N}(0, \mathbf{I}_K)$,
- $\epsilon \sim \mathcal{N}(0, \sigma^2 \mathbf{I}_p)$, $\sigma^2 > 0$,
- \mathcal{F} a ϵ jsou nezávislé.

Potom model (1.14) nazýváme *izotropní normální faktorový model*.

V modelu 1.12 platí (1.13). Lze ukázat, že odhad matice \mathbf{W} metodou maximální věrohodnosti v modelu 1.12 je dán předpisem

$$\widehat{\mathbf{W}} = \widehat{\mathbf{A}}_K (\widehat{\mathbf{L}}_K - \sigma^2 \mathbf{I}_K)^{\frac{1}{2}} \mathbf{R}, \quad (1.15)$$

kde $\widehat{\mathbf{A}}_K = (\widehat{\mathbf{a}}_1 | \dots | \widehat{\mathbf{a}}_K)$ je matice s prvními K vlastními vektory výběrové kovarianční matice $\widehat{\Sigma}$ jako sloupci, $\mathbf{R} \in \mathbb{R}^{K \times K}$ je libovolná ortonormální matice (tj. rotační matice), a $\widehat{\mathbf{L}}_K = \text{diag}(\widehat{l}_i)_{i=1}^K$ je diagonální matice s prvními K vlastními čísly na diagonále. Poznamenejme, že tato vlastní čísla nemohou být menší než σ^2 , jelikož $\widehat{l}_i - \sigma^2$, $i = 1, \dots, K$, jsou odhady vlastních čísel pozitivně semidefinitní matice $\mathbf{W}\mathbf{W}^\top$.

Maximálně věrohodný odhad (1.15) zde nebudeme odvozovat, zájemce odkazujeme na Tippinga a Bishopa (1999). Okomentujme však intuitivní spojitost našeho modelu s daným odhadem. Na rovnost (1.14) lze nahlížet jako na zašuměné zobrazení z prostoru nepozorovaných proměnných $\mathcal{F} \in \mathbb{R}^K$ do prostoru

pozorovaných proměnných $\mathcal{X} \in \mathbb{R}^p$. Předpoklad nezávislosti jednotlivých faktorů f_1, \dots, f_K zajišťuje, že odhad $\widehat{\mathbf{W}}$ je ortogonální matice. Stejně jako v případě AHK je ekvivalentní nekorelovanost HK a ortogonalita nábojů (viz poznámka 1.5), v případě modelu (1.14) nelze rozlišit mezi ortogonalitou \mathbf{W} a nezávislostí faktorů. Odhad $\widehat{\mathbf{W}}$ jakožto ortogonální matice je tedy škálovaná báze nějakého K -dimenzionálního podprostoru (sloupcového prostoru $\widehat{\mathbf{W}}$, tj hlavního podprostoru) pozorovaných proměnných \mathbb{R}^p . Nicméně náš model nijak nepreferuje žádnou konkrétní bázi v hlavním podprostoru. Např. jelikož model neklade žádné požadavky na uspořádání sloupců \mathbf{W} a faktory jsou zcela zaměnitelné, všechny permutace sloupců nějaké konkrétní matice musejí být stejně věrohodné. Obecněji, rotační invariance předpokládaného rozdělení faktorů má za následek stejnou věrohodnost rotovaných řešení. Metoda maximální věrohodnosti se tedy snaží spíše o odhadnutí hlavního podprostoru než nějaké jeho konkrétní báze¹. Všechny možné báze jsou zároveň vůči sobě rotovány ortonormálními maticemi $\mathbf{R} \in \mathbb{R}^{K \times K}$, předpis (1.15) tedy určuje celou třídu bází odhadnutého hlavního podprostoru.

Roweis (1998) poukazuje na to, že AHK lze vnímat jako maximálně věrohodný odhad v limitním modelu 1.12 pro $\sigma^2 \rightarrow 0$, tj.

$$\widehat{\mathbf{W}} = \widehat{\mathbf{A}}_K \widehat{\mathbf{L}}_K^{\frac{1}{2}} \mathbf{R}.$$

Bez ohledu na volbu rotační matice \mathbf{R} je

$$\widehat{\mathbf{W}}\widehat{\mathbf{W}}^\top = \widehat{\mathbf{A}}_K \widehat{\mathbf{L}}_K^{\frac{1}{2}} \widehat{\mathbf{A}}_K^\top,$$

jelikož $\mathbf{R}\mathbf{R}^\top = \mathbf{R}^\top\mathbf{R} = \mathbf{I}_K$. Odhad je tedy přesně nejlepší aproximace výběrové kovarianční matice $\widehat{\mathbf{\Sigma}}$ maticí hodnosti K podle Eckartovy-Youngovy-Minského věty. AHK lze tedy vnímat jako limitní případ faktorového modelu 1.12.

AHK, případně úloha aproximace matice maticí nízké hodnosti (1.8) nebo ekvivalentně úloha (1.3) nepředpokládají, že by pozorování $\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(n)}$ byla přesná, tj. bez chybového členu. Nijak se však nesnaží odlišit rozptyl způsobený tímto chybovým členem od rozptylu způsobeného faktory, které navíc explicitně nejsou vůbec uvažovány. AHK se tedy implicitně chová tak, jakoby byl všechen šum v datech kolmý k hlavním komponentám. Roweis (1998) tudíž navrhuje tzv. *rozumnou AHK (sensible principal component analysis)* založenou na normálním faktorovém modelu diskutovaném výše s neznámou mírou šumu σ^2 . Tato disproporce v přístupu k šumu je podle našeho názoru hlavním důvodem, proč by AHK neměla být vnímána jako speciální případ faktorové analýzy.

Na druhou stranu, na AHK můžeme stále nahlížet jako na pokus o odhadnutí ortonormální báze sloupcového prostoru matice \mathbf{W} , tj. prostoru do kterého jsou projektovány faktory (které nás třeba nemusejí vůbec zajímat). O AHK tedy můžeme volně přemýšlet jako o metodě popisu jistých faktorů v doméně pozorovaných dat. Jak uvidíme v kapitole 6 na příkladu reálných dat, mnohdy bývá výhodné mít tuto interpretaci, která je založena na pravděpodobnostním modelu 1.12, na paměti kvůli interpretaci.

¹Jednou konkrétní bází hlavního podprostoru dimenze K , na níž jsou kladeny požadavky postupného snižování rozptylů ve směrech daných vektorů, je prvních K nábojů.

1.5 Asymptotické vlastnosti

Asymptotické vlastnosti výběrových hlavních komponent odpovídají asymptotickým vlastnostem vlastních čísel a vlastních vektorů výběrových kovariančních matic. Tyto vlastnosti jsou studovány z pohledu teorie náhodných matic a příliš se liší v závislosti na uvažovaném modelu, než abychom je zde detailně popisovali. Proto odkazujeme na Paul (2013) pro hezkou, i když zdaleka ne vyčerpávající diskuzi různých výsledků.

Většina asymptotických výsledků v minulém století (mezi standardní prameny patří např. Anderson, 1963) se zabývala limitní situací, ve které počet proměnných p je fixní a počet pozorování n se blíží nekonečnu. Z hlediska moderní mnohorozměrné statistiky je však mnohem důležitější situace sdružené limity, kdy $p, n \rightarrow \infty$ a $p/n \rightarrow \tau \in [0, \infty]$. Této situaci se nyní budeme věnovat za předpokladů následujícího modelu.

Model 1.13. (*Špičatý normální kovarianční model a konzistence.*)

Nechť $K \leq p$ a $\mathcal{X} \sim \mathcal{N}(0, \Sigma)$ s kovarianční maticí

$$\Sigma = \sum_{i=1}^K d_i^2 \mathbf{v}_i \mathbf{v}_i^T + \sigma^2 \mathbf{I}_p, \quad (1.16)$$

kde $d_1^2 > \dots > d_K^2 > 0$, $\sigma^2 > 0$ a $\mathbf{v}_1, \dots, \mathbf{v}_K \in \mathbb{R}^p$ tvoří ortonormální systém, tj. $\mathbf{v}_i^T \mathbf{v}_j = \mathbb{1}_{[i=j]}$. Potom tento model nazýváme *špičatý normální kovarianční model*.

Definice 1.14. Mějme dále náhodný výběr $\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(n)}$ z modelu (1.16). Říkáme, že AHK je *konzistentní*, pokud pro vlastní vektory $\hat{\mathbf{v}}_1, \dots, \hat{\mathbf{v}}_K$ výběrové kovarianční matice $\hat{\Sigma}$ platí $|\hat{\mathbf{v}}_i^T \mathbf{v}_i| \rightarrow 0$ v pravděpodobnosti pro $n \rightarrow \infty$ a $i = 1, \dots, K$.

Poznámka 1.15. Pod poněkud volným pojmem „AHK je konzistentní“ rozumíme konzistenci nábojů v klasickém smyslu, tj. konvergenci výběrových vlastních vektorů k jejich populačním protějškům v pravděpodobnosti (až na znaménko). Přitom tuto konvergenci lze požadovat jen tehdy, jsou-li příslušná vlastní čísla různá. Pokud jsou vlastní čísla příslušná dvěma nábojům stejná, lze odhadnout pouze příslušný podprostor dimenze 2, nikoliv samostatné náboje. Dále, v naší definici konzistence „příhodně“ zapomínáme na konvergenci vlastních čísel. Jejich odhad však zahrnuje odhad zanedbatelného počtu koeficientů vůči počtu koeficientů, které je třeba odhadnout pro náboje. \diamond

Špičatý normální kovarianční model je ekvivalentní izotropnímu normálnímu faktorovému modelu 1.12, pouze faktorové proměnné jsou z pohledu špičatého modelu upozaděny. Vektory $\mathbf{v}_1, \dots, \mathbf{v}_K$ z (1.16) tvoří ortonormální bázi sloupcového prostoru matice \mathbf{W} z (1.14) a jsou přesně prvními K náboji. Tudíž špičatý model je přirozeným modelem pro studium konzistence AHK a bylo mu z tohoto pohledu věnováno v literatuře mnoho prostoru (např. Johnstone a Lu, 2009, Paul, 2007 nebo Amini a Wainwright, 2008). My zde uvedeme pouze jeden konkrétní výsledek, který je velmi důležitý pro AHK.

Věta 1.16. (*Johnstone a Lu, 2009, věta 1.*)

Uvažujme trojúhelníkové schéma, ve kterém jsou pozorování $\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(n)}$ nezávisle tažena z p -dimenzionálního modelu 1.13 pro $K = 1$, a necht počet parametrů p , vedoucí vlastní vektor \mathbf{v}_1 , příslušné vlastní číslo d_1^2 , a hladina šumu σ^2 všechny závisejí na počtu pozorování n . Necht platí

$$\frac{d_1^2}{\sigma^2} \rightarrow \omega > 0 \quad \text{a} \quad \frac{p}{n} \rightarrow \tau \geq 0 \quad \text{pro} \quad n \rightarrow \infty$$

a necht $\hat{\mathbf{v}}_1$ je první vlastní vektor výběrové kovarianční matice, tedy odhad \mathbf{v}_1 . Potom

$$|\hat{\mathbf{v}}_1^\top \mathbf{v}_1| \rightarrow \frac{(\omega^2 - \tau)_+}{\omega^2 + \omega\tau} \quad \text{skoro jistě pro} \quad n \rightarrow \infty. \quad (1.17)$$

Za předpokladů věty 1.16 tedy platí s pravděpodobností jedna, že limitní skalární součin $\hat{\mathbf{a}}_1^\top \mathbf{v}_1 < 1$ právě tehdy, když $\tau > 0$. To jinými slovy znamená, že pokud v situaci sdružené limity, kdy $p, n \rightarrow \infty$ a $p/n \rightarrow \tau \in [0, \infty]$, není $\tau = 0$, AHK nemůže být konzistentní v modelu 1.13 pro $K = 1$. Věta 1.16 ve větší obecnosti říká, že úhel mezi výběrovým a populačním nábojem konverguje skoro jistě k deterministické limitě, která je nulová tehdy a jen tehdy, když $\tau = 0$ (limitní veličina v (1.17) je kosinus úhlu mezi výběrovým a populačním nábojem). Mnozí autoři referují na výsledky podobného typu jako na *silnou nekonzistenci* AHK.

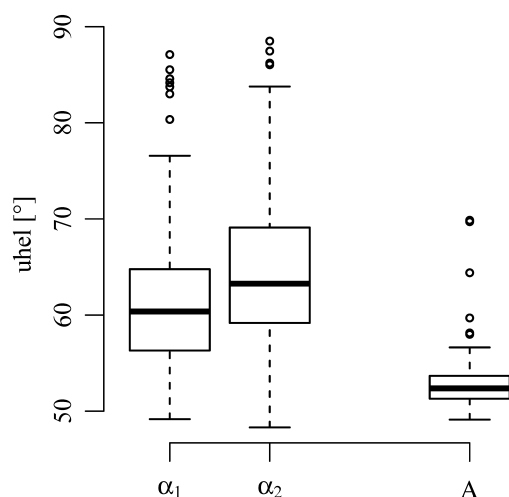
Paul (2007) později zobecnil výsledek věty 1.16 na špičatý normální kovarianční model pro $K > 1$. Zaujatého čtenáře také odkazujeme na článek Shena a kol. (2012), kteří jednotným jazykem popisují mnoho dosažených výsledků o nekonzistenci AHK, tyto výsledky částečně zobecňují a přinášejí srozumitelný rozbor konzistence a silné nekonzistence AHK v závislosti na uvažovaném modelu.

V mnohorozměrné limitní situaci ($p/n \rightarrow \tau > 0$) tedy není metoda AHK konzistentní. Praktická implikace tohoto poznatku je následující. Je-li počet proměnných p (což je zároveň počet koeficientů, které je třeba odhadnout pro každý samostatný náboj) srovnatelný s – nebo dokonce větší než – počet pozorování n , nelze považovat výběrové náboje za důvěryhodné odhady jejich populačních protějšků. Problém fakticky plyne z toho, že pro $p \approx n$ není výběrová kovarianční matice důvěryhodným odhadem populační kovarianční matice. Intuitivně se zdá být toto tvrzení zřejmé; nelze očekávat, že budeme na základě jakéhokoliv počtu pozorování schopni kvalitně odhadnout několikanásobně vyšší počet koeficientů. Pro ilustraci uvádíme následující příklad.

Příklad 1.17. Uvažujme špičatý normální kovarianční model pro $p = 500$, $n = 50$ a dvou hlavních komponentách. Necht první náboj \mathbf{v}_1 má nulové všechny složky s výjimkou složek $v_{1,1} = \dots = v_{1,10} = 1/\sqrt{10}$ a druhý náboj \mathbf{v}_2 má nulové všechny složky s výjimkou složek $v_{2,11} = \dots = v_{2,20} = 1/\sqrt{10}$. Necht $d_i^2 = 399$, $d_i^2 = 299$ a $\sigma^2 = 1$, tj. vlastní čísla příslušná prvním dvěma nábojům jsou 400 a 300, zatímco všechna zbylá vlastní čísla jsou rovna jedné.

Z tohoto modelu nasimulujeme $n = 50$ opakování, čímž získáváme matici dat $\mathbf{X} \in \mathbb{R}^{50 \times 500}$. Matici \mathbf{X} vycentrujme a odhadněme \mathbf{v}_1 a \mathbf{v}_2 pomocí prvních dvou pravých singulárních vektorů této vycentrované matice. Získané odhady označme $\hat{\mathbf{v}}_1$ a $\hat{\mathbf{v}}_2$. Nakonec spočtěme úhly mezi náboji a jejich odhady jako

$$\alpha_i = \arccos(|\mathbf{v}_i^\top \hat{\mathbf{v}}_i|), \quad i = 1, 2.$$



Obrázek 1.2: Úhly mezi prvními dvěma náboji a jejich odhady metodou AHK (značeny α_1 , α_2) a také úhel mezi skutečným a odhadnutým hlavním podprostorem dimenze 2 (značen A). Krabicové grafy jsou spočteny ze 100 nezávislých tahů popsáném v příkladu 1.17.

Uvedený postup jsme zopakovali nezávisle 100 krát. Na obrázku 1.2 jsou vyobrazeny spočítané úhly α_1 a α_2 převedeny do stupňů. Pro srovnání uvádíme také úhel A mezi skutečným a odhadnutým hlavním podprostorem² dimenze 2.

Přestože vlastní čísla příslušná prvním dvěma nábojům jsou vysoká a vlastní vektory by tudíž měly být dobře odhadnutelné, vysoká dimenze dat v kombinaci s malým počtem pozorování způsobuje, že výběrové náboje $\hat{\mathbf{v}}_1$ a $\hat{\mathbf{v}}_2$ jsou poměrně nespolehlivé odhady populačních nábojů \mathbf{v}_1 a \mathbf{v}_2 . Δ

V mnohorozměrné situaci při $p \approx n$ tedy není rozumné používat klasickou AHK. Chceme-li snižovat dimenzi dat ve smyslu hlavních komponent, je třeba zavést dodatečné strukturální požadavky. Nejčastějším strukturálním požadavkem je *řídkost* nábojů, tj. předpoklad, že velká část koeficientů je nulová a tudíž je třeba určit pozice malého počtu nenulových koeficientů a ty následně odhadnout. Tento předpoklad vede na analýzu řídkých hlavních komponent, kterými se budeme zabývat v kapitole 3.

²Tento úhel počítáme ve smyslu kanonické korelace mezi dvěma podprostory jako arkus kosinus nejmenší singulární hodnoty matice $(\mathbf{v}_1 | \mathbf{v}_1)(\hat{\mathbf{v}}_1 | \hat{\mathbf{v}}_2)^\top$, viz Björck a Golub (1973, sekce 3).

1.6 Shrnutí

V této kapitole jsme si představili klasickou analýzu hlavních komponent. Hlavní komponenty jsme si přitom zavedli několika způsoby:

1. jako lineární kombinace maximalizující výběrový rozptyl;
2. jako projekci pozorovaných bodů v \mathbb{R}^p do prostoru nižší dimenze s co nejmenší projekční chybou (danou součtem čtverců projekčních reziduí);
3. jako hledání aproximace matice dat maticí nízké hodnoti s co nejmenší Frobeniovou chybou.

Společným pojítkem pro tyto tři motivace HK přitom byl singulární rozklad matice dat. Poznamenejme, že motivaci 1 lze přisuzovat Hotellingovi (1933), od něhož patrně pochází označení hlavní komponenty (Jolliffe, 2002), zatímco motivaci 2 lze datovat až k Pearsonovi (1901).

Za AHK typicky nestojí žádný pravděpodobnostní model, viz definice 1.1 a 1.6. Z historického hlediska (Pearson, 1901) byly HK brány jako projekce dat v euklidovské geometrii, ve které je chyba projekce měřena reziduálním součtem čtverců. Nicméně, jak je obecně známo a jak jsme my ukázali při vyšetřování spojitosti AHK s faktorovou analýzou, užití MNČ implicitně předpokládá normální rozdělení pozorovaných dat.

Jak je vidět z diskuze v sekci 1.4, AHK přesahuje rámec statistické metody s pravděpodobnostním základem. Je to dáno patrně tím, že metoda je velmi populární v nejrůznějších vědních odvětvích. Přitom v mnohých z nich slouží výhradně jako metoda předzpracování dat za účelem další analýzy. Snažit se o interpretaci AHK z hlediska pravděpodobnostního modelu je tedy částečně krkolomné. Na druhou stranu, pravděpodobnostní interpretace může být často přínosná pro konkrétní data, jak uvidíme na analýze reálných dat v kapitole 6.

Nicméně to, že AHK bývá často (nejspíš většinou) používána bez přímého využití jejích statistických vlastností, ještě neznamená, že by tyto statistické vlastnosti neměly být brány v potaz a že by se AHK dala považovat za čistě numerickou metodu. Například výsledky o „ztracené“ konzistenci AHK ve sdruženém limitním případě $n, p \rightarrow \infty$ diskutované v sekci 1.5 důrazně naznačují například to, že pracujeme-li s daty, pro která $p \approx n$ nebo dokonce $p \gg n$, nelze považovat klasickou metodu AHK za příliš spolehlivou.

2. Penalizace nejmenších čtverců

V této kapitole připomeneme základní penalizační metody pro lineární model. Pro jednoduchost se omezujeme na práci s reziduálním součtem čtverců jakožto míry vhodnosti parametrického modelu pro daná data. Ani v případě uvažovaných penalizačních metod se nesnažíme o vyčerpávající popis všech vztahů a poznatků. To ostatně ani není možné vzhledem k faktu, že o některých penalizačních metodách byly napsány celé knihy. Tudíž budeme čtenáře během této kapitoly hojně odkazovat na existující literaturu. Náš přístup se přitom od citované literatury odlišuje kladením důrazu na geometrickou intuici a také na penalizaci pomocí ℓ_q -norem pro $q \in [0, 2]$, které částečně diskutujeme z pohledu inverzních problémů ve zpracování signálu.

Uvažujme klasickou úlohu odhadu vektoru regresních koeficientů $\beta^* \in \mathbb{R}^p$ v lineárním modelu

$$\mathbf{y} = \mathbf{X}\beta^* + \boldsymbol{\epsilon}, \quad (2.1)$$

kde $\mathbf{y} = (y_1, \dots, y_n)^\top$ je *odezva*, $\mathbf{X} \in \mathbb{R}^{n \times p}$ je *regresní matice*¹ a $\boldsymbol{\epsilon}$ je vektor *chyb* s nulovou střední hodnotou a kovarianční maticí $\sigma^2 \mathbf{I}_p$. Předpokládejme, že odezva \mathbf{y} i matice \mathbf{X} jsou centrované, není tudíž třeba uvažovat v modelu *intercept*.

Odhad regresních parametrů *metodou nejmenších čtverců* (MNČ) hledáme jako řešení optimalizační úlohy

$$\hat{\beta}_{OLS} = \arg \min_{\beta \in \mathbb{R}^p} \sum_{i=1}^n (y_i - (\mathbf{x}^{(i)})^\top \beta)^2 \equiv \arg \min_{\beta \in \mathbb{R}^p} \|\mathbf{y} - \mathbf{X}\beta\|_2^2. \quad (2.2)$$

Derivováním účelové funkce získáváme *soustavu normálních rovnic*

$$\mathbf{X}^\top \mathbf{X} \beta = \mathbf{X}^\top \mathbf{y}, \quad (2.3)$$

a jejím následným řešením dostáváme odhad

$$\hat{\beta}_{OLS} = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{y}, \quad (2.4)$$

je-li matice $\mathbf{X}^\top \mathbf{X} \in \mathbb{R}^{p \times p}$ invertibilní, což je právě tehdy, jsou-li sloupce matice regresorů \mathbf{X} lineárně nezávislé. V opačném případě má soustava (2.3) nekonečně mnoho řešení a pro jednoznačnost vyberme jedno z nich nahrazením neexistující inverze $(\mathbf{X}^\top \mathbf{X})^{-1}$ za *Mooreovou-Penroseovou pseudoinvertzi* $(\mathbf{X}^\top \mathbf{X})^\dagger$, viz Zvára (1989, sekce 2.4).

Poznámka 2.1. Častým předpokladem je, že $\boldsymbol{\epsilon}$ má normální rozdělení. Odhad parametru β^* metodou maximální věrohodnosti se pak shoduje s MNČ odhadem. My se zde pro jednoduchost vyhneme předpokladům o pravděpodobnostním rozdělení $\boldsymbol{\epsilon}$ a jako míru shody odhadnutého modelu s daty budeme používat účelovou funkci v (2.2), tzv. *reziduální součet čtverců*. Nicméně, všechny penalizační postupy popsané v této kapitole lze zobecnit na odhady metodou maximální věrohodnosti. \diamond

¹V celé této práci budeme uvažovat pevné regresory, tj. prvky matice \mathbf{X} jsou pevná čísla na které nenahlížíme jako na realizace nějakého rozdělení. Na tomto pohledu však nezáleží, protože nikde formálně nediskutujeme asymptotické vlastnosti v regresním modelu, které by mohly být pravděpodobnostním modelem na pozadí ovlivněny.

Bez ohledu na to, zda je matice $\mathbf{X}^T\mathbf{X}$ invertibilní, vektor *vyrovnaných hodnot*

$$\hat{\mathbf{y}} = \mathbf{X}\hat{\boldsymbol{\beta}}_{OLS} = \mathbf{X}(\mathbf{X}^T\mathbf{X})^\dagger\mathbf{X}^T\mathbf{y} =: \mathbf{H}\mathbf{y}, \quad (2.5)$$

je jednoznačně určenou projekcí odezvy do sloupcového prostoru regresní matice. Projekční matice \mathbf{H} se nazývá *hat matrix*. Lze snadno ukázat (Zvára, 1989, sekce 2.5), že optimální hodnotu reziduálního součtu čtverců v (2.2) lze vyjádřit pomocí *hat matrix* jako

$$\|\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}_{OLS}\|_2^2 = \mathbf{y}^T(\mathbf{I}_n - \mathbf{H})\mathbf{y}. \quad (2.6)$$

Je-li míra lineární závislosti mezi sloupci matice \mathbf{X} vysoká (což nutně nastává s počtem parametrů p srovnatelným s nebo větším než je počet pozorování n), je lineární model přeparametrizovaný a může docházet k jevu zvanému *overfitting*. Při tomto jevu je odhad parametru silně ovlivněn šumem v datech. Ne však proto, že by byla míra šumu σ^2 příliš velká, ale spíš kvůli tomu, že modelu je ponechána nadměrná volnost přizpůsobovat se jednotlivým pozorováním. V takovém případě nelze považovat odhad $\hat{\boldsymbol{\beta}}_{OLS}$ za důvěryhodný. Populárně si lze *overfitting* představit na návštěvníkovi opery, jehož sluch je tak jemný, že samotnou hudbu mu ruší každé drobné zašustění v sále.

V této kapitole si představíme několik způsobů regularizace úlohy (2.2) zavedením penalizace velikosti hledaných koeficientů do účelové funkce (2.2). Tato penalizace zajistí jednodušší model, čímž zlepší jeho interpretaci a potenciálně zamezí jevu *overfitting*.

Přesná měření ve správném modelu

Ještě než se dostaneme k zavedení penalizace do lineárního modelu (2.1), předpokládejme na okamžik, že je tento model platný s $\boldsymbol{\epsilon} = 0$. V takovém případě hledáme řešení soustavy lineárních rovnic

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta}. \quad (2.7)$$

Pokud je $\text{rank}(\mathbf{X}) < p$, je tato soustava špatně podmíněná a řešení může existovat nekonečně mnoho. V takovém případě má smysl hledat řešení v nějakém smyslu nejjednodušší. Jednotlivá řešení můžeme rozlišit pomocí penalizační funkce $\rho : \mathbb{R}^p \rightarrow \mathbb{R}$ a namísto (2.7) můžeme řešit úlohu

$$\hat{\boldsymbol{\beta}} = \arg \min_{\boldsymbol{\beta} \in \mathbb{R}^p} \rho(\boldsymbol{\beta}) \quad \text{s.t.} \quad \mathbf{y} = \mathbf{X}\boldsymbol{\beta}. \quad (2.8)$$

Penalizační funkce ρ vyjadřuje naše požadavky na jednoduchost řešení: řešením „jednoduchým“ přiřazuje nižší hodnoty než řešením „složitým“. Ještě než si tento mechanismus specifikujeme, zavedme si několik pojmů.

Definice 2.2. Pro vektor $\beta \in \mathbb{R}^p$ definujeme

- *oporu* $\text{supp}(\beta) = \{j \in [p]; \beta_j \neq 0\}$ jako množinu indexů nenulových koeficientů β ,
- ℓ_0 -normu $\|\beta\|_0 = \text{card}(\beta) = |\text{supp}(\beta)|$ jako počet nenulových koeficientů β , jinými slovy *kardinalitu* β ,
- ℓ_q -normu pro $q > 0$ jako $\|\beta\|_q = \left(\sum_{j=1}^p |\beta_j|^q \right)^{1/q}$.

Dále říkáme, že vektor β je *řídský*, pokud má malou kardinalitu.

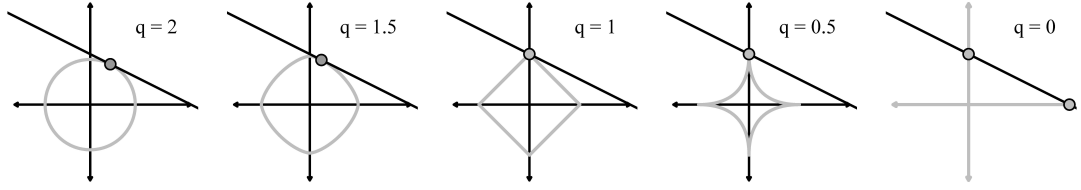
Poznámka 2.3. Pro $q \in (0,1)$ není $\|\cdot\|_q$ ve skutečnosti norma, jelikož nesplňuje trojúhelníkovou nerovnost (subaditivitu), tj. je to pouze kvazinorma. To samé platí pro ℓ_0 -normu, která navíc nesplňuje ani požadavek pozitivní homogenity při násobení skalárem. Mnozí autoři přesto běžně užívají označení „norma“ namísto „kvazinorma“, a pro kardinalitu vektoru se vžilo označení „norma“ díky Donohovi (2006). My také užíváme označení ℓ_q -norma pro $q \geq 0$ jako v definici 2.2. Pojem norma používáme zásadně pro vektorové prostory \mathbb{R}^p , nebude tedy docházet k žádné kolizi například s normami na L^p prostorech funkcí. \diamond

Díky výpočetní nenáročnosti penalizujeme nejčastěji ℓ_2 -normu, což vede na tzv. *Tikho-novovu regularizaci* (Tikhonov, 1963). Ve statistické literatuře je tato metoda známá pod pojmem *hřebenová regrese* (*ridge regression*, Hoerl a Kennard, 1970). Pro tuto volbu $\rho(\cdot) := \|\cdot\|_2^2$ má úloha (2.8) jednoznačné řešení. Koeficienty tohoto řešení jsou přitom „nuceny“ být malé, typicky však nenulové.

Naopak volba $\rho(\cdot) := \|\cdot\|_0$ penalizuje počet nenulových koeficientů, vybírá tedy mezi možnými řešeními ta řídská, ale naopak nijak nepenalizuje velikost koeficientů. Úlohu (2.8) chceme nejčastěji řešit právě s volbou $\rho(\cdot) := \|\cdot\|_0$, jelikož řídkost je patrně nejpřirozenější předpoklad na jednoduchost řešení. Řídká řešení také značně zjednodušují interpretovatelnost. Z toho důvodu je jejich hledání často cílem samo o sobě. Bohužel úloha (2.8) s $\rho(\cdot) := \|\cdot\|_0$ je NP-těžká, a její doposud známá řešení vyžadují kombinatorické hledání přes všechny možné opory β , kterých je 2^p .

Je obecně známo, že pro volby $\rho(\cdot) := \|\cdot\|_q^q$, $q > 1$, má úloha (2.8) jednoznačné řešení díky striktní konvexnosti účelové funkce, toto řešení však není řídké. Naopak pro $q \leq 1$ není řešení nutně jednoznačné, zato však má tendenci být řídké. Toto chování je způsobeno faktem, že pro $q > 1$ je ℓ_q -norma diferencovatelná, zatímco pro $q \leq 1$ není ℓ_q -norma diferencovatelná v $\beta_i = 0$ a odhad parametru β je přirozeně přitahován těmito singularitami („špičatými rohy“), viz obrázek 2.1. Pro detaily odkazujeme čtenáře např. na Murphyho (2012, sekce 13.3.1 a 13.3.2), který přehledně diskutuje případ $q = 1$, nebo na Fana a Liho (2006) pro obecnější diskuzi.

Velmi důležitou volbou penalizační funkce je ℓ_1 -norma, $\rho(\cdot) = \|\cdot\|_1$. Zatímco pro $q < 1$ je úloha (2.8) nekonvexní, pro $q > 1$ nebývají nalezená řešení řídká. Případ $q = 1$ je tedy jediný, který upřednostňuje řídká řešení a zároveň je výpočetně nenáročný. Úloha (2.8) s volbou $\rho(\cdot) := \|\cdot\|_1$ lze snadno přepsat na úlohu lineárního programování v $2p$ proměnných (stačí uvažovat kladnou a zápornou část každé proměnné).



Obrázek 2.1: Řešení penalizované lineární soustavy rovnic (2.8) pro různé ℓ_q -normy užitě jako penalizační funkce v situaci z příkladu 2.5. Souřadné osy odpovídají koeficientům β_1 a β_2 .

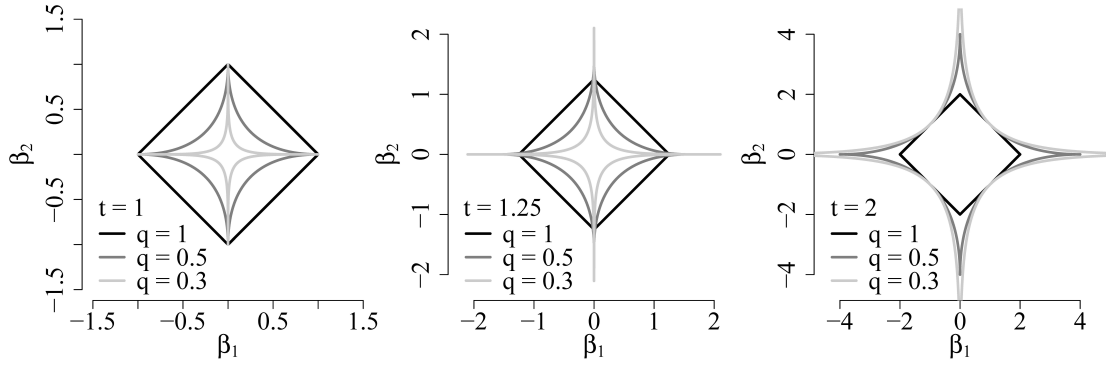
Poznámka 2.4. Penalizace pomocí ℓ_1 -normy žije – v kontextu diskutovaném v této sekci – vlastním životem v podoblasti zpracování signálu, která se nazývá *komprimované snímání*. Čtenáře odkazujeme například na Foucarta a Rauhuta (2013) pro pěkný úvod do problematiky. Ačkoliv by se z diskuze v předchozím odstavci mohlo zdát, že ℓ_1 -penalizace je tím, čím bychom se měli zabývat, mnohé (zejména experimentální, ale do jisté míry i teoretické) výsledky v oblasti komprimovaného snímání ukazují, že minimalizace nekonvexních ℓ_q -normou ($q < 1$) často vede k lepším výsledkům (Chartrand a Yin, 2008, Wipf a Nagarajan, 2010, Daubechies a kol., 2010). A to i přesto, že u nekonvexních problémů, na které užívání těchto norm vede, nejsme schopni garantovat nalezení skutečného globálního optima. \diamond

Příklad 2.5. Ilustrujme řešení úlohy (2.8) na dvourozměrném příkladu. Necht $\mathbf{X} \in \mathbb{R}^{n \times 2}$, $\text{rank}(\mathbf{X}) = 1$, $\mathbf{y} \in \mathbb{R}^2$ leží ve sloupcovém prostoru matice \mathbf{X} , a nulový prostor matice \mathbf{X} není rovnoběžný s žádnou ze souřadných os. Nepenalizovaná soustava (2.7) má v takovém případě nekonečně mnoho řešení, která leží na přímce v obecné poloze v \mathbb{R}^2 . Řešení penalizované úlohy (2.8) s $\rho(\cdot) := \|\cdot\|_q^q$, $q \in (0, 2]$ je vždy takový bod na této přímce, který se dotýká vrstevnice funkce $\|\beta\|_q^q$ co nejbližže počátku, viz obrázek 2.1. Zvláštním případem je penalizace $\rho(\cdot) := \|\cdot\|_0$, pro kterou žádné vrstevnice neexistují: jelikož $\text{rank}(\mathbf{X}) = 1$, mají všechna minima kardinalitu 1. Přitom všechny body v \mathbb{R}^2 s kardinalitou 1 jsou přesně souřadné osy. Řešeními jsou tedy průsečíky přímky se souřadnými osami. \triangle

Ačkoliv v situaci znázorněné na obrázku 2.1 dávají volby $q = 1$ a $q = 1/2$ stejné řešení $\hat{\beta}$, příslušné hodnoty účelových funkcí $\|\hat{\beta}\|_1$ a $\|\hat{\beta}\|_{1/2}^{1/2}$ nemusejí být nutně stejné. Záleží to na konkrétní pozici přímky, která určuje množinu řešení soustavy (2.7). Na obrázku 2.2 je pro srovnání zakresleno několik množin tvaru $\{\|\hat{\beta}\|_q^q = t\}$ pro různá q a t . Všimněme si, jak se tyto množiny pro pevné q liší v závislosti na t . Je důležité si uvědomit, že situace není tak jednoduchá, jak by se mohlo na první pohled z grafických výstupů se záměrnou absencí měřítka (jako náš obr. 2.1 nebo obr. 3.12 v Friedman a kol., 2005) zdát.

Zdůrazněme, že ačkoliv mluvíme o penalizaci pomocí ℓ_q -normou, myslíme tím penalizaci jejich q -tých mocnin. Tato penalizace pro malá q je motivována (viz např. Chartrand a Yin, 2008) limitním chováním

$$\lim_{q \rightarrow 0_+} \|\beta\|_q^q = \|\beta\|_0. \quad (2.9)$$



Obrázek 2.2: Hranice množin $\{\|\beta\|_q^q \leq t\}$ pro hodnoty $t = 1, \frac{5}{4}, 2$ a $q = 1, \frac{1}{2}, \frac{3}{10}$.

Ačkoliv rovnost (2.9) platí pro libovolné pevné $\beta \in \mathbb{R}^p$, tato rovnost neimplikuje konvergenci

$$\lim_{q \rightarrow 0_+} \{\beta \in \mathbb{R}^p; \|\beta\|_q^q \leq t\} = \{\beta \in \mathbb{R}^p; \|\beta\|_0 \leq t\} \quad (2.10)$$

v množinovém smyslu pro libovolné $t > 0$. Tato konvergence nemůže platit pro $t \leq 1$ ani $t > p$ (viz obrázek 2.2 pro ilustraci případu $p = 2$). Z toho plyne, že ačkoliv v úloze (2.8) je bezpečné aproximovat ℓ_0 -normu ℓ_q -normou pro malé q , v uvolněné úloze tvaru

$$\hat{\beta} = \arg \min_{\beta \in \mathbb{R}^p} \|\mathbf{y} - \mathbf{X}\beta\|_2^2 \quad \text{s.t.} \quad \rho(\beta) \leq t, \quad (2.11)$$

kde $t > 0$ je pevný parametr, obecně neplatí, že by pro $q \rightarrow 0_+$ globální minima úlohy (2.11) s $\rho(\cdot) := \|\cdot\|_q^q$ konvergovala v nějakém smyslu ke globálním minimům úlohy (2.11) s $\rho(\cdot) := \|\cdot\|_0$. Úloze ve tvaru (2.11) se budeme detailně věnovat později.

Model s chybovým členem

Nyní se vraťme zpět k modelu (2.1) s náhodným chybovým členem ϵ , tj. již nebudeme nepředpokládat, že by naše měření byla přesná. Z optimalizačního hlediska to znamená uvolnit vazebnou podmínku v (2.8) a řešit neomezenou úlohu

$$\hat{\beta} = \arg \min_{\beta \in \mathbb{R}^p} \|\mathbf{y} - \mathbf{X}\beta\|_2^2 + \lambda \rho(\beta), \quad (2.12)$$

kde λ je dopředu zvolený ladící parametr vyvažující poměr mezi očekávanou „velikostí“ chyb ϵ (mírou šumu) a „velikostí“ parametru β .

Úloha (2.12) pro $\rho(\cdot) := \|\cdot\|_0$ je v jistém smyslu (který bude vysvětlen v poznámce 2.6) ekvivalentní úloze (2.11), která se nazývá *úloha výběru nejlepší podmnožiny* (*best subset selection problem*). Jedná se o velmi důležitou úlohu statistického modelování, která je NP-těžká a její řešení vyžaduje kombinatorické hledání přes všechny možné opory vektoru parametru β , kterých je 2^p . Na druhou stranu, úloha (2.12) je stále konvexní a tudíž dobře řešitelná pro $q \geq 1$. Dalšími velmi důležitými případy tedy opět jsou $q = 2$, tzv. hřebenová regrese, a také $q = 1$, tzv. *lasso* (Tibshirani, 1996).

2.1 Hřebenová regrese

Řešení úlohy (2.12) s volbou penalizační funkce $\rho(\cdot) := \|\cdot\|_2^2$ vede na odhad metodou hřebenové regrese

$$\hat{\beta}_R = \arg \min_{\beta \in \mathbb{R}^p} \|\mathbf{y} - \mathbf{X}\beta\|_2^2 + \lambda \|\beta\|_2^2, \quad (2.13)$$

kde $\lambda \geq 0$ je předem daný ladící parametr. Odhad metodou hřebenové regrese lze také zavést jako řešení omezené optimalizační úlohy ve tvaru (2.11) jako

$$\hat{\beta}_R = \arg \min_{\beta \in \mathbb{R}^p} \|\mathbf{y} - \mathbf{X}\beta\|_2^2 \quad \text{s.t.} \quad \|\beta\|_2^2 \leq t, \quad (2.14)$$

kde t je předem daný parametr. Viz obrázek 2.3.

Poznámka 2.6. Mnozí autoři (např. Friedman a kol., 2005) nazývají úlohu (2.14) omezenou formulací problému a úlohu (2.13) *lagrangeovskou* formulací. Také uvádí, že mezi λ z (2.13) a t z (2.14) je vzájemně jednoznačný vztah a úlohy jsou tudíž ekvivalentní. To je pravda, dokud je omezení v (2.14) těsné, tj. dokud pro odhad platí $\|\hat{\beta}_R\|_2^2 = t$. Potom (2.13) je lagrangeovskou verzí úlohy (2.14) s tím rozdílem, že namísto hledání λ pro dané t (což je prováděno v metodě lagrangeových multiplikátorů) zadáváme přímo λ , a t nás nezajímá. Pokud však není omezení v (2.14) těsné pro t_0 , není těsné ani pro žádné $t > t_0$ a všem takovým t odpovídá $\lambda = 0$. Z toho důvodu nazýváme (2.13) raději *neomezenou formulací* problému. Neomezená formulace je vhodnější k přímému řešení, proto pracujeme výhradně s touto verzí. Problém (2.14) uvádíme pouze pro vybudování geometrické intuice; všechny obrázky se vztahují k této omezené verzi úlohy. \diamond

Úlohy (2.13) a (2.14) jsou striktně konvexní a snadno řešitelné. Pro srovnání, nepenalizovaná minimalizace čtverců (2.4) je konvexní, nemusí však být striktně konvexní. Řešení úlohy (2.13) získáme derivováním účelové funkce a následným řešením výsledné soustavy normálních rovnic

$$(\mathbf{X}^T \mathbf{X} + \lambda \mathbf{I}_p) \beta = \mathbf{X}^T \mathbf{y}, \quad (2.15)$$

Nyní bez ohledu na to, zda je $\mathbf{X}^T \mathbf{X}$ invertibilní, lze psát jednoznačné řešení jako

$$\hat{\beta}_R = (\mathbf{X}^T \mathbf{X} + \lambda \mathbf{I}_p)^{-1} \mathbf{X}^T \mathbf{y}. \quad (2.16)$$

Získaná jednoznačnost byla přesně důvodem prvního použití ℓ_2 -penalizace ve statistice (Hoerl a Kennard, 1970). Dnes je dobře známo (např. Friedman a kol., 2005), že metoda hřebenové regrese „tlačí“ regresní koeficienty směrem k nule. Míra tohoto zmenšování koeficientů závisí na ladícím parametru λ ; větší λ zřejmě způsobuje intenzivnější zmenšení. Použití ℓ_2 -penalizace tedy (jako použití každé jiné penalizace) vnáší do procedury odhadu jistou tendenčnost; odhady metodou hřebenové regrese nejsou nestranné ani za platnosti modelu 2.1 s předpokladem normality. Zároveň však ℓ_2 -penalizace snižuje celkový rozptyl a tím může zlepšit predikční schopnost modelu při vhodně zvoleném λ (Hoerl a Kennard, 1970, věta 4.3). Populárně lze říci, že hřebenová regrese potlačuje overfitting. Odhad $\hat{\beta}_R$ lze brát jako stabilizovanou verzi $\hat{\beta}_{OLS}$; žádáme čtenáře o porovnání (2.16) a (2.4).

Mezi hřebenovou regresí a obyčejným nepenalizovaným odhadem je mnoho dalších podobností. Např. penalizovaný reziduální součet čtverců má stejný tvar jako v (2.6), pouze projekční matice je odlišná.

Lemma 2.7. Penalizovaný reziduální součet čtverců pro odhad $\hat{\boldsymbol{\beta}}_R$ je tvaru

$$\|\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}_R\|_2^2 + \lambda\|\hat{\boldsymbol{\beta}}_R\|_2^2 = \mathbf{y}^\top(\mathbf{I}_n - \mathbf{H}_\lambda)\mathbf{y}, \quad (2.17)$$

kde $\mathbf{H}_\lambda = \mathbf{X}(\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I}_p)^{-1}\mathbf{X}^\top$.

Důkaz. Přeuspořádáním normálních rovnic (2.15) dostáváme

$$\lambda\hat{\boldsymbol{\beta}}_R = \mathbf{X}^\top(\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}_R). \quad (2.18)$$

Přenásobením $\hat{\boldsymbol{\beta}}_R^\top$ zleva dostáváme

$$\lambda\|\hat{\boldsymbol{\beta}}_R\|_2^2 = \hat{\boldsymbol{\beta}}_R^\top\mathbf{X}^\top(\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}_R) = (\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}_R)^\top\hat{\boldsymbol{\beta}}_R\mathbf{X}. \quad (2.19)$$

Jelikož triviálně platí

$$\lambda\|\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}_R\|_2^2 = (\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}_R)^\top\mathbf{y} - (\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}_R)^\top\mathbf{X}\hat{\boldsymbol{\beta}}_R = (\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}_R)^\top\mathbf{y} + \lambda\|\hat{\boldsymbol{\beta}}_R\|_2^2, \quad (2.20)$$

kde v druhé rovnosti jsme použili (2.18), dostáváme sečtením (2.19) a (2.20), že

$$\|\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}_R\|_2^2 + \lambda\|\hat{\boldsymbol{\beta}}_R\|_2^2 = (\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}_R)^\top\mathbf{y}. \quad (2.21)$$

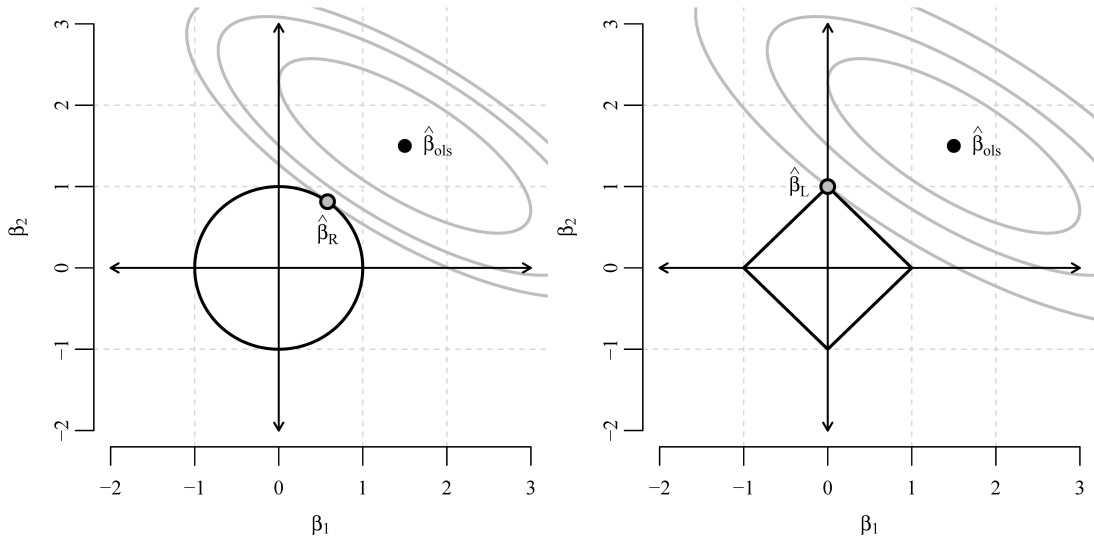
Konečně, přenásobením (2.16) maticí \mathbf{X} zleva dostáváme $\mathbf{X}\hat{\boldsymbol{\beta}}_R = \mathbf{H}_\lambda\mathbf{y}$. Dosažením tohoto vztahu do (2.21) dostáváme přesně (2.17). \square

Na závěr této sekce ještě poznamenejme, že pro ortogonální matici \mathbf{X} (tedy jsou-li regresory nekorelované) lze složky $\hat{\boldsymbol{\beta}}_R$ vyjádřit pomocí jednotlivých složek MNČ odhadu $\hat{\boldsymbol{\beta}}_{OLS}$, posunutých směrem k nule. Čtenáře odkazujeme znovu na Friedmana a kol. (2005, sekce 3.4.1).

Poznámka 2.8. Úlohu hřebenové regrese lze efektivně řešit pomocí singulárního rozkladu matice dat. Necht $\mathbf{X} = \mathbf{U}\mathbf{D}\mathbf{V}^\top$ je singulární rozklad \mathbf{X} hodnosti r , pak lze dosažením tohoto rozkladu do rovnosti (2.16) psát

$$\begin{aligned} \hat{\boldsymbol{\beta}}_R &= (\mathbf{V}\mathbf{D}^2\mathbf{V}^\top + \lambda\mathbf{I}_p)^{-1}\mathbf{V}\mathbf{D}\mathbf{U}^\top\mathbf{y} \\ &= \mathbf{V} \operatorname{diag}\left(\frac{1}{d_i^2 + \lambda}\right)_{i=1}^r \mathbf{V}^\top\mathbf{V}\mathbf{D}\mathbf{U}^\top\mathbf{y} \\ &= \mathbf{V} \operatorname{diag}\left(\frac{d_i}{d_i^2 + \lambda}\right)_{i=1}^r \mathbf{U}^\top\mathbf{y}. \end{aligned}$$

Namísto počítání inverze matice $(\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I}_p)$, což by vyžadovalo $O(p^3)$ operací, si tedy lze vystačit se singulárním rozkladem ($O(np^2)$ operací) a násobením matic menších rozměrů (méně než $O(np^2)$ operací). \diamond



Obrázek 2.3: Vlevo: Odhad metodou hřebenové regrese získaný řešením úlohy (2.14) s $t = 1$. Vpravo: Lasso odhad získaný řešením úlohy (2.23) s $t = 1$. Řešením je v obou případech bod ležící v přípustné oblasti dotýkající se vrstevnice reziduálního součtu čtverců ležící co nejbližší k MNČ odhadu.

2.2 Lasso

Praktický dopad hřebenové regrese na zlepšení predikční schopnosti modelu je značný. Ačkoliv ℓ_2 -penalizace vynucuje menší koeficienty než obyčejná MNČ, koeficienty bývají typicky nenulové, stejně jako v případě MNČ. Neméně důležité však je „povzbudit“ proceduru hledání odhadu tak, aby některé koeficienty byly odhadnuty jako nulové. Tím dojde ke spojení selekce důležitých proměnných se samotnou procedurou jejich odhadu.

Tohoto cíle lze dosáhnout například kontrolováním ℓ_1 -normy, jejíž použití v penalizačním členu vede na tzv. lasso odhad

$$\hat{\beta}_L = \arg \min_{\beta \in \mathbb{R}^p} \|\mathbf{y} - \mathbf{X}\beta\|_2^2 + \lambda \|\beta\|_1, \quad (2.22)$$

kde $\lambda \geq 0$ je opět předem daný ladící parametr. Neomezenou verzi úlohy lze znovu psát ve tvaru

$$\hat{\beta}_L = \arg \min_{\beta \in \mathbb{R}^p} \|\mathbf{y} - \mathbf{X}\beta\|_2^2 \quad \text{s.t.} \quad \|\beta\|_1 \leq t, \quad (2.23)$$

pro $t \geq 0$ je pevné.

Na obrázku 2.3 je ilustrován rozdíl mezi lasso odhadem a odhadem metodou hřebenové regrese na umělém příkladě s dvěma korelovanými regresory. Zatímco hřebenová regrese odhaduje oba koeficienty jako nenulové, lasso vybírá pouze jeden koeficient a druhý odhaduje nulou.

V případě nekorelovaných regresorů lze opět vyjádřit jednotlivé složky $\hat{\beta}_L$ pomocí příslušných složek $\hat{\beta}_{OLS}$, jako v případě hřebenové regrese. Pokud však \mathbf{X} není ortogonální, nelze $\hat{\beta}_L$ analyticky vyjádřit. Přesto lze lasso úlohu řešit poměrně genericky metodami konvexního programování (například jí lze přepsat jako úlohu kvadratického programování v $2p$ proměnných). Lasso odhad je však

natolik důležitý, že ve snaze o maximální zefektivnění jeho získávání bylo navrženo mnoho specializovaných algoritmů. Jeden z nich si představíme v sekci 2.5.

Poznámka 2.9. Lasso úloha (2.22) bývá často považována za konvexní uvolnění NP-těžkého nekonvexního problému výběru nejlepší podmnožiny, tj. problému (2.12) (nebo (2.11)) s volbou $\rho(\cdot) := \|\cdot\|_0$, který bychom ve skutečnosti chtěli řešit. Nicméně, jelikož ℓ_0 -norma nepenalizuje velikost koeficientů, „okolí“ bodu 0 jsou neomezená. Např. množina $\{\|\beta\|_0 \leq 1\} \subset \mathbb{R}^p$ je tvořena všemi souřadnými osami v daném prostoru (viz obr. 2.1 vpravo pro ilustraci v \mathbb{R}^2). Konvexní obal této množiny je vždy celý prostor \mathbb{R}^p . Označení „konvexní uvolnění“ tedy není zcela přesné v běžném optimalizačním smyslu pro nahrazení ℓ_0 -normy ℓ_1 -normou.

◇

Každá statistická procedura je posuzována zejména z pohledu následujících třech cílů: přesnost (predikce), interpretovatelnost (jednoduchost), a výpočetní náročnost. Lasso může potenciálně zlepšit predikční schopnost modelu oproti obyčejné MNČ podobně jako hřebenová regrese. Navíc zjednodušuje model díky tomu, že některé koeficienty jsou odhadnuty přímo nulami. A konečně, jako konvexní úloha je atraktivní z výpočetního hlediska.

2.3 Adaptivní lasso

Fan a Li (2001) považují za důležité, aby každá procedura provádějící současně výběr relevantních proměnných a odhad koeficientů měla tzv. věšternou vlastnost (*oracle property*). Procedura má vlastnost věšternou vlastnost v klasické asymptotické situaci, kdy počet parametrů $p \in \mathbb{N}$ je fixní a počet pozorování jde k nekonečnu $n \rightarrow \infty$, pokud je tato procedura:

- konzistentní, tj. $\widehat{\beta}(n) \rightarrow \beta^*$ v pravděpodobnosti pro $n \rightarrow \infty$;
- stejně efektivní, jako by byla MNČ s dodatečnou informací o skutečné opoře β^* (efektivní ve smyslu rychlosti konvergence k β^*).

Fan a Li (2001) se domnívají, že lasso nemá věšternou vlastnost, a navrhuje svůj nekonvexní SCAD odhad, který tuto vlastnost má.

Problém s metodou lasso je následující. Pokud je ladící parametr λ velký, lasso příliš silně přitahuje ve skutečnosti nenulové koeficienty směrem k nule. Pokud je naopak λ malé, lasso odhaduje příliš velký počet koeficientů jako nenulové a není konzistentní ani v odhadu opory. Meinshausen a Bühlmann (2006) ukázali, že druhý uvedený případ nastává v situaci, kdy je λ zvoleno s ohledem na optimální predikční schopnosti modelu (např. pomocí *křížového ověřování*, Friedman a kol., 2005). Meinshausen a Bühlmann (2006) také ukázali, že skutečně existují příklady regresní matice \mathbf{X} , ve kterých lasso nemůže mít vlastnost orákula pro žádnou volbu λ .

Na základě diskuze v předchozím odstavci se nabízí přirozené řešení: adaptivní ℓ_1 -penalizace, která by ve skutečnosti malé koeficienty tlačila k nule vyšší penalizací, a naopak velké koeficienty by penalizovala jen mírně. Zou (2006) navrhl

adaptivní lasso tímto způsobem, tj.

$$\hat{\boldsymbol{\beta}}_{AL} = \arg \min_{\boldsymbol{\beta} \in \mathbb{R}^p} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \lambda \sum_{j=1}^p w_j |\beta_j|, \quad (2.24)$$

kde $w_j \geq 0$, $j = 1, \dots, p$.

Váhy w_j se typicky odhadují z dat. Nejčastějšími volbami jsou

$$w_j = \frac{1}{\hat{\beta}_j^{(0)}} \quad \text{nebo} \quad w_j = \frac{1}{\hat{\beta}_j^{(0)} + \delta}, \quad (2.25)$$

kde $\delta > 0$ je malý stabilizační parametr a $\hat{\boldsymbol{\beta}}^{(0)}$ je počáteční odhad parametru $\boldsymbol{\beta}^*$, například odhad pomocí MNČ nebo lasso odhad s malým λ .

Poznámka 2.10. Úlohu (2.24) lze řešit jako obyčejnou lasso úlohu po jednoduché transformaci dat. Označme

$$\begin{aligned} b_i &= w_i \beta_i, & i &= 1, \dots, p, \\ \tilde{\mathbf{x}}_i &= \frac{1}{w_i} \mathbf{x}_i, & i &= 1, \dots, p. \end{aligned}$$

Je-li $\hat{\mathbf{b}}$ řešením obyčejné lasso úlohy

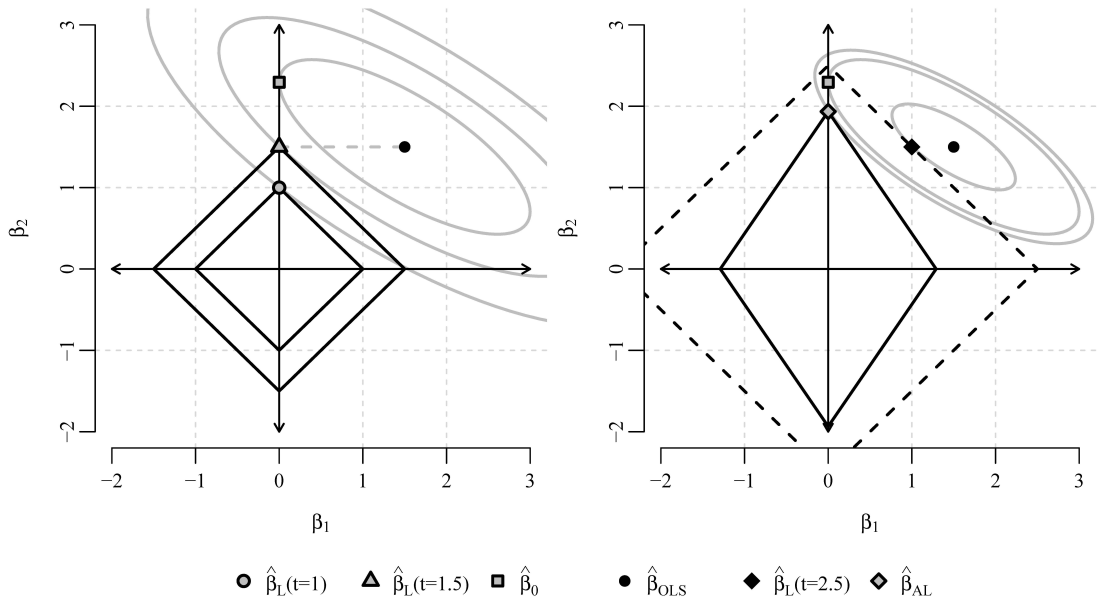
$$\arg \min_{\mathbf{b} \in \mathbb{R}^p} \|\mathbf{y} - \widetilde{\mathbf{X}}\mathbf{b}\|_2^2 + \lambda \|\mathbf{b}\|_1,$$

pak řešení úlohy (2.24) je dáno jako $\hat{\beta}_i = \frac{1}{w_i} \hat{b}_i$. \diamond

Zou (2006) ukázal, že za mírných podmínek regularity má adaptivní lasso vlastnost orákula. Namísto hledání nekonvexního SCAD odhadu, adaptivní lasso má požadované vlastnosti, ačkoliv se jedná pouze o dvoukrokový odhad, přičemž úloha řešená v obou krocích je konvexní.

Zdá se přirozené zobecnit tento dvoukrokový postup na vícekrokový. To bylo navrženo v Zou a Li (2008) a také v Candés a kol. (2008). Zatímco ve statistické komunitě není vícekrokový odhad příliš populární (dle našeho nejlepšího vědomí), v komunitě zabývající se zpracováním signálu je postup navržený Candésem a kol. často používaný.

Příklad 2.11. Na obrázku 2.4 je ilustrační příklad adaptivní lasso metody v situaci se dvěma korelovanými regresory. Stejné nastavení bylo použito i na obrázek 2.3, žádáme čtenáře o porovnání. Předpokládejme nyní, že chceme nalézt optimální (ve smyslu nejmenších čtverců) odhad $\boldsymbol{\beta}$ s kardinalitou 1. Tj. chceme řešit NP-těžkou úlohu výběru nejlepší podmnožiny velikosti 1 – úlohu (2.11) s volbou $\rho(\cdot) := \|\cdot\|_0$. Řešením této úlohy je $\hat{\boldsymbol{\beta}}_0$ z obr. 2.4. Při snaze nalézt $\hat{\boldsymbol{\beta}}_0$ řešením jednodušší lasso úlohy je bod $\hat{\boldsymbol{\beta}}_L(t = \frac{3}{2})$ získaný optimální volbou $t = \frac{3}{2}$ tím nejbližším bodem k $\hat{\boldsymbol{\beta}}_0$, jaký může lasso metoda poskytnout. Všimněme si, že dalším zvyšováním t přestane mít lasso odhad kardinalitu 1; dalším zvyšováním t se dokonce lasso odhad vzdaluje od $\hat{\boldsymbol{\beta}}_0$ a blíží se MNČ odhadu $\hat{\boldsymbol{\beta}}_{OLS}$. Oproti tomu adaptivní lasso odhad, který v prvním kroku nalezne jen dostatečně mírně penalizovaný odhad $\hat{\boldsymbol{\beta}}_L(t = \frac{5}{2})$ a v druhém kroku vyřeší úlohu adaptivního lasso metody (2.24) s $t \approx 1.3$ vahami $w_j = 1 / \left(\hat{\boldsymbol{\beta}}_L(t = \frac{5}{2}) \right)_j$, dokáže žádaný odhad $\hat{\boldsymbol{\beta}}_0$ aproximovat mnohem lépe. \triangle



Obrázek 2.4: Ilustrace k příkladu 2.11, situace je obdobná jako na obr. 2.3. *Vlevo:* řešení lasso úlohy pro volbu $t = 1$ a optimální volbu $t = 1.5$. *Vpravo:* adaptivní lasso s počátečním lasso odhadem s $t = \frac{5}{2}$.

2.4 Elastická síť

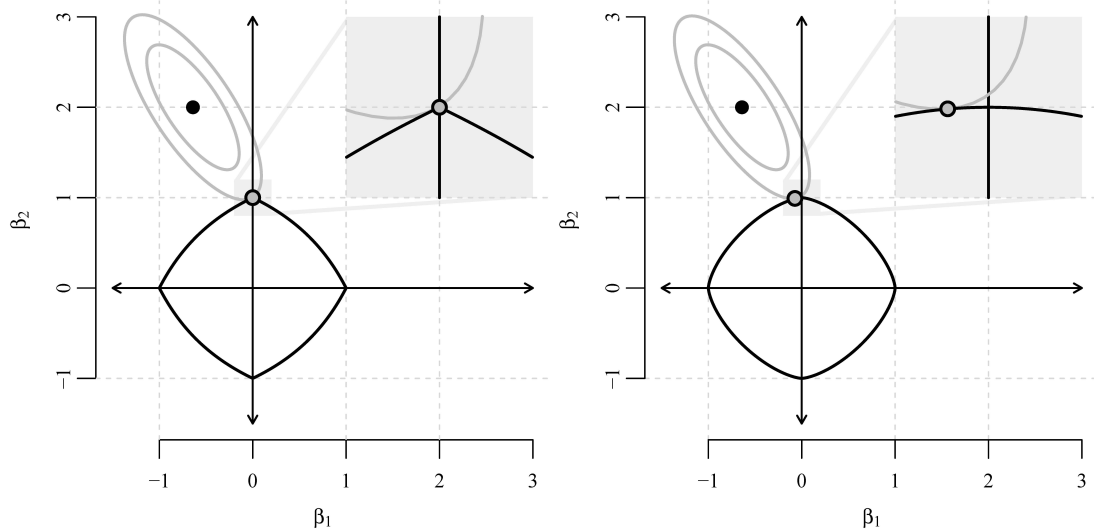
Zou a Hastie (2005) navrhli zkombinovat lasso s hřebenovou regresí a výsledný odhad, který penalizuje paralelně ℓ_1 - i ℓ_2 -normu koeficientů, nazvali *elastická síť*

$$\hat{\beta}_{EN} = \arg \min_{\beta \in \mathbb{R}^p} \|\mathbf{y} - \mathbf{X}\beta\|_2^2 + \lambda_1 \|\beta\|_1 + \lambda_2 \|\beta\|_2^2. \quad (2.26)$$

Zou a Hastie (2005) uvedli mnoho výhod, kterými podle nich disponuje elastická síť vůči lasso metodě; některé z těchto výhod jsou do jisté míry diskutabilní. My zde pouze uvedeme, že na elastickou síť se lze dívat jako na stabilizovanou verzi lasso metody ve stejném smyslu, jako je hřebenová regrese stabilizovanou verzí klasického MNČ odhadu. Úloha (2.26) je díky ℓ_2 -penalizaci striktně konvexní. Řešení je tedy jednoznačné, ale opět analyticky nevyjádřitelné. Většinu algoritmů používaných na hledání lasso odhadů lze použít jen s drobnými modifikacemi i k řešení úlohy (2.26).

Poznámka 2.12. Samozřejmě lze kombinovat ℓ_1 -penalizaci i s jinou ℓ_q -penalizací. Obecně, přidání ℓ_q -penalizace, $q > 1$, k lasso penalizaci zajistí striktně konvexní geometrii. Např. Liu a Ye (2010) se zabývají případem $q \in (1,2)$. Avšak výpočetní náročnost oproti ℓ_2 -penalizaci je vyšší, tudíž jsou třeba specializovaných algoritmů. Patrně z tohoto důvodu (v kombinaci s malou přidanou hodnotou takových postupů) si nejsme vědomi, že by se tyto obecnější součtové penalizace běžně užívaly. \diamond

Podobně jako lasso, i elastická síť produkuje řídké odhady parametrů, tj. odhady některých koeficientů mají tendenci být nulové. Důvody jsou stejné jako v případě lasso metody, opět zde uvádíme pouze geometrickou intuici na obrázku (2.5), na němž zároveň ilustrujeme, že penalizace součtu ℓ_1 - a ℓ_2 -normy je odlišná od penalizace ℓ_q -normy pro $q \in (1,2)$. Důvodem je, že součet ℓ_1 - a ℓ_2 -normy



Obrázek 2.5: Řešení penalizované úlohy nejmenších čtverců (2.11). *Vlevo:* ℓ_3 -penalizace. *Vpravo:* Penalizace elastické sítě $\rho(\cdot) := a \|\beta\|_1 + (1 - a)\|\beta\|_2^2$ pro $a = \frac{1}{2}$.

dědí singularity ℓ_1 -normy v $\beta_j = 0$, zatímco ℓ_q -norma pro $q > 1$ tyto singularity postrádá.

Obdobně jako adaptivní lasso bylo navrženo jako rozšíření klasické lasso metody, Zou a Zhangová (2009) navrhli adaptivní elastickou síť s převažovanou ℓ_1 -normou a ukázali, že má věšteinou vlastnost. Vícekrokovou adaptivní elastickou síť lze opět definovat prostým přidáním ℓ_2 -penalizace k vícekové adaptivní lasso metodě.

2.5 LARS algoritmus pro lasso

V předchozích sekcích jsme si připomněli lasso a některé jeho modifikace jako adaptivní lasso a elastickou síť. Jak již bylo diskutováno, řešení těchto modifikací se liší od řešení obyčejné lasso úlohy pouze nepatrně. Proto nyní představme algoritmus pro hledání lasso odhadu: LARS algoritmus navržený Efronem a kol. (2004).

Ještě před samotným popisem algoritmu znovu zdůrazněme, že existuje mnoho různých algoritmů pro hledání lasso odhadů. LARS přitom není považován za nikterak efektivní, ale šikovně využívá napůl kvadratickou a napůl lineární geometrickou strukturu lasso úlohy. Nenulové koeficienty jsou přidávány postupně v krocích od nejdůležitějšího k méně důležitým (ve smyslu korelovanosti s odezvou), a LARS tak vypočítává celou dráhu řešení lasso metody². Tím zprůhledňuje vztah mezi λ z neomezené verze problému a t z omezené verze.

Jelikož algebraické detaily LARS algoritmu vypadají mnohem složitěji než jeho jednoduchá geometrická motivace, a protože důkaz, že LARS skutečně řeší lasso problém, je poměrně komplikovaný, uvádíme zde pouze jeho neformální popis. LARS algoritmus postupuje následovně.

²Celou dráhou řešení rozumíme řešení pro všechny hodnoty parametru λ .

1. Polož odhad parametrů $\hat{\beta} = 0$ a odhad reziduí $\hat{\epsilon} = \mathbf{y}$. Najdi regresor \mathbf{x}_j nejvíce korelovaný s $\hat{\epsilon}$.
2. Začni odhad $\hat{\beta}_j$ posouvat po dráze směrem k jeho MNČ hodnotě v jednoduchém modelu $\hat{\epsilon} \approx \beta_j \mathbf{x}_j$ a během toho aktualizuj rezidua jako $\hat{\epsilon} = \mathbf{y} - \hat{\beta}_j \mathbf{x}_j$ (což snižuje korelaci mezi \mathbf{x}_j a $\hat{\epsilon}$), dokud není nějaký regresor \mathbf{x}_k korelovaný s $\hat{\epsilon}$ stejně jako \mathbf{x}_j .
3. Změň směr. Nyní posouvej koeficienty $\hat{\beta}_j$ a $\hat{\beta}_k$ směrem k MNČ odhadu koeficientů v modelu $\hat{\epsilon} \approx \beta_j \mathbf{x}_j + \beta_k \mathbf{x}_k$ a znovu aktualizuj rezidua, dokud se do hry nedostane další regresor.³
4. Pokračuj tímto způsobem dokud nejsou přidány všechny regresory nebo dokud není $\hat{\epsilon} = \mathbf{0}$ (nutně nastává v případě $p > n$).

Hlavním důvodem atraktivity LARS algoritmu je, že ve třetím kroku (a všech následujících krocích) lze snadno určit bod, ve kterém má být změněn směr. Algoritmus tedy postupuje v diskrétních krocích a v každém kroku je přidán právě jeden prediktor.

Předchozí popis algoritmu neodpovídá přímo lasso metodě nýbrž *forward selection*. Příslušnou modifikací pro lasso je následující krok:

- kdykoliv nějaký nenulový koeficient $\hat{\beta}_l$ narazí na nulu, odeber \mathbf{x}_l z modelu a spočítej nový směr v modelu bez tohoto regresoru.

Důvody, proč v této práci používáme právě LARS algoritmus jsou dva. Za prvé, díky postupnému přidávání (a ubírání) proměnných lze LARS algoritmu zadat místo ladícího parametru λ z (2.22) (případně parametru t z příslušné omezené verze úlohy) přímo požadovanou kardinalitu řešení s , tj. požadovaný počet nenulových koeficientů⁴. Za druhé, LARS přirozeně počítá celou dráhu řešení lasso problému, tj. řešení pro všechny možné kardinality $s = 1, \dots, p$. To může být žádoucí nejen ve spojení s PCA.

Poznámka 2.13. V celé dráze řešení spočtené LARS algoritmem je tedy implicitně zakódovaný vztah mezi parametry s , t a λ pro konkrétní úlohu. Pokud například zadáme LARS algoritmu jako vstupní parametr požadovanou kardinalitu s , můžeme klidně uvažovat, že jsme zadali parametr λ vedoucí k řešení příslušné kardinality a že LARS nalezne řešení penalizované úlohy (2.22) s takovým λ . \diamond

Hlavní nevýhodou LARS algoritmu je naopak jeho vysoká výpočetní náročnost, celá dráha řešení totiž není spočítána zadarmo. Efron a kol. (2004) sice tvrdí, že LARS dokáže spočítat celou dráhu řešení za stejnou cenu jako je cena výpočtu jednoho MNČ odhadu, ale toto tvrzení je pravdivé pouze z jistého úhlu pohledu:

³V příkladu 2.11 nastává tato situace pro omezenou verzi lasso úlohy s $t = \frac{3}{2}$. Na obr. 2.4 vlevo je znázorněna celá dráha LARS algoritmu sestávající z úseček $[(0,0)^T, (0, \frac{3}{2})^T]$ a $[(0, \frac{3}{2})^T, \hat{\beta}_{OLS}]$. LARS tedy mění směr jen jednou a končí 3. krokem.

⁴Pro srovnání, mnohé jiné algoritmy také nabízejí tuto možnost. Dosahují ji však poněkud méně přirozeně: neomezená lasso úloha (2.22) je řešena pro různé hodnoty λ , a následně je vybráno řešení s požadovanou kardinalitou.

je-li p zanedbatelné vůči n . V takovém případě však typicky lasso vůbec nepoužíváme a vystačíme si s obyčejným MNČ odhadem. Pokud p není zanedbatelné vůči n , může být LARS algoritmus velmi pomalý.

2.6 Analýza hlavních komponent pomocí penalizované regrese

V kapitole 1 jsme si ukázali několik různých zavedení HK pomocí různých optimalizačních úloh. Všechny tyto úlohy jsme nakonec řešili pomocí SVD rozkladu matice dat. Nicméně různé algoritmy k provedení singulárního rozkladu se značně liší, typicky podle toho, z jakého úhlu pohledu se díváme na úlohu, jejíž řešení je dáno singulárním rozkladem jisté matice. Obecně různé přeformulace úloh vedou na nové pohledy a nové algoritmy. V této sekci si ukážeme, že HK lze hledat také pomocí penalizované metody nejmenších čtverců. Část této sekce je přejata z Zou a kol. (2006), odkud pochází lemmata 2.15 a 2.17 a věta 2.16 společně s jejich důkazy.

Poznámka 2.14. Znění lemma 2.15 a věty 2.16 je zde oproti Zou a kol. (2006) pozměněné. Zou a kol. tvrdí, že „AHK dává vždy jednoznačné výsledky“, a obrnění tímto tvrzením jsou do jisté míry vágní v mnohých svých úvahách. Z našeho úhlu pohledu však např. náboje příslušné nulovým vlastním číslům (ta nutně existují v situaci $p > n$) nejsou jednoznačně určeny. To samé platí obecněji v případě dvou či více stejných nenulových vlastních čísel – náboje opět nejsou jednoznačně určeny. Vzhledem k tomu, že neklademe žádné předpoklady n na rozdělení, z něhož pochází naše data, ale naopak jen pracujeme s nějakou pozorovanou maticí, je nutné tyto případy uvažovat. \diamond

Elementární spojitost AHK a hřebenové regrese je dána následujícím lemma-tem.

Lemma 2.15. Necht $\mathbf{X} \in \mathbb{R}^{n \times p}$ je matice dat, $\mathbf{v}_1, \dots, \mathbf{v}_p$ jsou náboje dané spektrálním rozkladem $\mathbf{X}^\top \mathbf{X} = \mathbf{V} \mathbf{D}^2 \mathbf{V}^\top$ matice $\mathbf{X}^\top \mathbf{X}$ a $\mathbf{z}_1 = \mathbf{X} \mathbf{v}_1, \dots, \mathbf{z}_p = \mathbf{X} \mathbf{v}_p$ jsou HK. Necht pro $\lambda > 0$ a $k = 1, \dots, p$ je

$$\hat{\boldsymbol{\beta}}_k = \arg \min_{\boldsymbol{\beta} \in \mathbb{R}^p} \|\mathbf{z}_k - \mathbf{X} \boldsymbol{\beta}\|_2^2 + \lambda \|\boldsymbol{\beta}\|_2^2.$$

Potom $\frac{\hat{\boldsymbol{\beta}}_k}{\|\hat{\boldsymbol{\beta}}_k\|_2} = \mathbf{v}_k$ pro $k = 1, \dots, p$.

Důkaz. Z (2.16) víme, že

$$\hat{\boldsymbol{\beta}}_k = (\mathbf{X}^\top \mathbf{X} + \lambda \mathbf{I}_p)^{-1} \mathbf{X}^\top \mathbf{z}_k = (\mathbf{X}^\top \mathbf{X} + \lambda \mathbf{I}_p)^{-1} \mathbf{X}^\top \mathbf{X} \mathbf{v}_k.$$

Dosazením spektrálního rozkladu $\mathbf{X}^\top \mathbf{X}$ dostáváme

$$\hat{\boldsymbol{\beta}}_k = (\mathbf{V} \mathbf{D}^2 \mathbf{V}^\top + \lambda \mathbf{V} \mathbf{V}^\top)^{-1} \mathbf{V} \mathbf{D}^2 \mathbf{V}^\top \mathbf{v}_k = \mathbf{V} \operatorname{diag} \left(\frac{d_i^2}{d_i^2 + \lambda} \right)_{i=1}^p \mathbf{V}^\top \mathbf{v}_k = \frac{d_k^2}{d_k^2 + \lambda} \mathbf{v}_k.$$

Tvrzení dostáváme ihned po normalizaci. \square

Předchozí lemma říká, že při regresi HK na data je odhad koeficientů pomocí hřebenové regrese roven přesně nábojům. To není příliš překvapivé vzhledem k tomu, že HK jsou právě lineární kombinace dat určené náboji.

Zajímavá je role penalizace v lemmatu 2.15. Ta je přítomna proto, aby zajistila jednoznačnost nábojů. Náboje sice samy o sobě nemusejí být jednoznačně definovány, jak bylo diskutováno v poznámce 2.14, ale jsou-li dány HK, pak jednoznačné jsou. Všimněme si také, že rekonstrukce nábojů z HK v lemmatu 2.15 je zcela nezávislá na volbě parametru λ .

Předchozí lemma však není soběstačnou alternativou jak provádět AHK, neboť HK musejí být známy dopředu. Následující věta dává takovou soběstačnou alternativu.

Věta 2.16. Necht $\mathbf{X} \in \mathbb{R}^{n \times p}$ je matice dat, $\lambda > 0$, $K \in [p]$. Dále necht

$$\begin{aligned} (\widehat{\mathbf{A}}, \widehat{\mathbf{B}}) &= \arg \min_{\mathbf{A}, \mathbf{B} \in \mathbb{R}^{p \times K}} \sum_{i=1}^n \|\mathbf{x}^{(i)} - \mathbf{A}\mathbf{B}^\top \mathbf{x}^{(i)}\|_2^2 + \lambda \sum_{j=1}^K \|\mathbf{b}_j\|_2^2 \\ \text{s.t.} \quad &\mathbf{A}^\top \mathbf{A} = \mathbf{I}_K. \end{aligned} \quad (2.27)$$

Potom $\frac{\widehat{\mathbf{b}}_j}{\|\widehat{\mathbf{b}}_j\|_2} = \mathbf{v}_j$ pro $j = 1, \dots, K$, kde $\mathbf{v}_1, \dots, \mathbf{v}_K$ je prvních K vlastních vektorů výběrové kovarianční matice.

Důkaz. Pomocí ortonormálního doplňku matice (viz lemma B.2 v dodatcích) lze psát

$$\begin{aligned} \sum_{i=1}^n \|\mathbf{x}^{(i)} - \mathbf{A}\mathbf{B}^\top \mathbf{x}^{(i)}\|_2^2 &= \|\mathbf{X} - \mathbf{X}\mathbf{B}\mathbf{A}^\top\|_F^2 \\ &= \|(\mathbf{X} - \mathbf{X}\mathbf{B}\mathbf{A}^\top)(\mathbf{A}|\mathbf{A}_c)\|_F^2 \\ &= \|(\mathbf{X}\mathbf{A} - \mathbf{X}\mathbf{B}\mathbf{A}^\top \mathbf{A}|\mathbf{X}\mathbf{A}_c - \mathbf{X}\mathbf{B}\mathbf{A}^\top \mathbf{A}_c)\|_F^2 \\ &= \|(\mathbf{X}\mathbf{A} - \mathbf{X}\mathbf{B}|\mathbf{X}\mathbf{A}_c)\|_F^2 \\ &= \|\mathbf{X}\mathbf{A} - \mathbf{X}\mathbf{B}\|_F^2 + \|\mathbf{X}\mathbf{A}_c\|_F^2 \\ &= \sum_{j=1}^K \|\mathbf{X}\mathbf{a}_j - \mathbf{X}\mathbf{b}_j\|_F^2 + \|\mathbf{X}\mathbf{A}_c\|_F^2, \end{aligned}$$

kde v první, předposlední a poslední rovnosti jsme použili definici Frobeniovy normy a v druhé rovnosti jsme použili fakt, že Frobeniova norma je invariantní při ortonormální transformaci, viz poznámka B.7 v dodatcích.

Nyní tedy máme účelovou funkci (2.27) vyjádřenou ve tvaru

$$\sum_{i=1}^n \|\mathbf{x}^{(i)} - \mathbf{A}\mathbf{B}^\top \mathbf{x}^{(i)}\|_2^2 + \lambda \sum_{j=1}^K \|\mathbf{b}_j\|_2^2 = \sum_{j=1}^K \|\mathbf{X}\mathbf{a}_j - \mathbf{X}\mathbf{b}_j\|_F^2 + \lambda \sum_{j=1}^K \|\mathbf{b}_j\|_2^2 + \|\mathbf{X}\mathbf{A}_c\|_F^2. \quad (2.28)$$

To znamená, že pro fixní \mathbf{A} (a tím i fixní \mathbf{A}_c), řešení (2.27) je ekvivalentní řešení K nezávislých problémů hřebenové regrese

$$\min_{\mathbf{b}_j \in \mathbb{R}^p} \|\mathbf{X}\mathbf{a}_j - \mathbf{X}\mathbf{b}_j\|_2^2 + \lambda \|\mathbf{b}_j\|_2^2, \quad j = 1, \dots, K. \quad (2.29)$$

Tudíž pro $j = 1, \dots, K$ je

$$\widehat{\mathbf{b}}_j = (\mathbf{X}^\top \mathbf{X} + \lambda \mathbf{I})^{-1} \mathbf{X}^\top \mathbf{X} \mathbf{a}_j . \quad (2.30)$$

Dosazením těchto řešení zpět do tvaru účelové funkce z (2.28) dostáváme částečně optimalizovaný tvar účelové funkce

$$\begin{aligned} \sum_{i=1}^n \|\mathbf{x}^{(i)} - \mathbf{A} \widehat{\mathbf{B}}^\top \mathbf{x}^{(i)}\|_2^2 + \lambda \sum_{j=1}^K \|\widehat{\mathbf{b}}_j\|_2^2 &= \|\mathbf{X} \mathbf{A} \mathbf{c}\|_F^2 + \sum_{j=1}^K (\mathbf{X} \mathbf{a}_j)^\top (\mathbf{I} - \mathbf{H}_\lambda) (\mathbf{X} \mathbf{a}_j) \\ &= \|\mathbf{X} \mathbf{A} \mathbf{c}\|_F^2 + \text{tr} \left((\mathbf{X} \mathbf{A})^\top (\mathbf{I} - \mathbf{H}_\lambda) (\mathbf{X} \mathbf{A}) \right) , \end{aligned}$$

kde jsme v první rovnosti použili lemma 2.7 pro každé $j = 1, \dots, K$ zvlášť a v druhé rovnosti jsme využili definici operátoru stopy matice spolu s faktem, že \mathbf{A} je ortonormální. Matice \mathbf{H}_λ je hřebenová *hat matrix* z lemmatu 2.7.

Pokračováním v úpravách posledního výrazu dostáváme

$$\begin{aligned} \underbrace{\text{tr}(\mathbf{A} \mathbf{c}^\top \mathbf{X}^\top \mathbf{X} \mathbf{A} \mathbf{c}) + \text{tr}(\mathbf{A}^\top \mathbf{X}^\top \mathbf{X} \mathbf{A})}_{= \|\mathbf{X} \mathbf{A} \mathbf{c}\|_F^2 + \|\mathbf{X} \mathbf{A}\|_F^2 = \|\mathbf{X}(\mathbf{A} \mathbf{A} \mathbf{c})\|_F^2 = \|\mathbf{X}\|_F^2} - \text{tr}(\mathbf{A}^\top \mathbf{X}^\top \mathbf{H}_\lambda \mathbf{X} \mathbf{A}) &= \|\mathbf{X}\|_F^2 - \text{tr}(\mathbf{A}^\top \mathbf{X}^\top \mathbf{H}_\lambda \mathbf{X} \mathbf{A}) . \end{aligned}$$

Tudíž pro optimalizaci podle proměnné \mathbf{A} zbývá vyřešit úlohu

$$\arg \max_{\mathbf{A}} \text{tr}(\mathbf{A}^\top \mathbf{X}^\top \mathbf{H}_\lambda \mathbf{X} \mathbf{A}) \quad \text{s.t.} \quad \mathbf{A}^\top \mathbf{A} = \mathbf{I} .$$

Uvažováním spektrálního rozkladu $\mathbf{X}^\top \mathbf{X} = \mathbf{V} \mathbf{D}^2 \mathbf{V}^\top$ dostáváme

$$\begin{aligned} \mathbf{X}^\top \mathbf{H}_\lambda \mathbf{X} &= \mathbf{X}^\top \mathbf{X} (\mathbf{X}^\top \mathbf{X} + \lambda \mathbf{I})^{-1} \mathbf{X}^\top \mathbf{X} = \mathbf{V} \mathbf{D}^2 (\mathbf{D}^2 + \lambda \mathbf{I})^{-1} \mathbf{D}^2 \mathbf{V}^\top \\ &= \mathbf{V} \text{diag} \left(\frac{d_i^4}{d_i^2 + \lambda} \right)_{i=1}^p \mathbf{V}^\top , \end{aligned}$$

tedy musí být $\widehat{\mathbf{A}} = (\mathbf{v}_1, \dots, \mathbf{v}_K)$ na základě stejného argumentu jako v důkazu věty 1.2. Dosazením matice $\widehat{\mathbf{A}}$ spolu se spektrálním rozkladem $\mathbf{X}^\top \mathbf{X}$ zpět do (2.30) dostáváme

$$\widehat{\mathbf{b}}_j = \mathbf{V} \text{diag} \left(\frac{d_i^4}{d_i^2 + \lambda} \right)_{i=1}^p \mathbf{V}^\top \mathbf{v}_j = \frac{d_j^4}{d_j^2 + \lambda} \mathbf{v}_j . \quad (2.31)$$

pro $j = 1, \dots, K$.

Tvrzení věty dostáváme opět po normalizaci. □

V předchozí větě je úloha hledání prvních K nábojů přeformulována v nový optimalizační problém, který je opět nekonvexní. Důkaz věty přitom naznačuje přirozený algoritmus k řešení tohoto problému: inicializuj s ortonormální maticí $\mathbf{A}^{(0)}$ (např. jednotkovou) a iteruj pro $l = 0, 1, \dots$ mezi následujícími dvěma kroky, dokud nedosáhneš konvergence:

1. Při fixním \mathbf{A} , spočti sloupce $\mathbf{B}^{(l+1)}$ z předpisu (2.30) s $\mathbf{a}_j = \mathbf{a}_j^{(l)}$, $l = 1, \dots, K$ (tj. minimalizuj (2.27) přes \mathbf{B} řešením K nezávislých problémů hřebenové regrese (2.29) s příslušnou fixací matice \mathbf{A}).
2. Při fixním \mathbf{B} , polož $\mathbf{A}^{(l+1)} = \mathbf{U} \mathbf{V}^\top$, kde \mathbf{U} a \mathbf{V} jsou ze singulárního rozkladu matice $\mathbf{X}^\top \mathbf{X} \mathbf{B}^{(l+1)}$.

Při fixním \mathbf{B} minimalizujeme jen první člen účelové funkce z (2.27), který lze psát jako $\|\mathbf{X} - \mathbf{XBA}^\top\|_F^2$, za podmínky ortonormality. Druhý krok v tomto alternujícím iteračním schématu je tudíž odůvodněn následujícím lemmatem.

Lemma 2.17. (*Procrusteova rotace snížené hodnosti.*)

Nechť $\mathbf{M} \in \mathbb{R}^{n \times p}$ a $\mathbf{N} \in \mathbb{R}^{n \times K}$. Nechť \mathbf{UDV}^\top je singulární rozklad $\mathbf{M}^\top \mathbf{N}$. Potom řešení úlohy

$$\min_{\mathbf{A} \in \mathbb{R}^{p \times k}} \|\mathbf{M} - \mathbf{NA}^\top\|_F^2 \quad \text{s.t.} \quad \mathbf{A}^\top \mathbf{A} = \mathbf{I},$$

je dáno předpisem $\widehat{\mathbf{A}} = \mathbf{UV}^\top$.

Důkaz lemmatu zde neuvádíme, lze jej najít v Zou a kol. (2006).

Poznamenejme, že reformulace (2.27) úlohy hledání prvních K nábojů se nápadně podobá úloze (1.4), tedy jedné z alternativních formulací AHK z první kapitoly. Pokud bychom v (2.27) přidali omezení $\mathbf{A} = \mathbf{B}$ a naopak vypustili penalizaci, dostali bychom přesně úlohu (1.4).

Dále, reformulace (2.27) se zaměřuje spíše na hledání příslušného hlavního podprostoru, než na jednotlivé náboje – to má opět společné s formulací (1.4). Všimněme si, že pokud je $(\widehat{\mathbf{A}}, \widehat{\mathbf{B}})$ dané jako řešení (2.27), musí být také $(\widehat{\mathbf{A}}, \widehat{\mathbf{B}})$ řešením této úlohy pro $\widetilde{\mathbf{A}} = \widehat{\mathbf{A}}\mathbf{R}$ a $\widetilde{\mathbf{B}} = \widehat{\mathbf{B}}\mathbf{R}$, kde $\mathbf{R} \in \mathbb{R}^{K \times K}$ je libovolná ortonormální matice. Je totiž $\widetilde{\mathbf{A}}\widetilde{\mathbf{B}}^\top = \widehat{\mathbf{A}}\mathbf{R}^\top\mathbf{R}\widehat{\mathbf{B}}^\top = \widehat{\mathbf{A}}\widehat{\mathbf{B}}^\top$, také $\widetilde{\mathbf{A}}^\top\widetilde{\mathbf{A}} = \widehat{\mathbf{A}}^\top\widehat{\mathbf{A}}$ a také

$$\sum_{j=1}^K \|\widetilde{\mathbf{b}}_j\|_2^2 = \|\widetilde{\mathbf{B}}\|_F^2 = \|\widehat{\mathbf{B}}\mathbf{R}\|_F^2 = \|\widehat{\mathbf{B}}\|_F^2 = \sum_{j=1}^K \|\widehat{\mathbf{b}}_j\|_2^2. \quad (2.32)$$

Tedy optimalizační úloha (2.27) není schopna rozlišit mezi rotovanými řešeními.

Nyní máme nový algoritmus pro hledání prvních K nábojů. Dovolili jsme si jej popsat jen neformálně a nebudeme vyšetřovat, zda konverguje, protože jeho praktický přínos je zanedbatelný – AHK lze provést snadno jedním singulárním rozkladem matice dat. Nicméně nastolená souvislost AHK s penalizovanou MNČ nám umožní snadno modifikovat tento algoritmus tak, aby produkoval řídké odhady nábojů. Předpoklad řídkosti nábojů přitom může sloužit jako náprava nekonzistence AHK v mnohorozměrné situaci s $p > n$, čímž se budeme zabývat v následující kapitole.

2.7 Shrnutí

V této kapitole jsme připomněli pravděpodobně nejznámější penalizační metody pro lineární model, tj. hřebenovou regresi a lasso. Hlavními důvody pro zavedení penalizace jsou přitom snahy o:

- zabránění jevu zvanému *overfitting*;
- snížení rozptylu odhadu regresních parametrů a tím zlepšení predikčních schopností modelu;
- zjednodušení výsledného modelu.

Velmi často bychom chtěli penalizovat ℓ_0 -normu regresních koeficientů, což však vede na NP-těžkou úlohu. Penalizace ℓ_1 -normy je často vnímána jako „konvexní

uvolnění“ pro úlohu s ℓ_0 -penalizací. Jak jsme však diskutovali, ℓ_1 -penalizované řešení může dávat poměrně špatnou aproximaci požadovaného ℓ_0 -penalizovaného odhadu. Proto (a z mnoha dalších důvodů) byly navrženy různé modifikace lasso metody, z nichž jsme diskutovali elastickou síť a adaptivní lasso.

V závěru kapitoly jsme ukázali, že úlohu AHK lze přeformulovat do tvaru, pro jehož řešení se přirozeně nabízí alternující minimalizační algoritmus. Základním stavebním kamenem tohoto algoritmu je opakované řešení opakované řešení úloh hřebenové regrese. Tyto úlohy v příští kapitole nahradíme za úlohy elastické sítě, abychom vynutili řídkost odhadů regresních parametrů.

3. Analýza řídkých hlavních komponent

Je dobře známo, že AHK může být nekonzistentní v mnohorozměrném limitním případě, v němž počet parametrů p i počet pozorování n jdou k nekonečnu v jistém poměru, přičemž počet pozorování neroste výrazně rychleji než počet parametrů. Příklad výsledku o nekonzistenci AHK jsme diskutovali v sekci 1.5. Tyto výsledky naznačují, že AHK by neměla být aplikována na úlohy, v nichž $p \approx n$ nebo dokonce $p > n$. V této situaci nelze očekávat, že výběrové náboje budou důvěryhodnými odhady jejich populačních protějšků. K zajištění užitečnosti projekce dat do podprostoru nižší dimenze v podobném smyslu jako při AHK je třeba přidat jisté strukturální předpoklady, například řídkost hledaných nábojů.

Úlohu hledání prvního řídkého náboje lze formulovat jako úlohu

$$\arg \max_{\mathbf{w} \in \mathbb{R}^p} \mathbf{w}^\top \mathbf{X}^\top \mathbf{X} \mathbf{w} \quad \text{s.t.} \quad \mathbf{w}^\top \mathbf{w} = 1, \|\mathbf{w}\|_0 \leq s. \quad (3.1)$$

Řídkost (omezení na ℓ_0 normu) efektivně snižuje počet koeficientů, které je třeba odhadnout v jednotlivých nábojích, a tím umožňuje konzistentní odhad i v mnohorozměrné situaci. Bohužel, Zatímco AHK je výpočetně jednoduchá úloha, řešitelná singulárním rozkladem matice dat, úloha (3.1), tedy *analýza řídkých hlavních komponent* (AŘHK), je NP-těžký problém. Přitom odhad jednoho náboje se za předpokladu řídkosti skládá ze dvou částí: odhadu opory náboje a odhadu koeficientů v opoře. K řešení úlohy (3.1) (nebo jejích různých reformulací) byla navržena pestrá škála algoritmů, z nichž si některé představíme v této kapitole. Všechny algoritmy přitom uvažují jednu z ekvivalentních formulací úlohy AHK, zavádějí do ní ℓ_0 -penalizaci, a výsledný problém pak uvolňují na konvexní úlohu (v podobném smyslu jako je ℓ_1 -penalizace považována za konvexní uvolnění ℓ_0 -penalizace), případně vynucují řídkost hlavních komponent heuristicky. Ukázka jednoduchého algoritmu pro nalezení prvního náboje je dána v následujícím příkladě.

Příklad 3.1. (*Algoritmus pro AŘHK, Lu a Johnstone, 2009.*)

Nechť je dána centrovaná matice $\mathbf{X} \in \mathbb{R}^{n \times p}$. Naším cílem je nalézt první řídký náboj, tj. najít nějaké sub-optimální řešení úlohy (3.1) pro jisté $s \in [p]$.

Spočítejme rozptyly jednotlivých proměnných $\hat{\sigma}_j^2 = \sum_{i=1}^n x_{i,j}^2$. S jejich pomocí odhadněme oporu prvního náboje jako $\hat{S} = \{j; \hat{\sigma}_j^2 \geq c\}$, kde c je nějaká konstanta¹ zvolená například tak, aby S měla požadovanou kardinalitu.

Nechť $\hat{\mathbf{w}}_{\hat{S}} \in \mathbb{R}^{|\hat{S}|}$ je řešením úlohy

$$\arg \max_{\mathbf{w}_{\hat{S}} \in \mathbb{R}^{|\hat{S}|}} \mathbf{w}_{\hat{S}}^\top \mathbf{X}_{\hat{S}}^\top \mathbf{X}_{\hat{S}} \mathbf{w}_{\hat{S}} \quad \text{s.t.} \quad \mathbf{w}_{\hat{S}}^\top \mathbf{w}_{\hat{S}} = 1, \quad (3.2)$$

kde $\mathbf{X}_{\hat{S}} \in \mathbb{R}^{n \times |\hat{S}|}$ je matice vzniklá z matice \mathbf{X} vypuštěním sloupců, které nepatří

¹Lu a Johnstone (2009) volí c jako vhodný násobek kvantilu χ^2 -rozdělení.

do \widehat{S} . Doplňme nyní toto řešení o nuly na pozicích, které nepatří do \widehat{S} , tj. poloźme

$$\tilde{a}_{1,j} = \begin{cases} \widehat{w}_j, & j \in \widehat{S}, \\ 0, & j \notin \widehat{S}. \end{cases}$$

Vektor $\tilde{\mathbf{a}}_1$ je první řídký náboj. △

Zdůrazněme, že při studiu konzistence AŘHK je třeba striktně rozlišovat mezi konzistencí skutečného řešení úlohy (3.1), které je však v praxi nedosažitelné, a konzistencí nějakého sub-optimálního řešení, které bylo získáno nějakým výpočetně schůdným algoritmem, jako je algoritmus z předchozího příkladu. Pod pojmem „řídký náboj“ myslíme v této práci výstup nějakého algoritmu pro AŘHK, který má mnoho složek nulových. Pro přehledné shrnutí výsledků ohledně konzistence (AŘHK) jakožto konzistence skutečného řešení úlohy (3.1) odkazujeme čtenáře na Shena a kol. (2013). Další reference lze najít v Gu a kol. (2014).

Konzistence různých algoritmů pro AŘHK byla ukázána v různých modelech za různých předpokladů. Johnstone a Lu (2009), kteří formálně ukázali nekonzistenci AKH v normálním špičatém modelu s jednou komponentou, také ukázali, že při použití jednoduchého postupu předvýběru opory (viz příklad 3.1) lze odhadnout řídký náboj konzistentně (za jistých dodatečných předpokladů). Jejich výsledky byly rozšířeny v Paul (2007) na model o více komponentách. Amini a Wainwright (2008) ukázali konzistenci SDP algoritmu d’Aspremonta a kol. (2007), který je založený na semidefinitním uvolnění úlohy (3.1). Paul a Johnstone (2012) navrhli odlišný algoritmus založený na rozšířeném lagrangiánu spolu s důkazem konzistence. A konečně, Ma (2013) představil algoritmus iterativního prahování, který lze vnímat jako kompetitivnější iterativní verzi algoritmu Johnstonea a Lua (2009), spolu s důkazem konzistence. Poznamenejme, že všechny dosud citované prameny vyšetřují konvergenci v normálním špičatém modelu a navržené algoritmy se zaměřují na postupnou extrakci vlastních vektorů výběrové kovarianční matice. Druhé zmíněné může být z praktického hlediska nevhodné, jak bude diskutováno později.

Dalším problémem AHK (kromě „ztracené“ konzistence) v mnohorozměrném případě je špatná interpretovatelnost. Tento problém může být z praktického hlediska ještě důležitější než nekonzistence. Obyčejná AHK produkuje odhady nábojů, jejichž všechny koeficienty jsou typicky nenulové. To prakticky znemožňuje interpretaci jednotlivých os v hlavním podprostoru, pokud není p hodně malé. Z tohoto důvodu byly navrženy různé modifikace AHK, jako například pokusy o rotaci nábojů jakožto báze hlavního prostoru tak, aby co nejvíc koeficientů rotovaných nábojů bylo rovných nebo alespoň blízkých nule (Jolliffe, 1995). Mnohé algoritmy pro AŘHK lze vnímat pouze jako heuristiky řešící tento interpretační problém. Jednotlivé heuristiky vynucují různými prostředky nulovost koeficientů v nábojích a existují pro ně jen omezené teoretické záruky.

Moghaddam a kol. (2006) a d'Aspremont a kol. (2008) navrhli hladové algoritmy s triviální zárukou *globální konvergence*². Alespoň záruku globální konvergence mají také algoritmy Sigga a Buhmanna (2008), Journeého a kol. (2010), Maa (2011) a Lua a Zhanga (2012). Žádné záruky si nejsme vědomi pro známé algoritmy Jolliffa a kol. (2003), Zoua a kol. (2006), Shena a Huanga (2008), Wittenové a kol. (2009) nebo Jenattona a kol. (2010).

Pokud je nám známo, v literatuře se dostává nejvíce prostoru SDP algoritmu d'Aspremonta a kol. (2007) (viz např. renomované práce Bertheta a Rigolleta, 2013a a 2013b, a reference v nich), zatímco z praktického hlediska nejatraktivnějšími jsou algoritmy Zoua a kol. (2006), Sigga a Buhmanna (2008), případně Shena a Huanga (2008). Důvody jsou následující.

Amini a Wainwright (2008) ukázali, že v asymptotickém případě $n, p, s \rightarrow \infty$, ve kterém škálovaný počet pozorování $n/(s \log(p-s)) \rightarrow \tau \in [0, \infty]$ (s je kardinalita prvního náboje a n je počet pozorování tažený z jistého obecnějšího modelu, než je normální špičatý model), odhadne SDP algoritmus d'Aspremonta a kol. (2007) správně oporu prvního náboje, je-li τ větší než jistá konstanta τ_0 . Naopak pravděpodobnost správného odhadu opory při použití libovolného algoritmu se v tomto modelu limitně blíží nule je-li τ menší než jistá konstanta τ_1 . Amini a Wainwright (2008) tedy ukázali, že SDP algoritmus je (za jistých předpokladů, které v našem přehledu neuvádíme) asymptoticky optimální. Přesto však četné simulační studie porovnávající chování různých AŘHK algoritmů buď ukazují, že odhady získané SDP algoritmem jsou sub-optimální (např. Lu a Zhang 2012, nebo Ma, 2011), nebo SDP algoritmus vůbec neuvažují kvůli vysoké výpočetní náročnosti, nemožnosti volit přesně kardinalitu výstupních nábojů, nutnosti extrahovat náboje po jednom nebo nutnosti pracovat s příliš velkou kovarianční maticí (např. Shen a Huang, 2008, nebo Journeé a kol., 2010, a také tato práce).

Naopak, zejména S-PCA algoritmus Zoua a kol. (2006) má mnoho praktických výhod, které budou diskutovány v sekci 3.4.3. Nejspíše kvůli své praktické aplikovatelnosti se chování S-PCA algoritmu stalo standardní dolní mezí, kterou musejí nově navrhované algoritmy překonat. S-PCA algoritmus byl uvažován v drtivé většině experimentálních studií v pramenech citovaných výše a ve většině z nich byl na základě více či méně přesvědčivých důkazů výkonnostně překonán.

V naší práci zaujmeme praktický a algoritmicky orientovaný přístup. Zaměříme se na S-PCA algoritmus Zoua a kol. (2006) a jeho modifikace. Dále budeme pro porovnání uvažovat hladový algoritmus Moghaddama a kol. (2006) a patrně nejjednodušší možný přístup k AŘHK spočívající v prahování výsledných nábojů (tzv. *jednoduché prahování*). Ještě před popisem jednotlivých algoritmů v sekci 3.4 si v následujících sekcích přiblížíme několik rozdílů, které přináší řídkost oproti klasické AHK.

²Globální konvergencí rozumíme konvergenci proměnných k nějakému bodu, ne nutně globálnímu optimu, pro libovolnou inicializaci algoritmu. Naše optimalizační terminologie kopíruje terminologii Langeho (2013).

3.1 Deflační metody

Mnohé algoritmy jsou navrženy tak, aby hledaly náboje sekvenčně po jednom (Moghaddam a kol., 2006, d'Aspremont a kol., 2007, nebo algoritmus z příkladu 3.1). Po obdržení prvního náboje daným algoritmem je třeba provést deflaci kovarianční matice (případně matice dat) o extrahovaný náboj a následně aplikovat algoritmus znovu na matici po deflaci. Deflaci přitom rozumíme proces odebrání vektoru \mathbf{v} z množiny vlastních vektorů matice \mathbf{M} . V případě, že \mathbf{v} je vlastní vektor matice \mathbf{M} , lze tuto operaci snadno provést vynulováním příslušného vlastního čísla.

Věta 3.2. (*Hotellingova deflace.*)

Nechť $\lambda_1 \geq \dots \geq \lambda_p$ a $\mathbf{v}_1, \dots, \mathbf{v}_p$ jsou vlastní čísla a příslušné vlastní vektory pozitivně semidefinitní matice $\mathbf{M} \in \mathbb{R}^{p \times p}$. Pak pro $j = 1, \dots, p$ má matice $\widetilde{\mathbf{M}}_j = \mathbf{M} - \lambda_j \mathbf{v}_j \mathbf{v}_j^\top$ vlastní vektory $\mathbf{v}_1, \dots, \mathbf{v}_p$ s příslušnými vlastními čísly $\lambda_1, \dots, \lambda_{j-1}, 0, \lambda_{j+1}, \dots, \lambda_p$.

Důkaz. Pro $i \neq j$ platí

$$\widetilde{\mathbf{M}}_j \mathbf{v}_i = \mathbf{M} \mathbf{v}_i - \lambda_j \mathbf{v}_j \mathbf{v}_j^\top \mathbf{v}_i = \mathbf{M} \mathbf{v}_i = \lambda_i \mathbf{v}_i,$$

zatímco

$$\widetilde{\mathbf{M}}_j \mathbf{v}_j = \mathbf{M} \mathbf{v}_j - \lambda_j \mathbf{v}_j \mathbf{v}_j^\top \mathbf{v}_j = \lambda_j \mathbf{v}_j - \lambda_j \mathbf{v}_j = 0. \quad \square$$

Klasickou AHK tedy lze provádět sekvenčně. Po obdržení prvního náboje $\hat{\mathbf{a}}_1$ (a vlastního čísla, které je rovno rozptylu vysvětlenému pomocí první HK) stačí provést deflaci kovarianční matice dle předchozí věty a druhý náboj $\hat{\mathbf{a}}_2$ hledat jako první vlastní vektor nové matice. Tento postup je oblíbenou alternativou hledání vlastních vektorů matice, kdy jednotlivé vlastní vektory jsou získávány iterativně pomocí tzv. *mocninné iterační metody* (*power iteration method*). Obdobně lze postupovat při hledání nábojů jakožto pravých singulárních vektorů s deflací matice dat.

Co když jsme ale pomocí nějaké AŘHK metody nenašli skutečný vlastní vektor, ale nějaký řídký pseudo-vlastní vektor \mathbf{v} ? Zdůrazněme, že smyslem AŘHK je hledat vektory, které sice nejsou přímo vlastními vektory výběrové kovarianční matice $\widehat{\Sigma}$, ale my přitom doufáme, že jsou dobrými odhady vlastních vektorů skutečné kovarianční matice Σ , která nám však není známa. První řídký náboj tedy není vlastním vektorem $\widehat{\Sigma}$. Deflaci je však třeba provést na matici $\widehat{\Sigma}$, abychom mohli sekvenčně získat další řídký náboj.

Všimněme si, že jednoduchý důkaz předchozí věty závisí na tom, že vektor \mathbf{v} , kterým provádíme deflaci matice \mathbf{M} , je jejím skutečným vlastním vektorem. Pokud tomu tak není, nedochází k vynulování žádného vlastního čísla. V takovém případě sice lze stále chápat deflaci jako jakési snížení „špičatosti“ matice \mathbf{M} ve směru \mathbf{v} , ale při deflaci nevlastním vektorem může nastat množství problémů. Například se může stát, že matice po deflaci, tj. matice

$$\mathbf{M} - (\mathbf{v}^\top \mathbf{M} \mathbf{v}) \mathbf{v} \mathbf{v}^\top, \quad (3.3)$$

již není pozitivně semidefinitní, i když matice \mathbf{M} byla, viz následující příklad. To znamená, že algoritmy hledající řídké náboje sekvenčně a zároveň spoléhající na pozitivní semidefinitnost každé matice, z níž náboj extrahují (kromě prvního náboje je to vždy již matice po deflaci), by neměly užívat Hotellingovu deflaci. Přesto to například sekvenční algoritmy Moghaddama a kol. (2006) nebo d'Aspremonta a kol. (2007) dělají a zcela ignorují problémy s tím spojené.

Příklad 3.3. Uvažujme matici Σ danou rovností (1.2) z příkladu 1.3. Vlastní čísla této matice jsou $\lambda_1 = 4$ a $\lambda_2 = 1$. Po provedení Hotellingovy deflace řídkým vektorem $\mathbf{v}_1 = (1,0)^\top$ dostáváme matici

$$\begin{pmatrix} 0 & 3/2 \\ 3/2 & 5/2 \end{pmatrix},$$

jejíž vlastní čísla jsou $\lambda_1 \approx 3.2$ a $\lambda_2 \approx -0.7$, tj. matici, která není pozitivně semidefinitní. Podobná situace by nastala při deflaci druhým možným řídkým vektorem $\mathbf{v}_2 = (0,1)^\top$ namísto vektorem \mathbf{v}_1 . Podobná situace by také nastala při deflaci výběrové kovarianční matice $\hat{\Sigma}$ z příkladu 1.8 pomocí vektoru \mathbf{v}_1 nebo \mathbf{v}_2 . \triangle

Jednoduchá úprava deflačního schématu pro AŘHK umožňující vyhnout se výše zmíněnému problému s pozitivní definitností, je následující. Po získání řídkého náboje $\tilde{\mathbf{a}}_1$ v prvním kroku proved místo deflace pomocí $\tilde{\mathbf{a}}_1$ deflaci pomocí obyčejného náboje $\hat{\mathbf{a}}_1$, tj. vlastního vektoru. Toto schéma však přináší další problémy. Z kovarianční matice je odfiltrován rozptyl, který ještě nebyl vysvětlen, tedy každá další řídká HK pak může být sub-optimální z hlediska vysvětleného rozptylu, a také následující řídké náboje $\tilde{\mathbf{a}}_2, \dots$ nemusejí být ortogonální k předchozím nábojům.

Mackey (2009) diskutuje další problémy, které mohou deflací pomocí nevlastního vektoru u AŘHK nastat. Také navrhuje deflační postupy alternativní k Hotellingově deflaci, které uvedené problémy do jisté míry (nikdy však zcela) řeší. Nicméně je zřejmé, že metody extrahující všechny řídké komponenty najednou mají neoddiskutovatelnou výhodu v tom, že se zmíněným problémům zcela vyhýbají.

3.2 Důležitost opory nábojů

Pokud bychom v klasické regresi věděli, které proměnné jsou relevantní, pracovali bychom pouze s nimi a irelevantní proměnné bychom ignorovali. Podobně, pokud bychom znali oporu prvního řídkého náboje, získali bychom jeho nenulové koeficienty jako koeficienty klasického náboje příslušné podmatice (například ve výběrovém případě vypuštěním těch sloupců matice \mathbf{X} , které jsou příslušné nulovým koeficientům).

Předpokládejme nyní, že jsme získali první řídký náboj $\tilde{\mathbf{a}}_1$ pomocí libovolného algoritmu, tj. $\tilde{\mathbf{a}}_1$ je sub-optimální řešení úlohy 3.1. Necht $\hat{S} = \text{supp}(\tilde{\mathbf{a}}_1)$. Potom řešení úlohy (3.2) doplněné o nuly na pozicích, které nepaří do S , má zřejmě stejnou oporu jako $\tilde{\mathbf{a}}_1$, přičemž z definice AHK musí (pseudo-)hlavní komponenta s tímto řídkým nábojem vysvětlovat přinejmenším stejné množství rozptylu jako

řídka (pseudo-)hlavní komponenta s řídkým nábojem $\tilde{\mathbf{a}}_1$. Toto jednoduché pozorování vedlo Moghaddama a kol. (2006) ke studiu kombinatorických algoritmů, které nejprve odhadují oporu a následně odhadují nenulové koeficienty nábojů řešením úlohy (3.2) s příslušnou S .

Moghaddam a kol. (2006) tvrdí, že z řešení každého algoritmu pro AŘHK by měla být extrahována pouze jeho opora a řídký náboj by měl být dopočítán řešením příslušné podúlohy (3.2). Bez tohoto postupu, kterému říká „variační oprava“, podle něj nemají „spojité“ algoritmy jako například S-PCA (Zou a kol., 2006) šanci být konkurenceschopné co se týče vysvětleného rozptylu. My nesouhlasíme ani s jedním z těchto tvrzení. Zaprvé, variační oprava jde zcela proti paradigmatu, že metody odhadu by měly být spojité (viz např. Fan a Li, 2006). Zavedením této umělé nespojitosti do jinak spojitých metod jsme sice v naší simulační studii docílili nepatrného zlepšení ve vysvětleném rozptylu, avšak na úkor celkového zhoršení odhadu měřeném úhlem mezi skutečným nábojem a jeho řídkým odhadem (tyto výsledky neuvádíme). Zadruhé, jak uvidíme v simulační studii v páté kapitole, hladový algoritmus Moghaddama a kol. (2006) dává překvapivě špatné výsledky. Proto „variační opravu“ nepoužíváme pro jiný algoritmus než hladový algoritmus. Výše uvedené pozorování nicméně podtrhává důležitost správného odhadu opory nábojů.

Správně odhadnuté opory nábojů jsou také velmi důležité z hlediska interpretace výsledků AŘHK. V sekci 1.4 jsme diskutovali pravděpodobnostní interpretaci HK a jejich spojitost s modelem faktorové analýzy. Tvrdili jsme, že na náboje se lze dívat jako na projekce faktorů do domény dat. Předpokládejme teď na okamžik, že matice \mathbf{W} ve faktorovém modelu (1.14) je ortogonální a její sloupce jsou seřazeny sestupně podle normy. V takovém případě jsou jednotlivé koeficienty prvního náboje obrazem prvního faktoru v doméně pozorovaných dat. To znamená, že při odhadu nábojů se vlastně snažíme popsat faktor pomocí pozorovaných proměnných. Nulové koeficienty v odhadu např. prvního náboje nám pak říkají, které proměnné nesouvisejí s prvním faktorem. Ačkoliv tato interpretace AHK a faktorové analýzy je již značně odtrhnutá od matematického modelu, může být často nápomocná v praxi, jak uvidíme při analýze reálných dat v kapitole 6. I pokud upustíme od této volné interpretace, je-li i -tý koeficient nulový společně pro všech uvažovaných K nábojů, znamená to, že i -tá proměnná je irelevantní pro uvažovaný problém. Je patrné, že z tohoto úhlu pohledu může mít špatný odhad opor nábojů katastrofální dopad na interpretaci modelu.

3.3 Rozptyl vysvětlený řídkými komponentami

Jak jsme viděli v první kapitole (pozn. 1.5), pro obyčejnou AHK je nekorelovanost HK ekvivalentní s ortogonalitou nábojů. Klíčem byl fakt, že náboje jsou vlastní vektory kovarianční matice. Pokud ale provádíme modifikovanou AHK a například řídké náboje již nejsou pravými vlastními vektory, nelze požadovat, aby platila jak ortogonalita nábojů tak nezávislost hlavních komponent (Jolliffe a kol., 1995). Běžnou praxí přitom je obětovat nekorelovanost a hledat ortogonální náboje, tj. nějakou ortogonální bázi hlavního podprostoru.

Pokud ale řídké HK nejsou nekorelované, je příliš optimistické počítat rozptyl

vysvětlený prvními K řídkých HK jako součet rozptylů jednotlivých ŘHK. Zou a kol. (2006) proto navrhují definovat rozptyl vysvětlený j -tou ŘHK jako rozptyl reziduí z projekce j -té řídké HK na prvních $j - 1$ řídkých HK. Pro detaily odkazujeme na Zou a kol. (2006, sekce 3.4).

Pokud řídké HK nejsou ani nekorelované ani nemají ortogonální náboje (což nastává pro všechny algoritmy, které extrahují náboje postupně, jak jsme diskutovali v minulé sekci), je pohled na vysvětlený rozptyl z předchozího odstavce stále příliš optimistický. V takovém případě je vhodné použít úpravu z Shen a Huang (2008). Nicméně, my používáme rozptyl vysvětlený řídkými HK ve smyslu z Zou a kol. (2006), jelikož většina našich metod produkuje ortogonální náboje. Výjimkami jsou jednoduché prahování a hladový algoritmus, pro ty může být vysvětlený rozptyl spočtený v naší simulační studii nadhodnocený.

Konečně, v našich numerických příkladech vždy uvádíme rozptyl vysvětlený řídkou hlavní komponentou dělený rozptylem příslušné komponenty, což je také běžnou praxí. Tento podíl je vždy mezi nulou a jedničkou, jelikož z definice AHK nemůže žádná lineární kombinace dat vysvětlovat větší množství rozptylu než klasická první HK. Pro další HK se teoreticky může stát, že bude řídká modifikace vysvětlovat větší množství rozptylu než příslušná klasická HK, v praxi však tento jev nepozorujeme.

Poznámka 3.4. Podobně jako u AHK vyvstává i v případě AŘHK otázka, jak zvolit počet hlavních komponent K . V případě AŘHK je navíc třeba určit kardinality jednotlivých nábojů. Běžnou praxí je zvolit počet obyčejných hlavních komponent stejně jako v případě AHK a následně volit kardinality jednotlivých nábojů tak, aby řídké hlavní komponenty vysvětlovaly velkou část (např. 90 procent) rozptylu vysvětleného obyčejnými náboji. Tento postup však nelze doporučit hned z několika důvodů.

1. Johnstone (2001) diskutuje, jak samotná volba počtu obyčejných HK na základě vysvětleného rozptylu může být zavádějící v mnohorozměrném případě.
2. Volíme-li počet komponent a kardinality nábojů na základě rozptylu a odhadujeme-li pak ještě samotné náboje na základě (případně upravené verze) úlohy (3.1), dochází k příliš mnoha sekvenčním rozhodnutím pouze na základě vysvětleného rozptylu.
3. Snaha o dosažení co největšího vysvětleného rozptylu může mít negativní dopad na celkovou kvalitu řídkých odhadů nábojů. Je třeba si uvědomit, že základním problémem v mnohorozměrné situaci je příliš velká kumulace šumu napříč jednotlivými dimenzemi, tj. rozptylu, který se nepokoušíme vysvětlit.

Z těchto důvodů je velmi obtížné, ne-li nemožné, kvantifikovat nějaké obecné rady, jak postupovat při volbě počtu hlavních komponent a kardinalit příslušných nábojů. K provedení této volby je třeba zužitkovat každou dodatečnou informaci dostupnou v daném příkladě. \diamond

3.4 Algoritmy

V této sekci si představíme několik algoritmů pro AŘHK. Všechny uvedené algoritmy mají společné to, že umožňují přesně volit počet nenulových koeficientů v jednotlivých nábojích.

U všech algoritmů uvádíme časovou a paměťovou složitost tak, jak je uvedeno v původních pramenech. Nicméně je třeba poznamenat, že tato čísla jsou pouze orientační pro vzájemné porovnání efektivity jednotlivých algoritmů. Faktické doby běhu (a do jisté míry i teoretické složitosti algoritmů) se mohou značně lišit v závislosti na algoritmech použitých k nalezení spektrálních nebo singulárních rozkladů matic, případně řešení lasso úloh. Detailní diskuze výpočetních vlastností uvažovaných algoritmů není obsahem této práce.

3.4.1 Jednoduché prahování (ST-PCA)

Nejjednodušší, zcela heuristický přístup k AŘHK je tzv. jednoduché prahování (*simple thresholding*, ST-PCA), které spočívá v nastavení nejmenších koeficientů v nábojích na nulu a následným standardizováním obdrženého vektoru. Je-li $\hat{\mathbf{a}}_j$ j -tý náboj, jednoduché prahování spočívá v nastavení

$$c_i = \hat{a}_{j,i} \mathbb{1}_{[|\hat{a}_{j,i}| \geq \gamma]}, \quad i = 1, \dots, p,$$

kde $\gamma > 0$ je práh. Požadujeme-li, aby byl přesný počet s koeficientů tohoto náboje nastaven na nulu, volíme γ jako s -tý v absolutní největší prvek $\hat{\mathbf{a}}_j$. Následně obdržený vektor \mathbf{c} standardizujeme na řídký náboj položením

$$\tilde{\mathbf{a}}_j = \frac{\mathbf{c}}{\|\mathbf{c}\|_2}.$$

Tímto způsobem samozřejmě nedostaneme ani ortogonální náboje, ani nekorelované HK. Alternativou je přechod k postupné extrakci nábojů po jednom spolu s užitím deflačních metod, což někteří autoři také uvažují (např. d'Aspremont, 2008), ale my se snažíme deflaci vyhnout z důvodů uvedených výše.

Celkově vzato, jednoduché prahování je patrně tím nejjednodušším přístupem k AŘHK a jeho numerický výkon se stal standardním měřítkem pro složitější algoritmy. Navzdory nesmírné jednoduchosti tohoto přístupu, numerické výsledky obdržené jednoduchým prahováním mnohdy nejsou tak špatné, jak by mohl člověk čekat, na což také upozorňují mnozí autoři (např. Moghaddam a kol., 2006). Časová složitost jednoduchého prahování závisí na použitém algoritmu pro spektrální rozklad nebo singulárním rozklad matice. Naše nejjednodušší verze spočívá jen v provedení jednoho singulárního rozkladu, tedy má časovou složitost $O(np^2)$ a paměťovou složitost je $O(np)$.

Algoritmus 3.4.1: Greedy-PCA algoritmus Moghaddama a kol. (2006).

input : Centrovaná matice dat $\mathbf{X} = (\mathbf{x}_1 | \dots | \mathbf{x}_p) \in \mathbb{R}^{n \times p}$,
požadovaná kardinalita prvního náboje s .

output: První řídký náboj $\tilde{\mathbf{a}}_1$.

```
1 begin
2   Seřad proměnné sestupně podle jejich výběrového rozptylu.
3   Polož  $I_1 = \{1\}$ .
4   for  $j \in \{2, \dots, s\}$  do
5     Najdi index  $i$  takový, že první vlastní číslo matice  $\sum_{l \in I_{j-1} \cup \{i\}} \mathbf{x}_l \mathbf{x}_l^\top$ 
        je co největší.
6     Polož  $I_j = I_{j-1} \cup \{i\}$ .
7   end
8   Necht  $\tilde{\mathbf{a}}_{I_s}$  je první vlastní vektor matice  $\sum_{l \in I_s} \mathbf{x}_l (\mathbf{x}_l)^\top$ . Doplň jej
        nulami na příslušných pozicích abys získal  $\tilde{\mathbf{a}}_1$ .
9 end
```

3.4.2 Hladový algoritmus (Greedy-PCA)

Pro širší porovnání jsme se rozhodli vybrat další velmi jednoduchou metodu pro AHRK (implementačně jednoduchou, nikoliv jednoduchou z hlediska paměťové nebo časové složitosti). Důvody, proč jsme vybrali právě hladový algoritmus (Greedy-PCA) Moghaddama a kol. (2006) jsou v zásadě dva. Zaprvé kvůli jeho částečné podobnosti s LARS algoritmem pro lasso (který využívá algoritmus navržený Zouem a kol., 2006). Greedy-PCA také udržuje množinu aktivních proměnných, do které postupně přidává. Druhým důvodem je, že Moghaddam a kol. (2006) uzavírají simulační studii tím, že jejich algoritmus produkuje lepší výsledky než algoritmus Zoua a kol. (2006). Jak uvidíme v následující kapitole, naše závěry jsou zcela opačné.

Greedy-PCA hledá nejprve oporu náboje S , samotný náboj je spočítán až následně jako řešení podproblému (3.2). Počáteční opora je tvořena pouze indexem proměnné s největším výběrovým rozptylem. V každém kroku je do opory přidán jeden index – takový, který maximálně zvýší vysvětlený rozptyl. Takto postupujeme, dokud opora nemá požadovanou kardinalitu, případně dokud nejsou přidány všechny proměnné. V takovém případě získáme celou dráhu řešení.

Algoritmus (3.4.1) je základní verzí algoritmu z Moghaddam a kol. (2006). Autoři diskutují různá vylepšení jak kombinovat průchod popředu s průchodem pozadu, případně jak využít různých triků kombinatorické optimalizace k zefektivnění algoritmu. My však uvažujeme pouze základní verzi, která byla porovnána s dalšími algoritmy také v d'Aspremont a kol. (2008). Zajímavé je, že d'Aspremont a kol. (2008) navrhuje jiný hladový algoritmus s nižší složitostí ($O(p^3)$ pro výpočet celé dráhy řešení, oproti $O(p^4)$ operací potřebných pro Greedy-PCA), který se chová o něco málo hůře než Greedy-PCA, jehož řešení se de-facto snaží aproximovat. Oba algoritmy pak v numerické studii dávají lepší výsledky než ST-PCA. My však v naší simulační studii pozorujeme přesně opačný trend. Numerické výsledky

d'Aspremonta a kol. (2008) se nám přitom nepodařilo zreplikovat. Paměťová složitost Greedy PCA (i jeho „zrychlené verze“ d'Aspremonta a kol, 2008) je také $O(p^2)$ vzhledem k nutnosti pracovat s kovarianční maticí.

3.4.3 Regresní algoritmus (S-PCA)

Zou et al. (2006) přeformulovali AHK do regresního tvaru. Jak jsme připomněli v sekci 2.6, AHK lze formulovat jako úlohu (2.27), kterou lze řešit iterativně pomocí alternujícího algoritmu. Přidáním ℓ_1 -penalizace do této úlohy můžeme vynutit řídkost hledaných nábojů:

$$\begin{aligned} (\widehat{\mathbf{A}}, \widehat{\mathbf{B}}) = \arg \min_{\substack{\mathbf{A}, \mathbf{B} \in \mathbb{R}^{p \times K} \\ \text{s.t.} \quad \mathbf{A}^\top \mathbf{A} = \mathbf{I}_K}} \sum_{i=1}^n \|\mathbf{x}^{(i)} - \mathbf{A}\mathbf{B}^\top \mathbf{x}^{(i)}\|_2^2 + \lambda_2 \sum_{j=1}^K \|\mathbf{b}_j\|_2^2 + \sum_{j=1}^K \lambda_{1,j} \|\mathbf{b}_j\|_1 \end{aligned} \quad (3.4)$$

kde $\lambda_2, \lambda_{1,1}, \dots, \lambda_{1,K} > 0$ jsou ladící parametry.

Algoritmus pro řešení AHŘK úlohy (3.4) se nabízí jak drobná modifikace postupu, který jsme diskutovali v sekci 2.6. Jediným rozdílem je, že místo úloh hřebenové regrese nyní budeme řešit úlohy tvaru elastické sítě

$$\arg \min_{\mathbf{b}_j \in \mathbb{R}^p} \|\mathbf{X}\mathbf{a}_j - \mathbf{X}\mathbf{b}_j\|_2^2 + \lambda_2 \|\mathbf{b}_j\|_2^2 + \lambda_{1,j} \|\mathbf{b}_j\|_1, \quad j = 1, \dots, K. \quad (3.5)$$

Jako startovní bod algoritmu se nyní přirozeně nabízí řešení obyčejné AHK pomocí singulárního rozkladu \mathbf{X} . Celý postup je shrnut jako algoritmus 3.4.2.

Poznámka 3.5. S-PCA algoritmus má několik atraktivních vlastností, zejména z praktického úhlu pohledu:

1. požadovaná kardinalita (tj. požadovaný počet nenulových koeficientů) může být zvolena pro každý náboj;
2. zvolený počet ŘHK a příslušných nábojů je spočten najednou;
3. algoritmus nevyžaduje výpočet ani ukládání kovarianční matice;
4. výsledné náboje jsou ortogonální;
5. při uvolnění omezení na kardinalitu nábojů se metoda redukuje na obyčejnou AHK.

Vlastnosti (a)-(c) vyplývají z vlastností LARS algoritmu, který je použit k řešení úloh (3.5). Zde poznamenejme, že v původním článku autoři vlastnost (a) nevyzdvihují a algoritmus nediskutují příliš podrobně, což zřejmě vede k tomu, že vlastnost (a) bývá S-PCA algoritmu někdy upírána (Journée a kol., 2010). Vlastnost (d) vyplývá z ortogonality \mathbf{A} . Vlastnost (e) vyplývá z věty 2.16. K tomu také poznamenejme, že úloha 3.4 je spojitá v ladících parametrech. Navíc, výstup algoritmu se liší minimálně pro různé volby parametru λ_2 (vzpomeňme, že regresní PCA algoritmu ze sekce 2.6 byl na volbě ladícího parametru pro ℓ_2 -penalizaci zcela nezávislý). \diamond

Algoritmus 3.4.2: S-PCA algoritmus Zoua a kol. (2006).

input : Centrovaná matice dat $\mathbf{X} \in \mathbb{R}^{n \times p}$, ladící parametr λ_2 .
Počet hledaných řídkých nábojů $K \in [p]$.
Ladící parametry $\lambda_{1,1}, \dots, \lambda_{1,K}$ (nebo ekvivalentně požadované hodnoty kardinality nábojů s_1, \dots, s_K).
Požadovaná přesnost ξ_{conv} (defaultně 10^{-3}).
output: Řídké náboje ve sloupcích matice $\mathbf{B} \in \mathbb{R}^{p \times K}$.

```
1 begin
2   Polož  $l = 0$ .
3   Polož  $\mathbf{A}^{(0)} = \widehat{\mathbf{A}}$ , kde  $\widehat{\mathbf{A}} \in \mathbb{R}^{p \times K}$  je matice prvních  $K$  obyčejných
   nábojů.
4   repeat
5     for  $j \in \{1, \dots, K\}$  do
6       Získej  $\mathbf{b}_j^{(l+1)}$  jako řešení  $j$ -té z úloh (3.5) s  $\mathbf{a}_j = \mathbf{a}_j^{(l)}$  pomocí
       LARS algoritmu.
7       Znormuj  $\mathbf{b}_j^{(l+1)}$ .
8     end
9     Polož  $\mathbf{B}^{(l+1)} = (\mathbf{b}_1^{(l+1)} | \dots | \mathbf{b}_K^{(l+1)})$ .
10    Spočti singulární rozklad  $\mathbf{X}^T \mathbf{X} \mathbf{B}^{(l+1)} = \mathbf{U} \mathbf{D} \mathbf{V}^T$  a polož
     $\mathbf{A}^{(l+1)} = \mathbf{U} \mathbf{V}^T$ .
11    Polož  $l = l + 1$ .
12  until  $\forall i \in [p], j \in [K] : |b_{ij}^{(l)} - b_{ij}^{(l-1)}| < \xi_{\text{conv}}$ .
13 end
```

Na druhou stranu, S-PCA je typem algoritmu alternující minimalizace, a jsou-li algoritmy tohoto typu použity k řešení nekonvexních úloh, teoretické záruky pro nalezení řešení a jeho případné vlastnosti existují spíše výjimečně. Přestože jako zastavující pravidlo S-PCA je použita konvergence matice \mathbf{B} , není v současnou chvíli zřejmé, že algoritmus skutečně konverguje³, případně kam. Zou a kol. (2006) konvergenci algoritmu vůbec nezmiňují. Wittenová a kol. (2009) sice konstatují, že nelze garantovat konvergenci ke globálnímu minimu, ale otázka numerické konvergence (k lokálnímu minimu) zůstává nezodpovězena. Jak ukážeme v sekci 4.4.2, S-PCA algoritmus konverguje k lokálnímu minimu účelové funkce (3.4) pro libovolnou inicializaci. Inicializace pomocí obyčejné AHK se pak ukazuje být rozumným startovacím bodem v konkrétních příkladech.

Časová složitost algoritmu je $mO(p^3)$, kde m je počet iterací potřebných ke konvergenci, pokud bereme, že počet hledaných komponent je konstantní (což děláme při uvádění časové složitosti všech algoritmů). Časovou složitost lze ještě snížit na $mO(np^2)$, pokud budeme uvažovat, že kardinalita jednotlivých komponent je srovnatelná s n , což je při praktickém užití algoritmu vždy. Paměťová složitost je pak $O(np)$, jelikož není třeba počítat kovarianční matici.

³Jako kritérium konvergence používáme defaultní nastavení Zoua a kol. (2006): iterační postup je zastaven, pokud v poslední iteraci nedošlo v žádné složce matice \mathbf{B} ke změně větší než 10^{-3} .

3.4.4 Adaptivní modifikace (ADA-S-PCA)

V regresním tvaru AHK diskutovaném v sekci 2.6 se samozřejmě nabízí použití i jiných penalizačních metod než ℓ_1 -penalizace. Můžeme použít jak libovolnou penalizaci diskutovanou v kapitole 2, tak mnoho jiných penalizačních technik, které jsme nezmínili. V této sekci si představíme verzi, která vznikne nahrazením ℓ_1 -penalizace za adaptivní ℓ_1 -penalizaci. Tato modifikace (kterou označujeme dále ADA-S-PCA) je motivována úspěchem adaptivní lasso metody a byla již nejednou uvažována (např. Lengem a Wangem, 2009, nebo Chenem, 2011.).

ADA-S-PCA spočívá v následující úpravě S-PCA. Na začátku l -tého kroku spočítá váhy

$$\text{Let } w_{ji} = \frac{1}{a_{ij}^{(l)}}, \quad j = 1, \dots, K, \quad i = 1, \dots, p, \quad (3.6)$$

kde $a_{ij}^{(l)}$ jsou prvky matice $\mathbf{A}^{(l)}$ známé z předchozího kroku. Následně místo úloh elastické sítě (3.5) řeš úlohy adaptivní elastické sítě ve tvaru

$$\arg \min_{\mathbf{b}_j \in \mathbb{R}^p} \|\mathbf{X}\mathbf{a}_j - \mathbf{X}\mathbf{b}_j\|_2^2 + \lambda_2 \|\mathbf{b}_j\|_2^2 + \lambda_{1,j} \sum_{i=1}^p w_{ji} |b_{j,i}|, \quad j = 1, \dots, K. \quad (3.7)$$

Změny oproti algoritmu 3.4.2 jsou tedy dvě. Zaprvé, mezi 4. a 5. řádek algoritmu 3.4.2 přibude výpočet vah. A zadruhé, v 5. řádku nahradíme referenci (3.5) za (3.7).

Poznamenejme, že jsme se zabývali i různými vícekrokovými modifikacemi elastické sítě. Nepodařilo se nám však najít žádnou strategii volby vah a ladících parametrů, která by vedla ke znatelnému zlepšení numerických výsledků oproti ADA-S-PCA.

Stejně jako adaptivní lasso je jednoduchou modifikací lasso metody vylepšující jeho teoretické i numerické vlastnosti, ADA-S-PCA je jednoduchou modifikací S-PCA algoritmu vylepšující jeho (při absenci teoretických alespoň) numerické vlastnosti, jak uvidíme v kapitole 5. Přitom časová i paměťová složitost algoritmu zůstává stejná jako pro S-PCA, jelikož v každé iteraci je proveden pouze jeden singulární rozklad matice $n \times p$ navíc. Na druhou stranu, jak uvidíme v sekci 5.1, vlastnost (e) z poznámky 3.5 přestává platit pro ADA-S-PCA, což znamená, že ADA-S-PCA musí být sub-optimální z hlediska množství vysvětleného rozptylu, jsou-li požadované hodnoty kardinalit s_1, \dots, s_K vysoké.

3.5 Shrnutí

V této kapitole jsme naznačili, proč může řídkost sloužit jako náprava ke znovuzískání „ztracené konzistence“ AHK v mnohorozměrné situaci $p > n$. Přestože teoretické výsledky týkající se konzistence se značně liší v závislosti na uvažovaném modelu a algoritmu, AHŘK je dnes obecně považována za vhodnější pro mnohorozměrná data než obyčejná AHK.

I když odhlédneme od teoretických výsledků, nalezení řídkých nábojů je žádaný cíl sám o sobě kvůli lepší interpretovatelnosti výsledků, kterou řídkost přináší. Hledání řídkých nábojů je však NP-těžký problém, pro jehož řešení byly navrženy desítky různých algoritmů. Některé z těchto algoritmů jsou teoreticky podložené, ale nevhodné pro praktické použití. Jiné jsou z praktického hlediska snadno využitelné, ale neexistují pro ně žádné teoretické výsledky zaručující kvalitu nalezených řešení, a tak jsou brány spíše jako heuristiky.

My jsme k problému přistoupili z čistě praktického úhlu pohledu a zaměřili jsme se na S-PCA algoritmus Zoua a kol. (2006), který má mnoho z praktického hlediska atraktivních vlastností. Na druhou stranu, algoritmus se snaží řešit nekonvexní problém pomocí schématu alternující minimalizace a my si nejsme vědomi, že by pro něj existoval byť jen důkaz numerické konvergence, natož jiné teoretické výsledky o jeho chování. Přesto se chování S-PCA algoritmus stalo standardem, se kterým je téměř každý novější algoritmus pro AHK porovnáván.

Numerické fungování S-PCA algoritmu bylo z hlediska různých měř, které budou diskutovány v naší v kapitole 5, překonáno v četných studiích. Pokud je nám známo, vždy bylo lepšího fungování nového algoritmu dosaženo při obětování některé z vlastností z poznámky 3.5. Příkladem tohoto je i jednoduchá modifikace ADA-S-PCA. V následující kapitole je proto naším cílem je navrhnout algoritmus, který by překonával numerické fungování S-PCA a zároveň si uchovával všechny atraktivní vlastnosti z poznámky 3.5.

4. Nový algoritmus pro analýzu řídkých hlavních komponent

V této kapitole si představíme modifikaci S-PCA algoritmu pro AŘHK založenou na zavedení nekonvexní penalizace do regresní formulace AHK, kterou jsme představili v sekci 2.6.

Nejprve budeme studovat problém nejmenších čtverců penalizovaný diferencovatelnou nekonvexní funkcí a navrhneme iterativní algoritmus pro nalezení příslušného odhadu. Náš algoritmus bude verzí iterativně vážených nejmenších čtverců (*iteratively reweighted least squares*, IRLS). Poté použijeme naši penalizaci a navrhneme algoritmus k modifikaci S-PCA algoritmu z minulé kapitoly. Nakonec uvedeme důkazy konvergence jak pro naši modifikaci, tak pro původní verzi S-PCA algoritmu.

4.1 Nekonvexní penalizace nejmenších čtverců

Vraťme se zpět k penalizované úloze nejmenších čtverců

$$\hat{\boldsymbol{\beta}} = \arg \min_{\boldsymbol{\beta} \in \mathbb{R}^p} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \lambda\rho(\boldsymbol{\beta}), \quad (4.1)$$

kde $\lambda > 0$ je ladící parametr, $\mathbf{y} \in \mathbb{R}^n$, $\mathbf{X} \in \mathbb{R}^{n \times p}$ a $\rho : \mathbb{R}^p \rightarrow \mathbb{R}$ je penalizační funkce. Uvažujme penalizační funkci ve tvaru

$$\rho_{\log}(\boldsymbol{\beta}|\delta) = \sum_{j=1}^p \log(\beta_j^2 + \delta), \quad (4.2)$$

kde $\delta > 0$ je regularizační parametr. Motivací pro tuto volbu penalizační funkce je geometrie ρ_{\log} , neboť s vhodně zvoleným δ dokáže tato penalizační funkce aproximovat ℓ_0 -penalizaci s libovolnou přesností, viz obrázek 4.1 a příklad 4.1, a je přitom spojitě diferencovatelná v regresních parametrech $\boldsymbol{\beta}$. Další motivací je iterační algoritmus, který se přirozeně nabízí k minimalizaci úlohy (4.1) s touto volbou penalizační funkce.

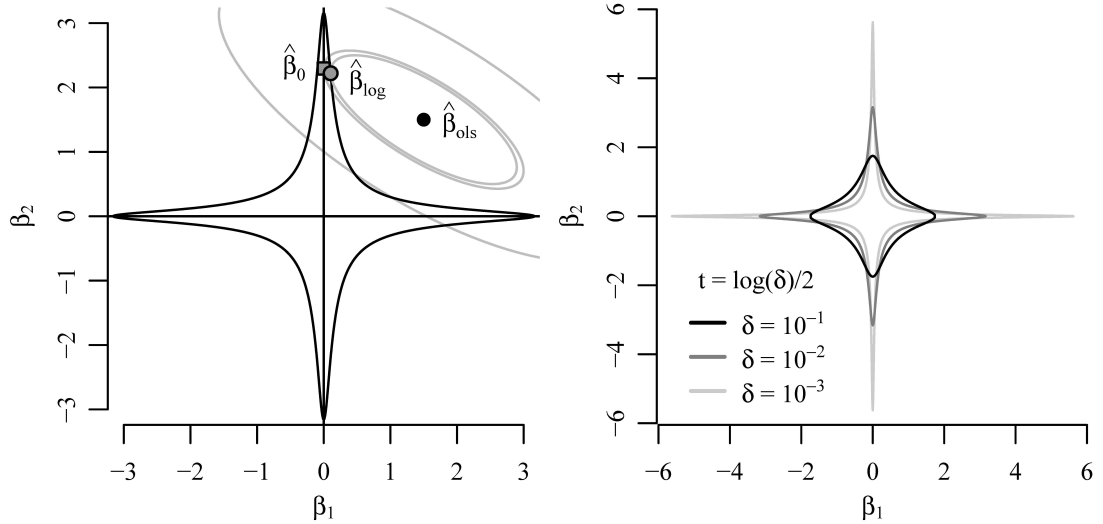
Příklad 4.1. Ilustrujme penalizační funkci ρ_{\log} na příkladu s dvěma regresory, který jsme používali v kapitole 2 k objasnění rozdílů mezi penalizacemi (viz příklad 2.11). Na obrázku 4.1 vlevo je znázorněno řešení omezené úlohy

$$\hat{\boldsymbol{\beta}}_{\log} = \arg \min_{\boldsymbol{\beta} \in \mathbb{R}^p} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 \quad \text{s.t.} \quad \rho_{\log}(\boldsymbol{\beta}|\delta) \leq t,$$

pro volbu $\delta = 10^{-2}$ a $t = \log(\delta)/2$. Žádáme čtenáře o porovnání s obrázky 2.3 a 2.4. Jak je vidět na obrázku 4.1 vpravo, penalizace ρ_{\log} dokáže pro vhodnou volbu parametrů aproximovat ℓ_0 -normu s libovolnou přesností. \triangle

Poznámka 4.2. Všimněme si, že pro $\delta_1 > \delta_2$ platí

$$\left\{ \boldsymbol{\beta} \in \mathbb{R}^p; \sum_{j=1}^p \log(\beta_j^2 + \delta_1) \leq t \right\} \subset \left\{ \boldsymbol{\beta} \in \mathbb{R}^p; \sum_{j=1}^p \log(\beta_j^2 + \delta_2) \leq t \right\}$$



Obrázek 4.1: Vlevo: Ilustrace k příkladu 4.1 se dvěma regresory, kde $\delta = 10^{-2}$ a $t = \log(\delta)/2$. Vpravo: Hranice množin $\{\rho_{\log}(\beta) \leq t\}$ pro $\delta = 10^{-1}, 10^{-2}, 10^{-3}$ a $t = \log(\delta)/2$.

pro libovolné $t > 0$ a naopak pro $t_1 < t_2$ je zřejmě

$$\left\{ \beta \in \mathbb{R}^p; \sum_{j=1}^p \log(\beta_j^2 + \delta) \leq t_1 \right\} \subset \left\{ \beta \in \mathbb{R}^p; \sum_{j=1}^p \log(\beta_j^2 + \delta) \leq t_2 \right\}$$

pro libovolné $\delta > 0$. Tudíž, abychom dobře aproximovali ℓ_0 -normu, je třeba vhodně zvolit oba parametry δ a t (případně δ a λ v neomezené verzi úlohy) a brát přitom v úvahu jejich vzájemné působení, viz obrázek 4.1 vpravo. Opět žádáme čtenáře o porovnání s penalizací pomocí ℓ_q -normy, tedy s obrázkem 2.2. Ještě poznamenejme, že na obrázku 2.2 se snažíme aproximovat ℓ_0 -normu způsobem, který zahrnuje volbu dvou parametrů: t a q . Naše logaritmičká penalizace ρ_{\log} se pokouší o to samé s pomocí taktéž dvou parametrů: t a δ . Rozdíl mezi logaritmičkou penalizací a ℓ_q -penalizací je ten, že logaritmičká penalizace je spojitě diferencovatelná. Algoritmy zaměřené na ℓ_q -minimalizaci obvykle zavádějí další regularizační parametr zajišťující spojitou diferencovatelnost. Pak jsou ale v úloze přítomny tři parametry namísto dvou. \diamond

Vraťme se nyní zpět k úloze (4.1) s logaritmičkou penalizací (4.2). Penalizovaný problém

$$\hat{\beta}_{\log} = \arg \min_{\beta \in \mathbb{R}^p} \|\mathbf{Y} - \mathbf{X}\beta\|_2^2 + \lambda \sum_{j=1}^p \log(\beta_j^2 + \delta). \quad (4.3)$$

je stále nekonvexní a tudíž jej nelze řešit jednoduše. Označme f účelovou funkci předchozí optimalizační úlohy, tj.

$$f(\beta|\delta) = \|\mathbf{y} - \mathbf{X}\beta\|_2^2 + \lambda \sum_{j=1}^p \log(\beta_j^2 + \delta). \quad (4.4)$$

Zavedme dále pomocnou funkci $g : \mathbb{R}^p \times (0, \infty)^p \rightarrow \mathbb{R}$ jako

$$g(\beta, \mathbf{w}|\delta) = \|\mathbf{y} - \mathbf{X}\beta\|_2^2 + \lambda \sum_{j=1}^p w_j \beta_j^2 + \lambda \sum_{j=1}^p [w_j \delta - \log(w_j) - 1], \quad (4.5)$$

kde $w_j > 0$ pro $j = 1, \dots, p$ jsou uvolňující proměnné. Funkce g sice není konvexní, ale je-li jedna z dvojice vektorových proměnných $\boldsymbol{\beta}$ a \mathbf{w} fixní, je g konvexní v druhé z těchto proměnné. Navrhujeme proto minimalizovat g pomocí alternujícího schématu, kdy jedna z dvojice vektorových proměnných $\boldsymbol{\beta}$ a \mathbf{w} je fixována a minimalizace probíhá přes nefixovanou proměnnou. Tedy při inicializaci s počátečním řešením $\mathbf{w}^{(0)}$ (například $w_j^{(0)} = 1, j = 1, \dots, p$) pro $l = 0, 1, \dots$ postupně řešme úlohy

$$\boldsymbol{\beta}^{(l+1)} = \arg \min_{\boldsymbol{\beta} \in \mathbb{R}^p} g(\boldsymbol{\beta}, \mathbf{w}^{(l)} | \delta), \quad (4.6)$$

$$\mathbf{w}^{(l+1)} = \arg \min_{\mathbf{w} \in \mathbb{R}^p} g(\boldsymbol{\beta}^{(l+1)}, \mathbf{w} | \delta). \quad (4.7)$$

Následující lemma shrnuje vlastnosti tohoto alternujícího schématu.

Lemma 4.3. Pro funkci f definovanou v (4.4) a pomocnou funkci g definovanou v (4.5) platí následující:

(a) $\forall \delta > 0, \forall \boldsymbol{\beta} \in \mathbb{R}^p :$

$$f(\boldsymbol{\beta} | \delta) = \min_{\mathbf{w} \in \mathbb{R}^p} g(\boldsymbol{\beta}, \mathbf{w} | \delta). \quad (4.8)$$

(b) $\forall l \in \mathbb{N}, \forall \delta > 0 :$ $\boldsymbol{\beta}^{(l+1)}$ z rovnosti (4.6) je jednoznačné řešení úlohy vážené hřebenové regrese

$$\arg \min_{\boldsymbol{\beta} \in \mathbb{R}^p} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \lambda \sum_{j=1}^p w_j^{(l)} \beta_j^2. \quad (4.9)$$

(c) $\forall l \in \mathbb{N}, \forall \delta > 0 :$ $\mathbf{w}^{(l+1)}$ z rovnosti (4.7) je dáno předpisem

$$w_j^{(l+1)} = \frac{1}{(\beta_j^{(l+1)})^2 + \delta}, \quad j = 1, \dots, p. \quad (4.10)$$

Důkaz je jednoduchý, vyžaduje pouze spočtení derivací funkce g .

Důkaz. Buď $j \in [p]$. Pro $\boldsymbol{\beta} \in \mathbb{R}^p$ je úloha (4.8) konvexní. Položením derivace

$$\frac{\partial}{\partial w_j} g(\boldsymbol{\beta}, \mathbf{w} | \delta) = \lambda \left[\beta_j^2 + \delta - \frac{1}{w_j} \right] \quad (4.11)$$

rovné nule dostáváme jednoznačné řešení

$$\hat{w}_j = \frac{1}{\beta_j^2 + \delta}. \quad (4.12)$$

Dosazením zpět do účelové funkce (4.8) dostáváme

$$\begin{aligned} \min_{\mathbf{w} \in \mathbb{R}^p} g(\boldsymbol{\beta}, \mathbf{w} | \delta) &= g(\boldsymbol{\beta}, \hat{\mathbf{w}} | \delta) \\ &= \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \lambda \sum_{j=1}^p \frac{\beta_j^2}{\beta_j^2 + \delta} + \lambda \sum_{j=1}^p \left[\frac{\delta}{\beta_j^2 + \delta} - \log \left(\frac{1}{\beta_j^2 + \delta} \right) - 1 \right] \\ &= \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \lambda \sum_{j=1}^p \frac{\beta_j^2}{\beta_j^2 + \delta} - \lambda \sum_{j=1}^p \left[\frac{\beta_j^2}{\beta_j^2 + \delta} + \log(\beta_j^2 + \delta) \right] \\ &= f(\boldsymbol{\beta} | \delta). \end{aligned}$$

Tím je ukázán bod (a).

Platnost bodu (b) je zřejmá. Při pevném $\mathbf{w} = \mathbf{w}^{(l)}$ pro nějaké $l \in \mathbb{N}$ není třeba při minimalizaci uvažovat poslední sčítanec funkce g z (4.5).

Bod (c) již byl ukázán odvozením rovnosti (4.12). \square

Poznámka 4.4. Úlohu (4.9) lze řešit jako obyčejnou úlohu hřebenové regrese po jednoduché transformaci dat, podobně jako v poznámce 2.10 v případě adaptivní lasso metody. Příslušná transformace dat nyní je

$$\begin{aligned} b_i &= \sqrt{w_i^{(l)}} \beta_i, \quad i = 1, \dots, p, \\ \tilde{\mathbf{x}}_i &= \frac{1}{\sqrt{w_i^{(l)}}} \mathbf{x}_i, \quad i = 1, \dots, p. \end{aligned}$$

Je-li $\hat{\mathbf{b}}$ řešením obyčejné úlohy hřebenové regrese

$$\arg \min_{\mathbf{b} \in \mathbb{R}^p} \|\mathbf{y} - \tilde{\mathbf{X}}\mathbf{b}\|_2^2 + \lambda \|\mathbf{b}\|_2^2,$$

pak řešení úlohy (4.9) je dáno jako $\beta_i^{(l+1)} = \frac{1}{\sqrt{w_i^{(l)}}} \hat{b}_i$. \diamond

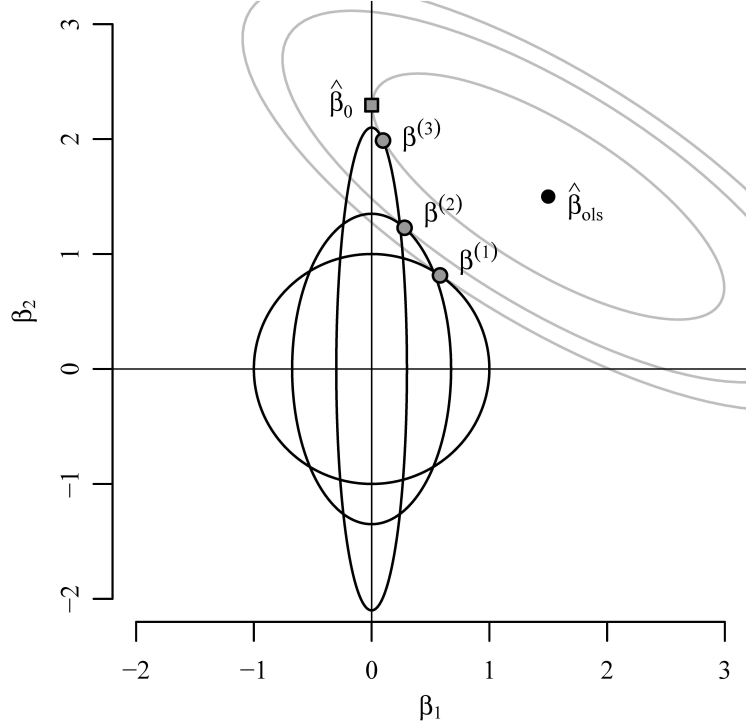
Příklad 4.5. Postup iteračního schématu (4.6)-(4.7) na příkladu s dvěma regresory je zachycen¹ na obrázku 4.2. Začneme-li s počátečními váhami $w_j^{(0)} = 1$, $j = 1, \dots, p$, je $\beta^{(1)}$ dáno řešením standardní hřebenové regrese. Následně dojde ke spočtení vah $\mathbf{w}^{(1)}$ pomocí vzorce (4.10). V druhém kroku je již $\beta^{(2)}$ dáno jako řešení vážené hřebenové regrese, a to samé platí pro všechna další $\beta^{(l)}$, $l = 3, \dots$. V tomto jednoduchém případě s dvěma regresory tak vlastně již v prvním kroku dojde k rozhodnutí, která z proměnných je důležitější, a toto rozhodnutí už zůstane nezměněno až do konce. \triangle

Ve složitějších případech s více korelovanými regresory dochází běžně k tomu, že důležitost koeficientů (z hlediska velikosti absolutní hodnoty odhadnutých koeficientů) se během iterací různě mění. Naopak, jsou-li regresory nekorelované, nemůže tento jev nastat a iterační schéma (4.6)-(4.7) pak odpovídá jisté formě prahování, to zde však nebudeme rozvádět.

Iterační schéma (4.6)-(4.7) se snaží řešit NP-těžký problém výběru nejlepší podmnožiny (úloha (4.1) s $\rho(\cdot) := \|\cdot\|_0$) pomocí úloh vážené hřebenové regrese. Jedná se tedy o algoritmus vážených nejmenších čtverců (IRLS), který se de-facto snaží aproximovat ℓ_0 -normu převažováním ℓ_2 -normy.

Lemma 4.3 (a) říká, že pokud se nám podaří minimalizovat funkci g , minimalizujeme tím i funkci f . Z lemmatu 4.3 (b) a (c) plyne, že alternující minimalizační schéma (4.6)-(4.7) sestává pouze z řešení úloh vážené hřebenové regrese a jednoduchého přepočítávání vah. Toto alternující schéma samozřejmě nezaručuje nalezení skutečného minima funkce g , jelikož g není konvexní v obou proměnných současně. Následující lemma říká, že iterační postup (4.6)-(4.7) je globálně konvergentní.

¹Ne však zcela přesně. Pro přehlednost není na obrázku 4.2 zachycen skutečný vývoj posloupnosti řešení, váhy jsou ručně změněny.



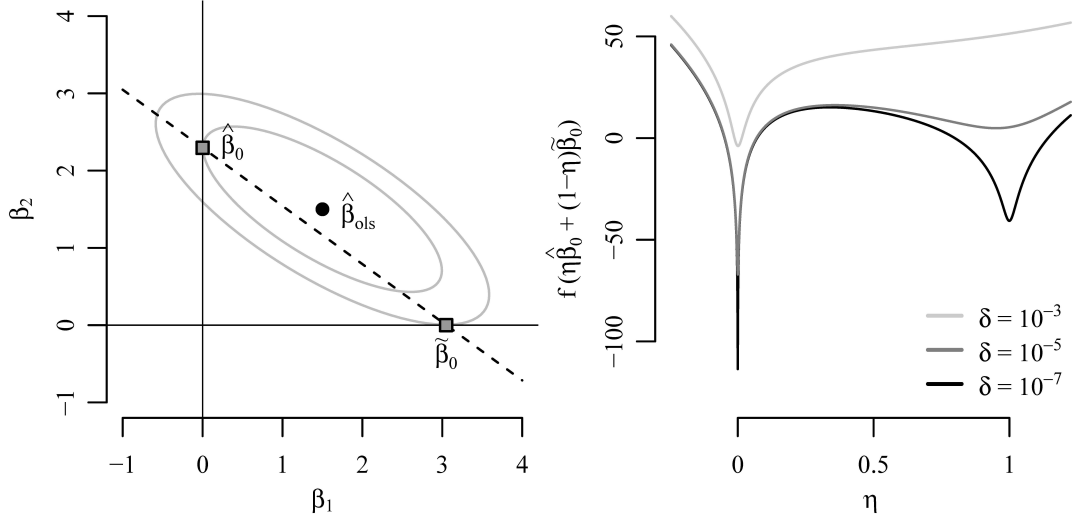
Obrázek 4.2: Ilustrace iteračního schématu (4.6)-(4.7) na jednoduchém příkladu s dvěma korelovanými regresory.

Lemma 4.6. Necht $w_j^{(0)} > 0$, $j = 1, \dots, p$, necht $\delta > 0$ and necht $\{\beta^{(l)}\}_{l=1}^{\infty}$ je posloupnost řešení spočtená postupem, který alternuje mezi řešením (4.6) a (4.7) pro $l \in \mathbb{N}$. Potom platí $\|\beta^{(l+1)} - \beta^{(l)}\|_2 \rightarrow 0$ pro $l \rightarrow \infty$ a každý limitní bod posloupnosti $\{\beta^{(l)}\}_{l=1}^{\infty}$ je stacionární bod funkce $f(\beta|\delta)$.

Lemma 4.6 lze dokázat poměrně jednoduše aplikováním obecné teorie takzvaných MM algoritmů (Lange, 2000). Z bodu (a) lemmatu 4.3 totiž plyne, že iterační schéma (4.6)-(4.7) je příkladem MM algoritmu. My zde však nebudeme zabíhat do podrobností ohledně MM algoritmů. Lemma 4.6 je pouze verzí obecnějšího tvrzení, které bude dokázáno později a jehož důkaz bude soběstačný, tj. nebude třeba žádné speciální teorie.

Z poznámky 4.2 plyne, že regularizační parametr δ bychom chtěli volit co nejmenší, aby penalizační funkce ρ_{\log} dobře aproximovala ℓ_0 -penalizaci. Na druhou stranu, čím je δ menší, tím je iterační algoritmus méně stabilní. To je způsobeno zvyšující se nekonvexitou penalizační funkce, viz obrázek 4.1 vpravo. Četné numerické studie (např. Daubechies a kol., 2010) proto doporučují, aby byl parametr δ variabilní, tj. aby byl zvolen relativně vysoký pro počáteční iterace, a následně snižován směrem k požadované hodnotě. Důvodem je, že pro velká δ je penalizace a tím i účelová funkce konvexní, což dává algoritmu šanci vyhnout se lokálním minimům blízko inicializace a nalézt kvalitnější řešení úlohy. Tuto funkci regularizačního parametru opět ilustrujeme na našem příkladu s dvěma regresory.

Příklad 4.7. Uvažujme opět ilustrační příklad s dvěma regresory. Necht je $\hat{\beta}_0$ znovu řešení úlohy výběru nejlepší podmnožiny velikosti 1 a $\hat{\beta}_0$ necht nyní označuje odhad při výběru opačného regresoru, viz obrázek 4.3 vlevo. Uvědomme si, že



Obrázek 4.3: Ilustrace k příkladu s dvěma regresory. *Vlevo:* Řešení jednoduchého lineárního modelu přímkou s pouze prvním (řešení $\tilde{\beta}_0$) respektive druhým (řešení $\hat{\beta}_0$) regresorem. Přerušovaná přímka je parametrizována pomocí $\eta \in \mathbb{R}$ implicitním předpisem $\eta\hat{\beta}_0 + (1-\eta)\tilde{\beta}_0 = 0$. *Vpravo:* Na této přímce je vykreslena účelová funkce úlohy (4.3) pro $\lambda = 10$ pro různé hodnoty δ .

$\hat{\beta}_0$ a $\tilde{\beta}_0$ jsou lokální řešení úlohy (4.1) s $\rho(\cdot) := \|\cdot\|_0$ a libovolným $\lambda > 0$; podobně jsou to lokální minima úlohy (4.3) pro libovolné $\lambda > 0$ a dostatečně malé $\delta > 0$. Pro ukázkou vykreslujeme na obrázku 4.3 vpravo hodnoty účelové funkce úlohy (4.3) na přímce procházející body $\hat{\beta}_0$ a $\tilde{\beta}_0$ pro různé hodnoty δ . Zatímco lokální minimum v $\tilde{\beta}_0$ je vyhlazeno pro dostatečně velká δ , pro dostatečně malá δ by mohl iterační algoritmus uvíznout v tomto sub-optimálním řešení při nevhodné volbě inicializace. \triangle

Zahrňme tedy do alternujícího schématu (4.6)-(4.7) změnu regularizačního parametru δ mezi iteracemi. Dostáváme algoritmus 4.1.1.

Body (b) a (c) lemmatu 4.3 stále vyjadřují, jak vypadají řešení úloh (4.13) a (4.14). Jediným rozdílem proti předchozímu alternujícímu schématu je, že δ v (4.10) je nyní nahrazeno variabilním parametrem δ_{l+1} . Naopak lemma 4.3 (a) se k algoritmu 4.1.1 nevztahuje. Apriori totiž není jasné, jakou funkci se algoritmus 4.1.1 snaží minimalizovat – pokud vůbec nějakou. Přesto stále platí globální konvergence ke stacionárnímu bodu limitní funkce $f(\beta|\delta_*)$, kde δ_* je nyní limitním bodem posloupnosti $\{\delta_l\}_{l=1}^\infty$.

Tvrzení 4.8. Nechť $\mathbf{X} \in \mathbb{R}^{n \times p}$ je matice dat, $\mathbf{y} \in \mathbb{R}^n$ je odezva a $\lambda > 0$ je ladící parametr. Nechť $w_j^{(0)} > 0$, $j = 1, \dots, p$, nechť $\delta_0 \geq \delta_1 \geq \dots$ je posloupnost regularizačních parametrů taková, že $\delta_l \rightarrow \delta_* > 0$, a nechť $\{\beta^{(l)}\}_{l=1}^\infty$ je posloupnost odhadů generovaná algoritmem 4.1.1. Potom $\|\beta^{(l+1)} - \beta^{(l)}\|_2 \rightarrow 0$ pro $l \rightarrow \infty$ a každý hromadný bod posloupnosti $\{\beta^{(l)}\}_{l=1}^\infty$ je stacionárním bodem $f(\beta|\delta_*)$.

Důkaz odkládáme do sekce 4.4.1.

Algoritmus 4.1.1: Iterativní algoritmus pro úlohu (4.3) s proměnlivým δ .

Input : Váhy $w_j^{(0)} \geq 0$ (defaultně $w_j^{(0)} = 1$), $j = 1, \dots, p$, a přesnost $\xi_{\text{conv}} > 0$.

Posloupnost $\{\delta_l\}_{l=0}^\infty$ taková, že $\delta_0 \geq \delta_1 \geq \dots$ a $\delta_l \rightarrow \delta_\star > 0$ (defaultně

$\delta_0 = 1$ a $\delta_{l+1} = \max(\delta_l/1.5, \delta_{\min})$, kde $\delta_{\min} = 10^{-9}$).

Output: Odhad $\hat{\beta}$ a index \hat{l} posledního použitého regularizačního parametru.

begin

 Polož $l = 0$.

repeat

$$\beta^{(l+1)} = \arg \min_{\beta \in \mathbb{R}^p} g(\beta, \mathbf{w}^{(l)} | \delta_l) \quad (4.13)$$

$$\mathbf{w}^{(l+1)} = \arg \min_{\mathbf{w} \in \mathbb{R}^p} g(\beta^{(l+1)}, \mathbf{w} | \delta_{l+1}) \quad (4.14)$$

 Polož $l = l + 1$.

until $\forall i \in [p] : |\beta_i^{(l)} - \beta_i^{(l-1)}| < \xi_{\text{conv}}$

 Polož $\hat{\beta} = \beta^{(l)}$ a $\hat{l} = l - 1$.

end

4.2 Nová modifikace regresního algoritmu (IRLS-S-PCA)

Nyní se nabízí jednoduchá modifikace S-PCA algoritmu pro AŘHK ze sekce 3.4.3. Namísto úloh elastické sítě bude IRLS-S-PCA algoritmus řešit úlohy typu (4.3) pomocí algoritmu 4.1.1 z minulé sekce. Postup je shrnut jako algoritmus 4.2.1.

Všimněme si, že samotný S-PCA algoritmus sice zavádí konvexní penalizaci, ale v rámci vnějšího nekonvexního schématu. Lasso je v regresním kontextu velmi populární nejen díky jeho výpočetní nenáročnosti a schopnosti provádět výběr důležitých proměnných spolu s jejich odhadem. Existuje nepřeberné množství metod, které dosahují těchto cílů. Pro žádnou z těchto metod však neexistuje tak obsáhlá literatura zkoumající jejich chování z teoretického úhlu pohledu, jako je tomu v případě lasso metody (viz např. Bühlmann a Van de Geer, 2011, pro detailní diskuzi). Jakékoliv teoretické výsledky o chování S-PCA jsou však těžko dosažitelné (a nám neznámé) kvůli vnějšímu alternujícímu schématu. Použitím nekonvexní penalizace uvnitř tohoto schématu tedy neztrácíme žádné vlastnosti.

Konvergence algoritmu bude ukázána v sekci 4.4.3, praktická implementace diskutována v sekci 4.5 a numerické chování algoritmu bude zkoumáno v páté kapitole.

Algoritmus 4.2.1: IRLS-S-PCA algoritmus.

Input : Centovaná matice $\mathbf{X} \in \mathbb{R}^{n \times p}$, hledaný počet řídkých nábojů $K \in [p]$.
Ladící parametry $\lambda_{1,1}, \dots, \lambda_{1,K}$.
Požadovaná přesnost $\xi_{\text{inner}} > 0$ (defaultně 10^{-6})
a $\xi_{\text{outer}} > 0$ (defaultně 10^{-3}).
Posloupnost $\{\delta_l\}_{l=0}^\infty$ taková, že $\delta_0 \geq \delta_1 \geq \dots$ a $\delta_l \rightarrow \delta_\star > 0$
(defaultně $\delta_0 = 1$ a $\delta_{l+1} = \max(\delta_l/1.5, \delta_{\min})$, kde $\delta_{\min} = 10^{-9}$).

Output: Řídké náboje ve sloupcích matice $\mathbf{B} \in \mathbb{R}^{p \times K}$.

begin

 Polož $l = 0$ a $L = 0$.

 Polož $\mathbf{A}^{(0)} = \widehat{\mathbf{A}}$, kde $\widehat{\mathbf{A}} \in \mathbb{R}^{p \times K}$ je matice prvních K obyčejných nábojů.

repeat

for $j \in \{1, \dots, K\}$ **do**

 Získej vektor $\mathbf{b}_j^{(l+1)}$ a index L_j jako výstup algoritmu 4.1.1 s
 vstupy $\mathbf{y} = \mathbf{X}\mathbf{a}_j$, $\{\delta_l\}_{l=0}^\infty = \{\delta_l\}_{l=L}$, $\xi_{\text{conv}} = \xi_{\text{inner}}$ a $w_i^{(0)} = 1$,
 $i \in 1, \dots, p$.

 Znormuj $\mathbf{b}_j^{(l+1)}$.

end

 Polož $\mathbf{B}^{(l+1)} = (\mathbf{b}_1^{(l+1)} | \dots | \mathbf{b}_K^{(l+1)})$.

 Polož $l = \min(L_j; j \in [K])$.

 Spočti singulární rozklad $\mathbf{X}^\top \mathbf{X} \mathbf{B}^{(l+1)} = \mathbf{U} \mathbf{D} \mathbf{V}^\top$ a polož

$\mathbf{A}^{(l+1)} = \mathbf{U} \mathbf{V}^\top$.

 Polož $l = l + 1$.

until $\forall i \in [p], j \in [K] : |b_{ij}^{(l)} - b_{ij}^{(l-1)}| < \xi_{\text{outer}}$.

end

4.3 Spojitost s existující literaturou

Ještě než přejdeme k důkazům konvergence regresních algoritmů pro AŘHK, budeme v této sekci diskutovat spojitosti a rozdílnosti našeho přístupu s již existující literaturou. Nejprve diskutujeme penalizaci nejmenších čtverců.

Fan a Li (2001) se zabývali hledáním metody, která by byla asymptoticky stejně efektivní jako věšebná metoda, jíž by byla dopředu známa skutečná opora hledaného vektoru. Autoři se domnívali, že velmi populární lasso není tak efektivní jako věšebná metoda. Navrhli proto pracovat s konkávními penalizačními funkcemi a výsledný nekonvexní problém řešit iterativně pomocí postupu založeném na lokálně kvadratických aproximacích. Náš iterační postup (4.6)-(4.7) s fixním regularizačním parametrem δ lze chápat jako algoritmus lokálně kvadratických aproximací Fana a Liho (2001), avšak použitý pro jinou penalizační funkci. Zde poznamenejme, že Fan a Li ukazují konvergenci svého algoritmu pouze experimentálně.

Jako jistou formu odpovědi na Fana a Liho (2001), Zou (2006) navrhl adaptivní lasso (viz sekce 2.3), které disponuje všestebnými vlastnostmi požadovanými Fanem a Lim (2001), a zároveň se jedná o pouze dvoukrokový odhad řešící konvexní optimalizační problém v každém kroku. Zdánlivá disproporce mezi Fanem a Lim (2001) a Zouem (2006) byla objasněna v článcích Huntera a Liho (2005) a Zoua a Liho (2008). Hunter a Li (2005) ukázali, že lokálně kvadratické aproximace Fana a Liho (2001) jsou ve skutečnosti instancí MM algoritmu. Také navrhli jako první použít malý fixní regularizační parametr δ , motivováni tím, aby algoritmus nesdílel nevýhodu postupného vyřazování proměnných z modelu (tzv. *backward variable selection*), tj. aby proměnná odhadnutá nulou v jistém kroku nemusela být nutně nulová ve všech následujících krocích. Zou a Li (2008) potom navrhli použít namísto lokálně kvadratických aproximací lokálně lineární aproximace vhodné penalizační funkce, které vedou k iterativnímu převažování ℓ_1 -penalizace. Motivaci autorů lze jednoduše shrnout tak, že nejlepší konvexní aproximace konkávní funkce je funkce lineární. Zou a Li (2008) také ukázali, že adaptivní lasso je dvoukrokový odhad v tomto schématu a že tento dvoukrokový odhad může být asymptoticky stejně efektivní jako plně iterované schéma. Tím se autoři dostali zpět k adaptivní lasso metodě.

Poznamenejme, že teoretické výsledky v článcích citovaných v předchozích dvou odstavcích se týkají buď asymptotického případu s fixním p a $n \rightarrow \infty$ nebo případu sdružené limity $p/n \rightarrow \tau > 0$, nicméně pro $\tau \leq 1$, tj. roste-li počet pozorování rychleji než počet parametrů. V situaci $p > n$ četné empirické studie naznačují, že je vhodné uvažovat plně iterovaný odhad. Například postup Candese a kol. (2008), který je prakticky totožný s lokálně lineárními aproximacemi Zoua a Liho (2008), je v praxi velmi populární. Simulační studie také ukazují, že lokálně kvadratické aproximace vedoucí na algoritmus převážených nejmenších čtverců (IRLS) často překonávají lokálně lineární aproximace (viz Chartrand a Yin, 2008, Zhang a Kingsbury, 2010, nebo Wipf a kol., 2010, a další reference uvnitř). Nejsme si však vědomi žádných teoretických výsledků, které by podporovaly tato pozorování.

Z algoritmického hlediska má náš přístup asi nejbliže (co je nám známo) k Daubechies a kol. (2010) a Lai a kol. (2013), přičemž rozdíly jsou následující. Oba články se soustředí na ℓ_q -penalizaci, nikoliv na naši logaritmickou penalizaci, kterou lze v jistém smyslu chápat jako limitní případ pro $q \rightarrow 0$. Dále Daubechies a kol. (2010) navrhuje použít variabilní regularizační parametr δ , ale soustředí se na omezený problém bez šumu. Naopak Lai a kol. (2013) uvažují sice šum, ale nechávají parametr δ fixní. Fundamentálním rozdílem mezi naším přístupem a přístupem článků citovaných v tomto a předchozím odstavci je, že my na daný problém nahlížíme jako na odhad parametrů ve statistickém modelu, zatímco citované články nahlíží na úlohu jako na inverzní problém ve zpracování signálu.

Za zmínku stojí ještě práce Mazumdera a kol. (2011), kteří se také zabývají iteračními algoritmy pro řešení penalizovaných nejmenších čtverců s nekonvexní penalizací, a pro svůj algoritmus ukazují konvergenci s použitím obdobných metod, jako použijeme my v sekci 4.4. Tito autoři však namísto iterativního převažování penalizační funkce volí přístup iterativního prahování, který je fundamentálně odlišný. Přesto jejich přístup k volbě parametrů a jeho paralelizace (což v naší práci příliš nediskutujeme) bychom mohli pohotově využít pro náš algoritmus.

Z hlediska přístupu k analýze řídkých hlavních komponent je naše práce přirozeně propojena s pracemi Zoua a kol. (2006) a Wittenové a kol. (2009), na něž de-facto navazujeme. Náš přístup k AŘHK z hlediska alternující minimalizace se také shoduje s přístupem Richtárika a kol. (2012). Nicméně Richtárik a kol. se soustředí na algoritmy extrahující jednotlivé náboje sekvenčně, neuvažují tak ve své komparativní studii S-PCA algoritmus.

4.4 Globální konvergence algoritmů

V této sekci ukážeme globální konvergenci uvažovaných algoritmů. Konvergenci ADA-S-PCA algoritmu explicitně neukazujeme, nicméně důkaz by byl velmi obdobný jako pro IRLS-S-PCA algoritmus. Sekce 4.4.1 obsahuje důkaz tvrzení 4.8, který je potřebný k ukázaní konvergence IRLS-S-PCA. V sekcích 4.4.2 a 4.4.3 ukazujeme globální konvergenci S-PCA a IRLS-S-PCA algoritmů; tedy ukážeme, že pro dané algoritmy platí $\|\mathbf{B}^{(l)} - \mathbf{B}^{(l-1)}\|_F \rightarrow 0$ a že zastavovací pravidla použitá v algoritmech 3.4.2 a 4.2.1 jsou řádně podložena.

4.4.1 Nekonvexní penalizace nejmenších čtverců

K samotnému důkazu tvrzení 4.8 budeme potřebovat několik pomocných výsledků. Ty jsou shrnuty v následujících dvou lemmatech.

Lemma 4.9. Necht $\mathbf{X} \in \mathbb{R}^{n \times p}$, $\mathbf{y} \in \mathbb{R}^n$, $\boldsymbol{\beta}, \boldsymbol{\beta}^* \in \mathbb{R}^p$, $\delta > 0$ a necht f a g jsou funkce definované v (4.4) a (4.5). Potom platí následující tvrzení.

(a) Pro $w_j = \frac{1}{\beta_j^2 + \delta}$, $j = 1, \dots, p$ je

$$\nabla_{\boldsymbol{\beta}} f(\boldsymbol{\beta}|\delta) = \nabla_{\boldsymbol{\beta}} g(\boldsymbol{\beta}, \mathbf{w}|\delta).$$

(b) Platí

$$\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 - \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}^*\|_2^2 \geq 2(\mathbf{X}\boldsymbol{\beta}^* - \mathbf{y})^\top (\mathbf{X}\boldsymbol{\beta} - \mathbf{X}\boldsymbol{\beta}^*).$$

(c) Pro $x, y \in \mathbb{R}$ platí

$$\log(x^2 + \delta) - \log(y^2 + \delta) - 2\frac{y(x-y)}{x^2 + \delta} \geq \frac{(x-y)^2}{x^2 + \delta},$$

a rovnost nastává právě tehdy, když $x = y$.

Důkaz. K ukázaní (a) spočtíme gradienty

$$\nabla_{\boldsymbol{\beta}} f(\boldsymbol{\beta}|\delta) = -2\mathbf{X}^\top (\mathbf{y} - \mathbf{X}\boldsymbol{\beta}) + 2\lambda \left[\frac{\beta_j}{\beta_j^2 + \delta} \right]_{j=1}^p \quad (4.15)$$

$$\nabla_{\boldsymbol{\beta}} g(\boldsymbol{\beta}, \mathbf{w}|\delta) = -2\mathbf{X}^\top (\mathbf{y} - \mathbf{X}\boldsymbol{\beta}) + 2\lambda [w_j \beta_j]_{j=1}^p \quad (4.16)$$

a dosadme do $\nabla_{\boldsymbol{\beta}} g(\boldsymbol{\beta}, \mathbf{w}|\delta)$ váhy $w_j = \frac{1}{\beta_j^2 + \delta}$.

K ukázání (b) rozepišme

$$\begin{aligned}\|\mathbf{X}\boldsymbol{\beta} - \mathbf{X}\boldsymbol{\beta}^*\|_2^2 &= \|(\mathbf{X}\boldsymbol{\beta} - \mathbf{y}) - (\mathbf{X}\boldsymbol{\beta}^* - \mathbf{y})\|_2^2 \\ &= \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 - \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}^*\|_2^2 - 2(\mathbf{X}\boldsymbol{\beta} - \mathbf{y})^\top(\mathbf{X}\boldsymbol{\beta}^* - \mathbf{y})\end{aligned}\quad (4.17)$$

a také

$$\begin{aligned}(\mathbf{X}\boldsymbol{\beta}^* - \mathbf{y})^\top(\mathbf{X}\boldsymbol{\beta} - \mathbf{X}\boldsymbol{\beta}^*) &= (\mathbf{X}\boldsymbol{\beta}^* - \mathbf{y})^\top[(\mathbf{X}\boldsymbol{\beta} - \mathbf{y}) - (\mathbf{X}\boldsymbol{\beta}^* - \mathbf{y})] \\ &= (\mathbf{X}\boldsymbol{\beta}^* - \mathbf{y})^\top(\mathbf{X}\boldsymbol{\beta} - \mathbf{y}) - \|\mathbf{X}\boldsymbol{\beta}^* - \mathbf{y}\|_2^2.\end{aligned}\quad (4.18)$$

Sečtením rovnice (4.17) s dvojnásobkem rovnice (4.18) dostáváme

$$\|\mathbf{X}\boldsymbol{\beta} - \mathbf{X}\boldsymbol{\beta}^*\|_2^2 + 2(\mathbf{X}\boldsymbol{\beta}^* - \mathbf{y})^\top(\mathbf{X}\boldsymbol{\beta} - \mathbf{X}\boldsymbol{\beta}^*) = \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 - \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}^*\|_2^2,$$

odkud vypuštěním $\|\mathbf{X}\boldsymbol{\beta} - \mathbf{X}\boldsymbol{\beta}^*\|_2^2 > 0$ dostáváme výsledek.

Ad (c): jelikož $2\frac{y(x-y)}{x^2+\delta} + \frac{(x-y)^2}{x^2+\delta} = \frac{x^2-y^2}{x^2+\delta}$, stačí ukázat, že

$$\log(x^2 + \delta) - \log(y^2 + \delta) - \frac{x^2 - y^2}{x^2 + \delta} \geq 0.$$

Pro pevné $x > 0$ je snadné ověřit derivováním, že funkce

$$h(y) = \log(x^2 + \delta) - \log(y^2 + \delta) - \frac{x^2 - y^2}{x^2 + \delta}$$

má jediné minimum v $y = x$. To samé platí pro $x < 0$ a $x = 0$. \square

Lemma 4.10. Za předpokladů tvrzení 4.8 platí pro $l \in \mathbb{N}$, že

$$(\mathbf{X}\boldsymbol{\beta}^{(l+1)} - \mathbf{y})^\top(\mathbf{X}\boldsymbol{\beta}^{(l)} - \mathbf{X}\boldsymbol{\beta}^{(l+1)}) = -\lambda \sum_{j=1}^p \frac{\beta_j^{(l+1)}(\beta_j^{(l)} - \beta_j^{(l+1)})}{(\beta_j^{(l)})^2 + \delta_l}. \quad (4.19)$$

Důkaz. Z podmínek optimality prvního řádu musí pro parametr $\boldsymbol{\beta}^{(l+1)}$ platit $\nabla_{\boldsymbol{\beta}} g(\boldsymbol{\beta}^{(l+1)}, \mathbf{w}^{(l)}|\delta) = 0$. Odtud dostáváme

$$\begin{aligned}0 &= (\boldsymbol{\beta}^{(l)} - \boldsymbol{\beta}^{(l+1)})^\top \nabla_{\boldsymbol{\beta}} g(\boldsymbol{\beta}^{(l+1)}, \mathbf{w}^{(l)}|\delta) \\ &= -2(\mathbf{X}\boldsymbol{\beta}^{(l)} - \mathbf{X}\boldsymbol{\beta}^{(l+1)})^\top(\mathbf{y} - \mathbf{X}\boldsymbol{\beta}^{(l+1)}) + 2\lambda(\boldsymbol{\beta}^{(l)} - \boldsymbol{\beta}^{(l+1)})^\top [w_j^{(l)}\beta_j]_{j=1}^p \\ &= 2(\mathbf{X}\boldsymbol{\beta}^{(l+1)} - \mathbf{y})^\top(\mathbf{X}\boldsymbol{\beta}^{(l)} - \mathbf{X}\boldsymbol{\beta}^{(l+1)}) + 2\lambda \sum_{j=1}^p \frac{\beta_j^{(l+1)}(\beta_j^{(l)} - \beta_j^{(l+1)})}{(\beta_j^{(l)})^2 + \delta_l},\end{aligned}$$

kde v poslední rovnosti jsme zaměnili pořadí činitelů v prvním členu a dosadili váhy $w_j^{(l)} = \frac{1}{(\beta_j^{(l)})^2 + \delta_l}$, jejichž konkrétní tvar plyne z lemmatu 4.6. \square

Nyní jsme připraveni dokázat tvrzení 4.8.

Důkaz. (Tvzení 4.8.)

Buď $l \in \mathbb{N}$ pevné. Odhadněme

$$\begin{aligned}
& f(\boldsymbol{\beta}^{(l)}|\delta_l) - f(\boldsymbol{\beta}^{(l+1)}|\delta_{l+1}) = \\
& = \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}^{(l)}\|_2^2 - \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}^{(l+1)}\|_2^2 + \\
& \quad + \lambda \sum_{j=1}^p \left[\log \left((\beta_j^{(l)})^2 + \delta_l \right) - \log \left((\beta_j^{(l+1)})^2 + \delta_{l+1} \right) \right] \\
& \geq 2(\mathbf{X}\boldsymbol{\beta}^{(l+1)} - \mathbf{y})^\top (\mathbf{X}\boldsymbol{\beta}^{(l)} - \mathbf{X}\boldsymbol{\beta}^{(l+1)}) + \\
& \quad + \lambda \sum_{j=1}^p \left[\log \left((\beta_j^{(l)})^2 + \delta_l \right) - \log \left((\beta_j^{(l+1)})^2 + \delta_{l+1} \right) \right] \\
& \geq \lambda \sum_{j=1}^p \left[\log \left((\beta_j^{(l)})^2 + \delta_l \right) - \log \left((\beta_j^{(l+1)})^2 + \delta_{l+1} \right) - 2 \frac{\beta_j^{(l+1)}(\beta_j^{(l)} - \beta_j^{(l+1)})}{(\beta_j^{(l)})^2 + \delta_l} \right].
\end{aligned}$$

V první nerovnosti jsme použili lemma 4.9 (b) a v druhé nerovnosti lemma 4.10.

Nyní si všimněme, že δ_{l+1} můžeme v posledním výrazu nahradit za δ_l a nezvýšit tím hodnotu výrazu, jelikož $\delta_l > \delta_{l+1}$. Jakmile máme v celém výrazu stejné δ , můžeme aplikovat lemma 4.9 (c) pro každý člen sumy zvlášť. Dostáváme odhad

$$f(\boldsymbol{\beta}^{(l)}|\delta_l) - f(\boldsymbol{\beta}^{(l+1)}|\delta_{l+1}) \geq \lambda \sum_{j=1}^p \frac{(\beta_j^{(l+1)} - \beta_j^{(l)})^2}{(\beta_j^{(l)})^2 + \delta_l}.$$

Konečně, opět nahradme δ_l za větší δ_0 , čímž získáme

$$f(\boldsymbol{\beta}^{(l)}|\delta_l) - f(\boldsymbol{\beta}^{(l+1)}|\delta_{l+1}) \geq \lambda \sum_{j=1}^p \frac{(\beta_j^{(l+1)} - \beta_j^{(l)})^2}{(\beta_j^{(l)})^2 + \delta_0}. \quad (4.20)$$

Z nerovnosti (4.20) plyne, že algoritmus 4.1.1 produkuje posloupnost monotónně klesajících funkčních hodnot $\{f(\boldsymbol{\beta}^{(l)}|\delta_l)\}_{l=1}^p$. Tedy pro $l \in \mathbb{N}$ platí

$$f(\boldsymbol{\beta}^{(1)}|\delta_1) \geq f(\boldsymbol{\beta}^{(l)}|\delta_l) \geq \lambda \sum_{j=1}^p \log \left((\beta_j^{(l)})^2 + \delta_l \right) \geq \lambda \sum_{j=1}^p \log \left((\beta_j^{(l)})^2 + \delta_\star \right), \quad (4.21)$$

odkud dostáváme stejnoměrnou omezenost regresních koeficientů, tj. existuje konstanta $c > 0$ taková, že pro libovolná $l \in \mathbb{N}$ a $j \in [p]$ platí $|\beta_j^{(l)}| < c$. Pokračováním v odhadech z (4.20) dostáváme

$$f(\boldsymbol{\beta}^{(l)}|\delta_l) - f(\boldsymbol{\beta}^{(l+1)}|\delta_{l+1}) \geq \frac{\lambda}{c^2 + \delta_0} \sum_{j=1}^p (\beta_j^{(l+1)} - \beta_j^{(l)})^2 = \frac{\lambda}{c^2 + \delta_0} \|\boldsymbol{\beta}^{(l+1)} - \boldsymbol{\beta}^{(l)}\|_2^2. \quad (4.22)$$

Z nerovností v (4.21) spolu s již zmíněnou monotonicitou posloupnosti funkčních hodnot $\{f(\boldsymbol{\beta}^{(l)}|\delta_l)\}_{l=1}^\infty$ plyne, že

$$f(\boldsymbol{\beta}^{(l)}|\delta_l) \rightarrow f^\star \in \mathbb{R} \quad \text{pro } l \rightarrow \infty.$$

Tudíž sečtením (4.22) přes $l \in \mathbb{N}$ dostáváme

$$f(\boldsymbol{\beta}^{(1)}|\delta_1) - f^\star \geq \frac{\lambda}{c^2 + \delta_0} \sum_{l=1}^\infty \|\boldsymbol{\beta}^{(l+1)} - \boldsymbol{\beta}^{(l)}\|_2^2.$$

Odtud z Cauchyho podmínky konvergence plyne

$$\|\boldsymbol{\beta}^{(l+1)} - \boldsymbol{\beta}^{(l)}\|_2 \rightarrow 0. \quad (4.23)$$

Dále, pro funkci $f(\boldsymbol{\beta}|\delta)$ platí pro libovolné $\delta > 0$, že $(\boldsymbol{\beta}|\delta) \rightarrow \infty$ pro $\|\boldsymbol{\beta}\|_2 \rightarrow \infty$ (f je tzv. *coercive*) a úrovněvé množiny funkce (*level sets*) $\{\boldsymbol{\beta} \in \mathbb{R}^p; \|\boldsymbol{\beta}\|_2 \leq t\}$, $t \in \mathbb{R}$, jsou kompaktní. Tudíž existuje $\{\boldsymbol{\beta}^{(l_m)}\}_{m=1}^\infty \subset \{\boldsymbol{\beta}^{(l)}\}_{l=1}^\infty$ taková, že platí $\boldsymbol{\beta}^{(l_m)} \rightarrow \boldsymbol{\beta}^* \in \mathbb{R}^p$. Ze spojitosti musí být také

$$w_j^{(l_m)} = \frac{1}{(\beta_j^{(l_m)})^2 + \delta_{l_m}} \rightarrow \frac{1}{(\beta_j^*)^2 + \delta_*} = w_j^*$$

a z (4.23) také $\boldsymbol{\beta}^{(l_m+1)} \rightarrow \boldsymbol{\beta}^*$.

Spočtením limity podmínky optimality prvního řádu

$$\nabla_{\boldsymbol{\beta}} g(\boldsymbol{\beta}^{(l_m+1)}, \mathbf{w}^{(l_m)} | \delta_{l_m}) = 0$$

pro $m \rightarrow \infty$ dostáváme $\nabla_{\boldsymbol{\beta}} g(\boldsymbol{\beta}^*, \mathbf{w}^* | \delta_*) = 0$. Konečně, použitím lemma 4.9 (a) dostáváme $\nabla f(\boldsymbol{\beta}^* | \delta_*) = 0$, čímž je tvrzení dokázáno. \square

Ještě poznamenejme, že důkaz by šel provést též pomocí věty o pevném bodě. Alternativně by se dalo tvrzení dokázat také použitím Zangwillovy věty o globální konvergenci (Zangwill, 1969). Obě tyto verze důkazu by vyžadovaly obdobné kroky jako náš soběstačný důkaz.

4.4.2 S-PCA algoritmus

V této sekci ukážeme, že zastavovací kritérium S-PCA algoritmu 3.4.2 je skutečně splněno, tj. že algoritmus konverguje bez ohledu na volbu inicializace.

Tvrzení 4.11. Necht $\mathbf{X} \in \mathbb{R}^{n \times p}$ je centrovaná matice dat, $\lambda_2 > 0$ je ladící parametr, $K \in [p]$ je počet hledaných řídkých nábojů a $\lambda_{1,1}, \dots, \lambda_{1,K}$ jsou ladící parametry (nebo ekvivalentně s_1, \dots, s_K jsou požadované kardinality nábojů). Necht $\{\mathbf{B}^{(l)}\}_{l=1}^\infty$ je posloupnost matic typu $p \times K$ řídkých nábojů získaná algoritmem 3.4.2. Pak platí

$$\|\mathbf{B}^{(l)} - \mathbf{B}^{(l-1)}\|_F \rightarrow 0 \quad \text{pro } l \rightarrow \infty,$$

tedy algoritmus 3.4.2 je konvergentní.

Ještě před samotným důkazem poznamenejme, že důkaz konvergence je zcela nezávislý na počáteční volbě matice $\mathbf{A}^{(0)}$ v algoritmu 3.4.2. Inicializace pomocí obyčejných nábojů se nabízí velmi přirozeně a my ji bez výhrady užíváme. Pokud bychom však z nějakého důvodu chtěli využít jiné inicializace, na konvergenci algoritmu to nebude mít vliv.

Důkaz. Označme h účelovou funkci úlohy (3.4), tj. $h : \mathbb{R}^{p \times K} \times \mathbb{R}^{p \times K} \rightarrow \mathbb{R}$ s předpisem

$$h(\mathbf{A}, \mathbf{B}) = \sum_{i=1}^n \|\mathbf{x}^{(i)} - \mathbf{A}\mathbf{B}^\top \mathbf{x}^{(i)}\|_2^2 + \lambda_2 \sum_{j=1}^K \|\mathbf{b}_j\|_2^2 + \sum_{j=1}^K \lambda_{1,j} \|\mathbf{b}_j\|_1.$$

S-PCA algoritmus se snaží minimalizovat právě tuto účelovou funkci. Ladící parametry $\lambda_{1,1}, \dots, \lambda_{1,K}$ mohou být zadány implicitně pomocí požadovaných kardinalit s_1, \dots, s_K ve smyslu poznámky 2.13.

Podobně jako v důkazu věty 2.16 lze psát

$$\begin{aligned} h(\mathbf{A}, \mathbf{B}) &= \|\mathbf{X}\mathbf{A} - \mathbf{X}\mathbf{B}\|_F^2 + \|\mathbf{X}\mathbf{A}_c\|_F^2 + \lambda_2 \sum_{j=1}^K \|\mathbf{b}_j\|_2^2 + \sum_{j=1}^K \lambda_{1,j} \|\mathbf{b}_j\|_1 \\ &= \sum_{j=1}^K \|\mathbf{X}\mathbf{a}_j - \mathbf{X}\mathbf{b}_j\|_2^2 + \lambda_2 \sum_{j=1}^K \|\mathbf{b}_j\|_2^2 + \sum_{j=1}^K \lambda_{1,j} \|\mathbf{b}_j\|_1 + \|\mathbf{X}\mathbf{A}_c\|_F^2 \\ &= \sum_{j=1}^K h_j(\mathbf{a}_j, \mathbf{b}_j) + \|\mathbf{X}\mathbf{A}_c\|_F^2, \end{aligned}$$

kde jsme označili $h_j(\mathbf{a}_j, \mathbf{b}_j) = \|\mathbf{X}\mathbf{a}_j - \mathbf{X}\mathbf{b}_j\|_2^2 + \lambda_2 \|\mathbf{b}_j\|_2^2 + \lambda_{1,j} \|\mathbf{b}_j\|_1$ pro $j \in [K]$.

Uvažujme nyní $j \in [K]$ pevné a počítejme

$$\begin{aligned} &h_j(\mathbf{a}_j^{(l)}, \mathbf{b}_j^{(l)}) - h_j(\mathbf{a}_j^{(l)}, \mathbf{b}_j^{(l+1)}) = \\ &= \|\mathbf{X}\mathbf{a}_j^{(l)} - \mathbf{X}\mathbf{b}_j^{(l)}\|_2^2 - \|\mathbf{X}\mathbf{a}_j^{(l)} - \mathbf{X}\mathbf{b}_j^{(l+1)}\|_2^2 + \\ &\quad + \lambda_2 \left[\|\mathbf{b}_j^{(l)}\|_2^2 - \|\mathbf{b}_j^{(l+1)}\|_2^2 \right] + \lambda_{1,j} \left[\|\mathbf{b}_j^{(l)}\|_1 - \|\mathbf{b}_j^{(l+1)}\|_1 \right] \quad (4.24) \\ &\geq 2 \left(\mathbf{X}\mathbf{b}_j^{(l+1)} - \mathbf{X}\mathbf{a}_j^{(l)} \right)^\top \left(\mathbf{X}\mathbf{b}_j^{(l)} - \mathbf{X}\mathbf{b}_j^{(l+1)} \right) + \\ &\quad + \lambda_2 \left[\|\mathbf{b}_j^{(l)}\|_2^2 - \|\mathbf{b}_j^{(l+1)}\|_2^2 \right] + \lambda_{1,j} \left[\|\mathbf{b}_j^{(l)}\|_1 - \|\mathbf{b}_j^{(l+1)}\|_1 \right]. \end{aligned}$$

V předchozí nerovnosti jsme využili lemma 4.9 (b) pro $\mathbf{y} = \mathbf{X}\mathbf{a}_j^{(l)}$.

Nyní je třeba využít faktu, že $\mathbf{b}_j^{(l+1)}$ je řešením příslušné úlohy elastické sítě, konkrétně

$$\mathbf{b}_j^{(l+1)} = \arg \min_{\mathbf{b}_j \in \mathbb{R}^p} \|\mathbf{X}\mathbf{a}_j^{(l)} - \mathbf{X}\mathbf{b}_j\|_2^2 + \lambda_2 \|\mathbf{b}_j\|_2^2 + \lambda_{1,j} \|\mathbf{b}_j\|_1.$$

Karushovy–Kuhnovy–Tuckerovy (KKT) podmínky pro tuto úlohu² říkají, že

$$-2\mathbf{X}^\top \left(\mathbf{X}\mathbf{a}_j^{(l)} - \mathbf{X}\mathbf{b}_j^{(l+1)} \right) + 2\lambda_2 \mathbf{b}_j^{(l+1)} + \lambda_{1,j} \hat{\mathbf{u}} = 0, \quad (4.25)$$

kde $\hat{\mathbf{u}} \in \mathbb{R}^p$ je vektor (*subgradient* pro ℓ_1 -normu) splňující:

1. $\forall i \in [p] : |\hat{u}_i| \leq 1$ a
2. $\left(\mathbf{b}_j^{(l+1)} \right)^\top \hat{\mathbf{u}} = \|\mathbf{b}_j^{(l+1)}\|_1$.

Vynásobením rovnosti (4.25) členem $\left(\mathbf{b}_j^{(l)} - \mathbf{b}_j^{(l+1)} \right)^\top$ zleva dostáváme rovnost

$$\begin{aligned} 2 \left(\mathbf{X}\mathbf{b}_j^{(l)} - \mathbf{X}\mathbf{b}_j^{(l+1)} \right)^\top \left(\mathbf{X}\mathbf{a}_j^{(l)} - \mathbf{X}\mathbf{b}_j^{(l+1)} \right) &= \\ &= 2\lambda_2 \left(\mathbf{b}_j^{(l)} - \mathbf{b}_j^{(l+1)} \right)^\top \mathbf{b}_j^{(l+1)} + \lambda_{1,j} \left(\mathbf{b}_j^{(l)} - \mathbf{b}_j^{(l+1)} \right)^\top \hat{\mathbf{u}}, \end{aligned}$$

²Čtenáře odkazujeme na Murphyho (2012, sekce 13.3.2) pro hezké shrnutí potřebného subdiferenciálního kalkulu pro ℓ_1 -penalizaci s ilustracemi. Tibshirani (2013) pak odvozuje KKT podmínky přímo pro elastickou síť.

kteřou lze po změně znaménka a využití druhé vlastnosti subgradientu psát jako

$$\begin{aligned} 2(\mathbf{X}\mathbf{b}_j^{(l+1)} - \mathbf{X}\mathbf{a}_j^{(l)})^\top (\mathbf{X}\mathbf{b}_j^{(l)} - \mathbf{X}\mathbf{b}_j^{(l+1)}) &= \\ &= -2\lambda_2(\mathbf{b}_j^{(l)} - \mathbf{b}_j^{(l+1)})^\top \mathbf{b}_j^{(l+1)} - \lambda_{1,j}[(\mathbf{b}_j^{(l)})^\top \hat{\mathbf{u}} - \|\mathbf{b}_j^{(l+1)}\|_1]. \end{aligned}$$

Dosažením poslední rovnosti do (4.24) lze pokračovat v odhadech

$$\begin{aligned} h_j(\mathbf{a}_j^{(l)}, \mathbf{b}_j^{(l)}) - h_j(\mathbf{a}_j^{(l)}, \mathbf{b}_j^{(l+1)}) &\geq \\ &\geq \lambda_2[\|\mathbf{b}_j^{(l)}\|_2^2 - \|\mathbf{b}_j^{(l+1)}\|_2^2 - 2(\mathbf{b}_j^{(l)} - \mathbf{b}_j^{(l+1)})^\top \mathbf{b}_j^{(l+1)}] + \\ &\quad + \lambda_{1,j}[\|\mathbf{b}_j^{(l)}\|_1 - (\mathbf{b}_j^{(l)})^\top \hat{\mathbf{u}}]. \end{aligned} \quad (4.26)$$

Penalizace ℓ_1 -normy $\mathbf{b}_j^{(l+1)}$ se vzájemně vyrušily. Zbytek členu u $\lambda_{1,j}$ lze nyní vypustit, jelikož z první vlastnosti subgradientu musí být $\|\mathbf{b}\|_1 \geq \mathbf{b}^\top \hat{\mathbf{u}}$ pro libovolné $\mathbf{b} \in \mathbb{R}^p$. Dostáváme tedy

$$\begin{aligned} h_j(\mathbf{a}_j^{(l)}, \mathbf{b}_j^{(l)}) - h_j(\mathbf{a}_j^{(l)}, \mathbf{b}_j^{(l+1)}) &\geq \lambda_2[\|\mathbf{b}_j^{(l)}\|_2^2 - \|\mathbf{b}_j^{(l+1)}\|_2^2 - 2(\mathbf{b}_j^{(l)} - \mathbf{b}_j^{(l+1)})^\top \mathbf{b}_j^{(l+1)}] \\ &= \lambda_2 \sum_{i=1}^p \left[(b_{ij}^{(l)})^2 - (b_{ij}^{(l+1)})^2 - 2(b_{ij}^{(l)} - b_{ij}^{(l+1)})b_{ij}^{(l+1)} \right] \\ &= \lambda_2 \sum_{i=1}^p (b_{ij}^{(l)} - b_{ij}^{(l+1)})^2 = \lambda_2 \|\mathbf{b}_j^{(l)} - \mathbf{b}_j^{(l+1)}\|_2^2, \end{aligned}$$

kde předposlední rovnost je pouze aplikace triviální číselné rovnosti pro každý sčítanec zvlášť. Tím jsme tedy získali nerovnost

$$h_j(\mathbf{a}_j^{(l)}, \mathbf{b}_j^{(l)}) - h_j(\mathbf{a}_j^{(l)}, \mathbf{b}_j^{(l+1)}) \geq \lambda_2 \|\mathbf{b}_j^{(l)} - \mathbf{b}_j^{(l+1)}\|_2^2. \quad (4.27)$$

Protože $j \in [K]$ bylo libovolné, platí tato nerovnost pro všechny $j \in [K]$.

Zbytek důkazu konvergence S-PCA algoritmu je již standardní argumentace.

Účelová funkce $h(\mathbf{A}, \mathbf{B})$ je omezená zdola nulou. Dále, funkční hodnota účelové funkce zřejmě klesá s každou iterací algoritmu 3.4.2. Zapsané ve správném pořadí (tak jak probíhají iterace algoritmu) to znamená, že pro $l \in \mathbb{N}_0$ platí

$$h(\mathbf{A}^{(l)}, \mathbf{B}^{(l)}) \geq h(\mathbf{A}^{(l)}, \mathbf{B}^{(l+1)}) \geq h(\mathbf{A}^{(l+1)}, \mathbf{B}^{(l+1)}). \quad (4.28)$$

Z monotonie posloupnosti funkčních hodnot a omezenosti zdola tedy musí být

$$h(\mathbf{A}^{(l)}, \mathbf{B}^{(l)}) \rightarrow h^* \in \mathbb{R} \quad \text{pro } l \rightarrow \infty \quad (4.29)$$

Dále, využitím nerovností (4.27) a (4.28) dostáváme

$$\begin{aligned} h(\mathbf{A}^{(l)}, \mathbf{B}^{(l)}) - h(\mathbf{A}^{(l+1)}, \mathbf{B}^{(l+1)}) &\geq h(\mathbf{A}^{(l)}, \mathbf{B}^{(l)}) - h(\mathbf{A}^{(l)}, \mathbf{B}^{(l+1)}) \\ &\geq \sum_{j=1}^K \left[h_j(\mathbf{a}_j^{(l)}, \mathbf{b}_j^{(l)}) - h_j(\mathbf{a}_j^{(l)}, \mathbf{b}_j^{(l+1)}) \right] \\ &\geq \lambda_2 \sum_{j=1}^K \|\mathbf{b}_j^{(l)} - \mathbf{b}_j^{(l+1)}\|_2^2 = \lambda_2 \|\mathbf{B}^{(l)} - \mathbf{B}^{(l+1)}\|_F^2 \end{aligned}$$

Sečtením těchto nerovností pro $l \in \mathbb{N}$ dostáváme odhad

$$h(\mathbf{A}^{(1)}, \mathbf{B}^{(1)}) - h^* \geq \lambda_2 \sum_{l=1}^{\infty} \|\mathbf{B}^{(l)} - \mathbf{B}^{(l+1)}\|_F^2, \quad (4.30)$$

ze kterého okamžitě plyne tvrzení. \square

Poznamenejme, že náš důkaz konvergence S-PCA algoritmu podhaluje význam ℓ_2 -penalizace v účelové funkci úlohy (3.4). Na první pohled by se mohlo zdát, že při přítomnosti ℓ_1 -penalizace je již ℓ_2 -penalizace nadbytečná, že zajišťuje pouze celkově vzato nepotřebnou jednoznačnost řešení jednotlivých úloh elastických sítí a mohla by být vypuštěna. Například Leng a Wang (2009) ℓ_2 -penalizaci skutečně ve svém článku vypouštějí. Nicméně předchozí důkaz konvergence by nefungoval, pokud by bylo $\lambda_2 = 0$. Alternativy by přitom byly dvě. Buď bychom namísto vypuštění členu u $\lambda_{1,j}$ v (4.26) museli najít jiný horní odhad pro $\|\mathbf{b}_j^{(l)} - \mathbf{b}_j^{(l+1)}\|_2^2$ právě pomocí subgradientu ℓ_1 -normy (k čemuž bychom museli předpokládat tzv. *one at a time* podmínku pro lasso, viz Efron a kol., 2004, abychom nenaráželi na problémy s nejednoznačností) nebo bychom nesměli v důkazu oddělovat aktualizace proměnných \mathbf{A} a \mathbf{B} , což by důkaz značně zkomplikovalo. V obou případech si nejsme jisti, zda lze obdržet konvergenci algoritmu bez dodatečných předpokladů. Při testování zdrojového kódu Leng a Wang (2009)³, který, zdá se, skutečně nevyužívá ℓ_2 -penalizace, jsme ve výjimečných případech narážely na různé chybové hlášky. Domníváme se, že to může být způsobeno právě absencí ℓ_2 -penalizace.

4.4.3 IRLS-S-PCA algoritmus

IRLS-S-PCA algoritmus sestává z vnějšího a vnitřního iteračního schématu. Vnější schématem je samotné tělo algoritmu 4.2.1 a vnitřním schématem je volaný algoritmus 4.1.1. Obě tato schémata jsou nekonvexní. K důkazu konvergence IRLS-S-PCA je nutné dobře kontrolovat regularizační parametr δ tak, jak je to popsáno v algoritmu 4.2.1.

Samotný důkaz konvergence je potom pouze kombinací výsledků z předchozích dvou sekcí.

Tvrzení 4.12. Necht $\mathbf{X} \in \mathbb{R}^{n \times p}$ je centrovaná matice dat, $\lambda_2 > 0$ je ladící parametr, $K \in [p]$ je počet hledaných řídkých nábojů a $\lambda_{1,1}, \dots, \lambda_{1,K}$ jsou ladící parametry. Necht $\{\mathbf{B}^{(l)}\}_{l=1}^{\infty}$ je posloupnost matic typu $p \times K$ řídkých nábojů získaná algoritmem 4.2.1. Pak platí

$$\|\mathbf{B}^{(l)} - \mathbf{B}^{(l-1)}\|_F \rightarrow 0 \quad \text{pro } l \rightarrow \infty,$$

tedy IRLS-S-PCA algoritmus je konvergentní.

Důkaz. Označme

$$h(\mathbf{A}, \mathbf{B} | \delta) = \sum_{i=1}^n \|\mathbf{x}^{(i)} - \mathbf{A}\mathbf{B}^T \mathbf{x}^{(i)}\|_2^2 + \lambda_2 \sum_{j=1}^K \|\mathbf{b}_j\|_2^2 + \sum_{j=1}^K \lambda_{1,j} \sum_{i=1}^p \log(b_{ij}^2 + \delta)$$

³Zdrojový kód je dostupný na webové stránce Chenleie Leng (http://www.stat.nus.edu.sg/~stalcr/research.html).

a pro $j \in [K]$ označme

$$h_j(\mathbf{a}_j, \mathbf{b}_j | \delta) = \|\mathbf{X}\mathbf{a}_j - \mathbf{X}\mathbf{b}_j\|_2^2 + \lambda_2 \|\mathbf{b}_j\|_2^2 + \lambda_{1,j} \sum_{i=1}^p \log(b_{ij}^2 + \delta).$$

Dále necht posloupnost $\{L^{(l)}\}_{l=1}^{\infty}$ zachycuje všechny změny parametru L z algoritmu 4.2.1. S tímto označením můžeme obdobně jako v důkazu konvergence S-PCA algoritmu psát

$$\begin{aligned} h(\mathbf{A}^{(l)}, \mathbf{B}^{(l)} | \delta_{L^{(l)}}) - h(\mathbf{A}^{(l+1)}, \mathbf{B}^{(l+1)} | \delta_{L^{(l+1)}}) &\geq \\ &\geq h(\mathbf{A}^{(l)}, \mathbf{B}^{(l)} | \delta_{L^{(l)}}) - h(\mathbf{A}^{(l)}, \mathbf{B}^{(l+1)} | \delta_{L^{(l+1)}}) \\ &\geq \sum_{j=1}^K \left[h_j(\mathbf{a}_j^{(l)}, \mathbf{b}_j^{(l)} | \delta_{L^{(l)}}) - h_j(\mathbf{a}_j^{(l)}, \mathbf{b}_j^{(l+1)} | \delta_{L^{(l+1)}}) \right]. \end{aligned}$$

Nyní, z nerovnosti (4.22) plyne, že vnitřní iterační schéma garantuje snížení funkční hodnoty v každém kroku. Díky tomu můžeme uvažovat, že vnitřní schéma je pouze jednokrokové, čímž jen snížíme rozdíl funkčních hodnot mezi iteracemi IRLS-S-PCA algoritmu. Díky nerovnosti (4.22) tedy dostáváme

$$h(\mathbf{A}^{(l)}, \mathbf{B}^{(l)} | \delta_{L^{(l)}}) - h(\mathbf{A}^{(l+1)}, \mathbf{B}^{(l+1)} | \delta_{L^{(l+1)}}) \geq \sum_{j=1}^K \frac{\lambda_{1,j}}{c_j^2 + \delta_0} \|\mathbf{b}_j^{(l)} - \mathbf{b}_j^{(l+1)}\|_2^2,$$

kde c_j jsou konstanty získané obdobně jako konstanta c v důkazu tvrzení 4.8.

Zvolíme-li nyní $C = \min \left(\frac{\lambda_{1,j}}{c_j^2 + \delta_0}; j = 1, \dots, K \right)$, dostáváme odhad

$$h(\mathbf{A}^{(l)}, \mathbf{B}^{(l)} | \delta_{L^{(l)}}) - h(\mathbf{A}^{(l+1)}, \mathbf{B}^{(l+1)} | \delta_{L^{(l+1)}}) \geq C \|\mathbf{B}^{(l)} - \mathbf{B}^{(l+1)}\|_F^2. \quad (4.31)$$

Zbytek důkazu je již ekvivalentní konci důkazu konvergence S-PCA. Nerovnosti (4.28), (4.29) a (4.31) získáváme úplně stejně, pouze funkce h nyní závisí ještě na regularizačních parametrech.

Tvrzení pak získáváme z analogie rovnosti (4.31). \square

4.5 Praktická implementace

IRLS-S-PCA algoritmus jsme implementovali v jazyce **R**. V naší implementaci jsme se odchýlili od pseudokódu algoritmu 4.2.1 následujícími způsoby.

Zaprvé, definujeme váhy jinak než je uvedeno v lemmatu 4.3 (c). Používáme tzv. *neseparabilní schéma*, kdy váhy v l -tém kroku nejsou definovány pouze pomocí předchozího řešení $\beta^{(l-1)}$, nýbrž pomocí několika posledních řešení. Konkrétně počítáme váhy jako (znormovaný) součin vah z lemmatu 4.3 (c) spočtený přes několik posledních iterací. Důvod této heuristiky je pouze drobné zlepšení výsledků obdržných IRLS-S-PCA algoritmem, které jsme pozorovali při testování různých možností volby vah. Podobné zlepšení v důsledku aplikace neseparabilního schématu bylo pozorováno také v Gorodnitsky a Rao (1997) nebo Candés a kol. (2008).

Zadruhé, během iterací vnitřního schématu (algoritmus 4.1.1) prahujeme velmi malé koeficienty (je-li absolutní hodnota nějakého koeficientu v l -tém kroku odhadnuta v absolutní hodnotě nižší než požadovaná přesnost ξ_{conv} , je tento koeficient prahován na nulu). Tato heuristika byla opět poprvé navržena v Gorodnitsky a Rao (1997), kde však měla mnohem zásadnější význam, jelikož autoři nepoužívali regularizační parametr δ , a tudíž jednou prahovaný koeficient již nutně zůstával nulový ve všech dalších iteracích.

Zatřetí, abychom zajistili konvergenci k dostatečně řídkému odhadu nábojů bez nutnosti hledat vhodný ladící parametr λ , zvyšujeme mezi iteracemi parametr λ tehdy, pokud nedochází k prahování parametrů popsaném v předchozím odstavci.

A konečně, používáme defaultní nastavení posloupnosti $\{\delta_l\}_{l=1}^{\infty}$ tak, jak je uvedeno v algoritmu 4.2.1 pokaždé, když je toto vnitřní schéma zavoláno vnějším schématem algoritmu 4.2.1.

Mohli bychom diskutovat, že pouze poslední z jmenovaných odchylek od pseudokódu algoritmu 4.2.1 může zničit dokázanou globální konvergenci algoritmu. Nicméně možná ztráta konvergence kvůli těmto heuristikám není důležitá. Konvergenci můžeme numericky kontrolovat za běhu algoritmu a pokud vidíme, že algoritmus diverguje (po jistém počtu kroků se nesnižuje norma rozdílu posledních dvou řešení), můžeme heuristiky opustit a přejít k jednomu ze schémat, pro které máme konvergenci zaručenou. Navzdory použitým heuristikám jsme však divergenci algoritmu v naší numerické studii nepozorovali, proto numerickou kontrolu konvergence neimplementujeme.

Poznámka 4.13. Naše implementace IRLS-S-PCA algoritmu si zachovává příznivé vlastnosti S-PCA algoritmu diskutované v poznámce 3.5. Vlastnost (a) jsme zajistili výše popsaným prahováním. Vlastnost (b) plyne z faktu, že bez omezení na kardinalitu nábojů algoritmus startuje z řešení obyčejné AHK (inicializace vah) a je okamžitě zastaven. Vlastnosti (c)-(e) pocházejí z vnějšího iteračního schématu, které má IRLS-S-PCA společné s S-PCA. Spolu s našimi důkazy konvergence tedy IRLS-S-PCA algoritmus disponuje stejnými vlastnostmi jako S-PCA algoritmus, který patří k v praxi nejužívanějším algoritmům pro AŘHK. Proti S-PCA však IRLS-S-PCA vykazuje lepší chování v naší simulační studii v kapitole 5. \diamond

Pro úplnost ještě uvedme, že úlohy vážené hřebenové regrese (4.9), které jsou řešeny v každém kroku iteračního schématu, řešíme pomocí singulárního rozkladu. Úlohu (4.9) převádíme na úlohu obyčejné hřebenové regrese převážením dat, viz poznámka 4.4, a tu pak řešíme singulárním rozkladem, viz poznámka 2.8.

4.6 Shrnutí

V této kapitole jsme diskutovali nekonvexní penalizaci metody nejmenších čtverců. Navrhli jsme iterační algoritmus pro řešení této nekonvexní úlohy pomocí alternující minimalizace pomocné funkce o dodatečných proměnných, které lze interpretovat jako váhy. Ukázali jsme, že výsledný algoritmus postupuje řešením vážených úloh hřebenové regrese a je tedy instancí IRLS algoritmu.

Specifikem našeho přístupu je konkrétní volba penalizace (logaritmická spojitě diferencovatelná funkce) a variabilita regularizačního parametru. Význam obojího jsme ilustrovali na jednoduchém příkladu.

Navržený iterační postup pro minimalizaci dané penalizované úlohy nejmenších čtverců jsme zakomponovali do alternujícího schématu pro hledání řídkých nábojů. Konkrétně jsme nahradili ℓ_1 -penalizaci používanou v S-PCA algoritmu za studovanou nekonvexní penalizaci. Vnikl nový algoritmus pro AŘHK, jehož konvergenci jsme ukázali spolu s konvergencí původního S-PCA algoritmu.

5. Simulační studie

V této kapitole porovnáme na několika simulačních příkladech algoritmy ze sekce 3.4 s nově navrženým algoritmem, který jsme představili v minulé kapitole.

Poznámka 5.1. Kvalitu numerických výsledků posuzujeme z několika hledisek. Označme \mathbf{X} matici nasimulovaných dat, $\tilde{\mathbf{v}}_1, \dots, \tilde{\mathbf{v}}_K$ řádké náboje získané jedním z algoritmů (odhady skutečných nábojů $\mathbf{v}_1, \dots, \mathbf{v}_K$) a $\hat{\mathbf{v}}_1, \dots, \hat{\mathbf{v}}_K$ obyčejné náboje získané pomocí AHK. Pak nás zajímá:

1. Shoda opory $\tilde{\mathbf{v}}_i$ s oporou \mathbf{v}_i , $i = 1, \dots, K$.
2. Úhly mezi $\tilde{\mathbf{v}}_i$ a \mathbf{v}_i , $i = 1, \dots, K$, případně úhly mezi hlavními podprostory (spočteny obdobně jako v příkladu 1.17), jejichž ortonormální báze jsou $\tilde{\mathbf{v}}_1, \dots, \tilde{\mathbf{v}}_K$ a $\mathbf{v}_1, \dots, \mathbf{v}_K$.
3. Množství rozptylu, jež je vysvětlen řídkými komponentami $\mathbf{X}\tilde{\mathbf{v}}_1, \dots, \mathbf{X}\tilde{\mathbf{v}}_K$. Rozptyl vysvětlený řídkou komponentou $\mathbf{X}\tilde{\mathbf{v}}_i$ je vždy uváděn relativně vůči rozptylu vysvětlenému obyčejnou hlavní komponentou $\mathbf{X}\hat{\mathbf{v}}_i$.

Cílem každé AŘHK metody je samozřejmě získávat řídké náboje, které mají co největší shodu opory se skutečným nábojem, co nejmenší úhel od skutečného náboje a vysvětlují co nejvíce rozptylu. První dva cíle jsou ve vzájemném souladu, zatímco třetí jim může částečně odporovat, jak bylo diskutováno v bodě 3 poznámky 3.4. \diamond

Všechny simulace byly provedeny v software **R** (verze 3.3.3, R Core Team, 2017), implementace jednotlivých algoritmů je vlastní (přiložena k této práci v elektronické podobě) s výjimkou S-PCA algoritmu, pro který byla použita implementace z balíku `elasticnet` (Zou a Hastie, 2012). ADA-S-PCA je jen jednoduchou úpravou S-PCA algoritmu. Pro implementaci IRLS-S-PCA byla použita kostra S-PCA algoritmu, v níž byl LARS algoritmus vnitřním optimalizačním schématem ze sekce 4.1.

5.1 Normální špičatý model s jednou komponentou

Nechť $p = 200$. Nechť vektor \mathbf{w} má oporu S získanou prostým náhodným výběrem z množiny $[p]$ o rozsahu $s = 50$ a na této opoře má složky rovnoměrně rozdělené na intervalu $(0,1)$. Nechť je \mathbf{v} znormovaná verze \mathbf{w} , tj. $\mathbf{v} = \mathbf{w}/\|\mathbf{w}\|_2$.

Uvažujme nyní kovarianční matici Σ mající první vlastní vektor \mathbf{v} s příslušným vlastním číslem $\theta > 1$ (bude specifikováno později) získanou následujícím způsobem (Shen a Huang, 2008). Nejprve vytvoříme matici

$$\mathbf{W} = \left(\mathbf{v} \left| \frac{\mathbf{w}_2}{\|\mathbf{w}_2\|_2} \right| \dots \left| \frac{\mathbf{w}_p}{\|\mathbf{w}_p\|_2} \right| \right),$$

kde $\mathbf{w}_2, \dots, \mathbf{w}_p$ mají nezávislé rovnoměrně rozdělené složky. Na tuto matici následně použijeme Gramovu-Schmidtovu ortogonalizační metodu k získání ortonormální matice

$$\mathbf{V} = (\mathbf{v}|\mathbf{v}_2|\dots|\mathbf{v}_p).$$

Připomeňme, že \mathbf{V} je přesně matice \mathbf{Q} z QR-rozkladu matice \mathbf{W} . Zbývá dodat vlastní čísla ve formě diagonální matice

$$\mathbf{\Lambda} = \text{diag}(\theta, 1, \dots, 1) \in \mathbb{R}^{p \times p},$$

abychom získali požadovanou kovarianční matici

$$\mathbf{\Sigma} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^\top.$$

Ve skutečnosti není třeba počítat přímo matici $\mathbf{\Sigma}$, jelikož pro generování dat nám stačí „odmocninová“ matice $\mathbf{V}\mathbf{\Lambda}^{\frac{1}{2}}$, kde $\mathbf{\Lambda}^{\frac{1}{2}} = \text{diag}(\sqrt{\lambda_{i=1}^p})$. Pro $\mathcal{X} \sim \mathcal{N}_p(0, I)$ je

$$\mathbf{V}\mathbf{\Lambda}^{\frac{1}{2}}\mathcal{X} \sim \mathcal{N}_p(0, \mathbf{\Sigma}). \quad (5.1)$$

Z normálního rozdělení (5.1) generujeme výše uvedeným způsobem $n = 50$ nezávislých pozorování a získanou matici dat sloupcově centrujeme. Následně nalezneme odhad prvního náboje pomocí uvažovaných algoritmů a ohodnotíme jej spočtením měr kvality, které jsme popsali v poznámce 5.1. Tuto proceduru nezávisle opakujeme 100-krát pro každou hodnotu $\theta \in \{3, 4, 5, 6, 7\}$. Jednu proceduru generování dat pro konkrétní θ nazýváme *tah*.

Poznámka 5.2. Model (5.1), z něhož generujeme data, je velmi podobný normálnímu špičatému modelu o jedné komponentě. Také je zde pouze jeden vlastní vektor (v obecnějším případě jich může být samozřejmě více, pokud do matice \mathbf{W} dodáme více ortogonálních vektorů „ručně“) oddělený od šumu ve zbylých směrech. Rozdíl je, že kovarianční matice normálního špičatého modelu je z principu řídká. Jsou-li S_1, \dots, S_K opory prvních K nábojů, může mít kovarianční matice špičatého modelu nanejvýš $|S_1 \cup \dots \cup S_K|^2 + (p - K)$ nenulových prvků. Oproti tomu model diskutovaný výše je obecnější, neboť jeho kovarianční matice nemusí obsahovat žádné nulové prvky. Naopak randomizace opory \mathbf{w} je téměř zbytečná, přesto ji však provádíme, aby hladový algoritmus nemohl být nějakým způsobem zvýhodněn. \diamond

Nejprve se podívejme, jak dobře odhadují jednotlivé algoritmy oporu skutečného náboje \mathbf{v} . Pro každý tah a každý uvažovaný algoritmus spočteme řídký odhad náboje pro všechny hodnoty kardinality $s = 1, \dots, 200$, čímž získáme kompletní dráhy řešení. Přirozeným způsobem porovnání těchto drah je vykreslit poměr správně odhadnutých nenulových koeficientů (*true positive rate*, TPR) proti počtu špatně odhadnutých nenulových koeficientů (*false positive rate*, FPR). Tím získáme pro každý algoritmus křivku ležící uvnitř čtverce $[0, 1]^2$ vedoucí z bodu $(0, 0)$ do bodu $(1, 1)$. Viz obrázek 5.1.

Obě osy grafu na obrázku 5.1 závisí implicitně na kardinalitě s . Chování algoritmů pro správně nastavenou kardinalitu $s = 50$ je zachyceno průsečíky křivek pro jednotlivé algoritmy s plnou šedou čarou v pozadí (počet skutečně nulových koeficientů je 150, zatímco počet nenulových je 50). Výše položená křivka znamená lepší chování algoritmu. Ideální chování by bylo, kdyby dráha algoritmu šla

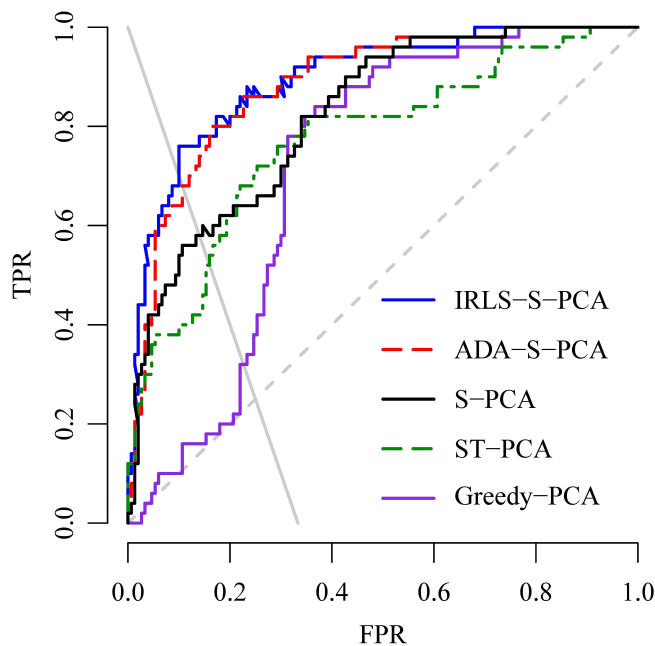
po úsečce z bodu (0,0) do bodu (0,1) a z něj po další úsečce do bodu (1,1). To by znamenalo, že algoritmus při libovolném nastavení $s \leq 50$ odhadne jako nenulové pouze ty koeficienty, které skutečně nenulové jsou, a až pro $s \geq 50$ začíná vybírat i koeficienty fakticky nulové. Naše nastavení je příliš těžké, než aby nějaký z algoritmů docílil tohoto optimálního chování. Naopak, za velmi špatné chování by se dalo považovat, pokud by křivka nějakého algoritmu ležela pod přerušovanou diagonálou. To by znamenalo, že daný algoritmus není v odhadování opory lepší než náhodný výběr.

Chceme-li shrnout fungování algoritmu z celé jeho dráhy do jednoho čísla, nabízí se spočítat plochu pod grafem. Křivka na obrázku 5.1 je analogií k tzv. ROC křivce – populárního nástroje při posuzování vlastností binárních klasifikátorů. Viz. např. Antoch a kol., 2010. Plocha pod grafem je pak nejčastěji užívanou jednorozměrnou charakteristikou ROC křivky. Tato charakteristika leží v intervalu $[0,1]$. V ideálním případě bychom chtěli číslo rovno jedné, a naopak by toto číslo nemělo klesnout pod $1/2$, jak jsme popsali v předchozím odstavci. Na obrázku 5.2 jsou vyznačeny průměrné plochy pod grafem dráhy jednotlivých algoritmů pro různé hodnoty θ spočtené ze 100 nezávislých tahů.

Zadruhé, dobře odhadnutá opora neznámá nutně dobrý odhad náboje \mathbf{v} . Na obrázku 5.3 zobrazujeme úhly mezi \mathbf{v} a jeho řídkými odhady získanými pomocí různých algoritmů pro správnou volbu kardinality $s = 50$ a $\theta = 5, 6, 7$.

Zatřetí, na obrázku 5.4 vykresluje průměrné množství rozptylu vysvětlené získanou řídkou HK dělené množstvím rozptylu, které vysvětluje první klasická HK.

Nakonec, jelikož informace na obrázcích se pro přehlednost (s výjimkou ob-



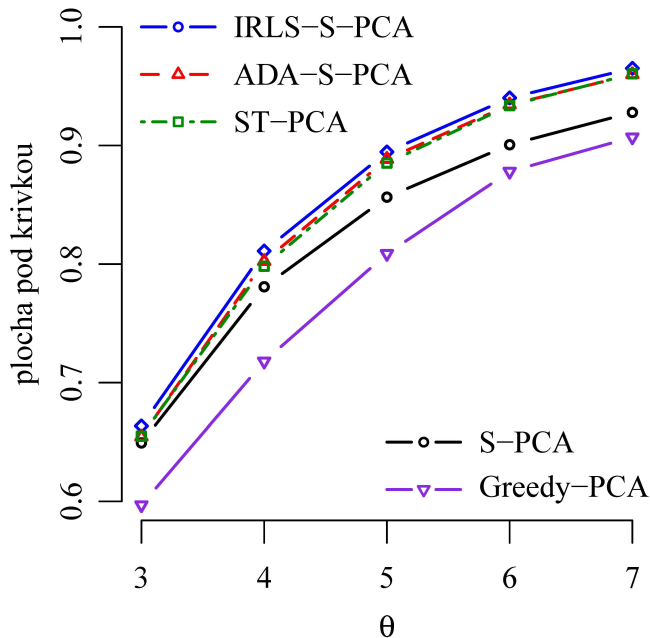
Obrázek 5.1: Křivky znázorňující počet správně odhadnutých nenulových koeficientů (TPR) proti počtu špatně odhadnutých nenulových koeficientů na základě jednoho tahu z modelu (5.1) pro volbu $\theta = 5$. Křivka ve vyšší úrovni znamená lepší chování algoritmu.

Tabulka 5.1: Směrodatné odchyly pro různé veličiny a různé algoritmy na základě 100 nezávislých tahů o 100 pozorováních z modelu (5.1) pro $\theta = 5$. *Plocha pod křivkou* je spočtena integrací přes všechny hodnoty kardinality s . Směrodatné odchyly pro *množství vysvětleného rozptylu* a *úhel mezi nábojem a jeho odhadem* jsou udány pro správnou volbu kardinality $s = 50$.

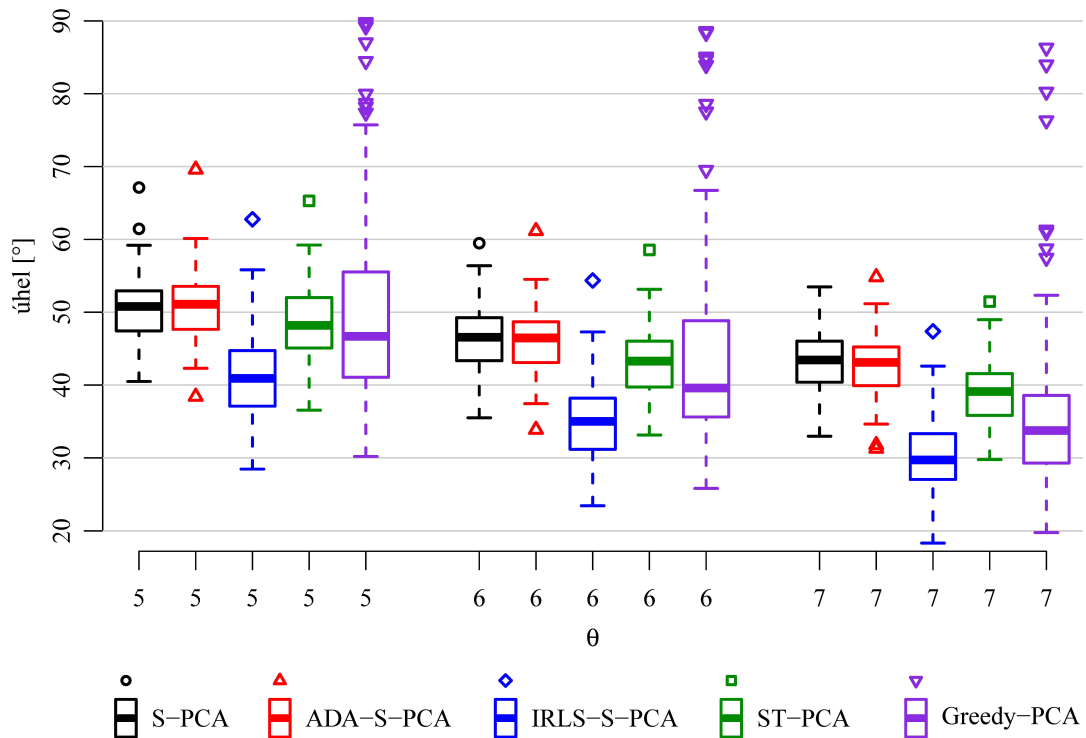
Algoritmus	Plocha pod křivkou	Vysvětlený rozptyl	Úhel [°]
S-PCA	0.05	0.20	4.33
ADA-S-PCA	0.05	0.21	4.73
IRLS-S-PCA	0.05	0.15	6.16
ST-PCA	0.06	0.15	5.16
Greedy-PCA	0.09	0.24	14.1

rázku 5.3) omezují na podání informace o průměrech přes 100 simulačních tahů, doplňujeme tyto grafy také o spočtené směrodatné odchyly různých veličin. V tabulce 5.1 jsou dány výběrové směrodatné odchyly spočtené ze 100 tahů s volbou $\theta = 5$ pro plochy pod křivkou, pro úhly mezi skutečným a odhadnutým nábojem (při nastavení $s = 50$) a pro podíl vysvětleného rozptylu (opět při nastavení $s = 50$).

Nyní okomentujme obdržené výsledky. Zaprvé, Greedy-PCA dává spíše špatné výsledky z hlediska různých měř s výjimkou vysvětleného rozptylu. Greedy-PCA se chová poměrně nestabilně, viz obrázek 5.3 a tabulka 5.1. Přestože pro většinu tahů dává kompetitivní výsledky, pro některé tahy produkuje velmi špatné výsledky jako například pro tah zachycený na obrázku 5.1. Jeho špatné odhadování opory způsobuje, že i přes normalizaci popsanou v sekci 3.2 vysvětluje pouze průměrné množství rozptylu, nižší než například ST-PCA.



Obrázek 5.2: Průměrná plocha pod křivkou pro křivky znázorněné na obrázku 5.1. Větší plocha pod křivkou znamená lepší chování algoritmu. Pro každé $\theta = 3, 4, 5, 6, 7$ jsou dané průměry spočteny ze 100 nezávislých tahů z modelu 5.1.

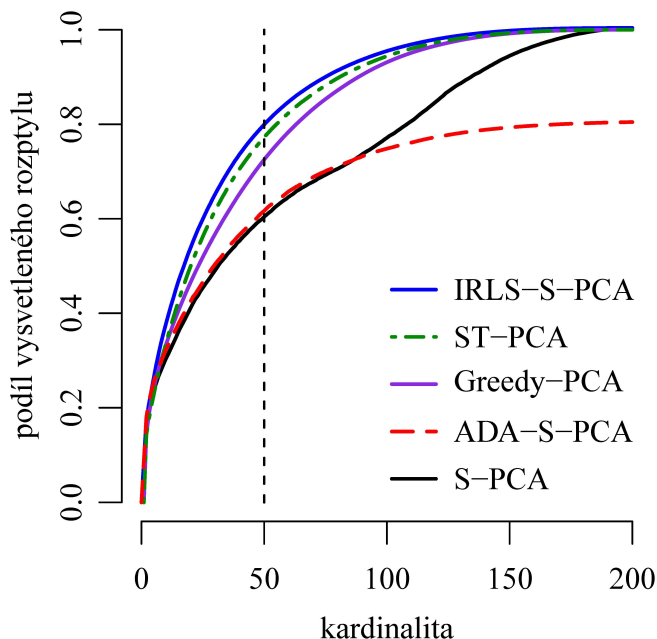


Obrázek 5.3: Krabicové diagramy pro úhly mezi skutečným nábojem \mathbf{v} a jeho odhady spočtené různými algoritmy. Nižší úhly znamenají lepší chování algoritmu. Každý krabicový graf je vytvořen na základě 100 nezávislých tahů.

Zadruhé, ačkoliv ADA-S-PCA znatelně zlepšuje kvalitu odhadu opory oproti S-PCA, jeho výkon není o nic lepší než S-PCA z hlediska vysvětleného rozptylu nebo úhlů mezi náboji. Je to dáno patrně ztrátou vlastnosti (d) z poznámky 3.5, která je evidentní z obrázku 5.4. Pouze informativně uvedme, že jsme zkoumali také chování ADA-S-PCA s normalizací, která byla popsána v sekci 3.2. Metoda tím byla značně vylepšena a kvalitativně se zařadila mezi IRLS-S-PCA a ST-PCA, jejíž fungování na tomto – nutno říci poměrně jednoduchém – příkladě je překvapivě dobré.

Konečně, námi navržený IRLS-S-PCA algoritmus produkuje celkově nejlepší výsledky. V odhadu opory se vyrovnává druhému nejlepšímu ADA-S-PCA a navíc v množství vysvětleného rozptylu těsně překonává druhé nejlepší ST-PCA. Všechny metody pak IRLS-S-PCA překonává v tom, že úhly mezi odhadem $\tilde{\mathbf{v}}$ získaným pomocí IRLS-S-PCA a samotným vektorem \mathbf{v} jsou nejnižší.

Poznámka 5.3. Pro ilustraci uvedme přibližné doby běhu jednotlivých algoritmů při $\theta = 5$ a $s = 50$. Doba běhu algoritmů v jednom tahu byla v jednotkách sekund přibližně $8 \cdot 10^{-3}$ pro ST-PCA, $2 \cdot 10^{-1}$ pro S-PCA, $3 \cdot 10^{-1}$ pro ADA-S-PCA, $7 \cdot 10^{-1}$ pro IRLS-S-PCA a $7 \cdot 10^0$ pro Greedy-PCA. Počet iterací pro S-PCA algoritmus se pohyboval v rozpětí 4-7 zatímco ADA-S-PCA v rozpětí 5-10. Tyto vnější iterace provádí také IRLS-S-PCA, jejich počet byl pro něj v rozmezí 2-5. Počty vnitřních iterací (iterace z řešení penalizovaných problémů) se pro IRLS-S-PCA pohybovaly v rozpětí 9-11. Počty iterací samozřejmě závisí na požadované přes-



Obrázek 5.4: Rozptyl vysvětlený řídkými HK dělený rozptylem vysvětleným obyčejných HK. Vyšší podíl vysvětleného rozptylu znamená lepší chování algoritmu. Pro $n = 100$, $\theta = 5$ a všechny hodnoty kardinality $s = 1, \dots, 200$ je udán relativní vysvětlený rozptyl jako průměr přes 100 nezávislých tahů z modelu (5.1).

nosti konvergence. Ta byla nastavena jako 10^{-3} pro vnější iterace všech S-PCA algoritmů a 10^{-6} pro vnitřní iterace IRLS-S-PCA. Toto nastavení bylo zachováno ve všech výpočtech. \diamond

5.2 Normální špičatý model s dvěma komponentami

Uvažujme nyní normální špičatý model o $K = 2$ komponentách pro $n = 50$ a $p = 500$. Model je převzatý z Shen and Huang (2008, příklad 2), takže naše výsledky umožňují přímé porovnání. Pro jeden tah generujeme

$$\begin{aligned} \mathbf{w}_1 &= (u_1, \dots, u_{10}, 0, \dots, 0, 0, \dots, 0) \in \mathbb{R}^{500}, \\ \mathbf{w}_2 &= (0, \dots, 0, u_{11}, \dots, u_{20}, 0, \dots, 0) \in \mathbb{R}^{500}, \end{aligned}$$

kde u_1, \dots, u_{20} jsou nezávislé rovnoměrně rozdělené koeficienty na $(0,1)$. První dva náboje \mathbf{v}_1 a \mathbf{v}_2 jsou normované verze \mathbf{w}_1 a \mathbf{w}_2 o kardinalitách $s_1 = s_2 = 10$. Jejich příslušná vlastní čísla jsou 400 a 300, ostatní vlastní čísla jsou všechna rovna jedné. Pomocí procedury z minulého příkladu generujeme (hustou) kovarianční matici Σ s prvními dvěma vlastními vektory \mathbf{v}_1 a \mathbf{v}_2 . Následně generujeme $n = 50$ pozorování z modelu

$$\mathcal{X} \sim \mathcal{N}(0, \Sigma). \quad (5.2)$$

Matici dat sloupcově centrujeme, a následně určujeme odhady nábojů pomocí uvažovaných algoritmů a tyto odhady hodnotíme z hlediska měř kvality, které jsme popsali v poznámce 5.1. Opět provádíme 100 nezávislých tahů.

Tabulka 5.2: Výběrové mediány spočteny ze 100 nezávislých tahů. Mediány jsou udány pro úhly mezi náboji \mathbf{v}_1 a \mathbf{v}_2 a jejich řídkými odhady $\tilde{\mathbf{v}}_1$ a $\tilde{\mathbf{v}}_2$ a také pro úhel mezi skutečným a odhadnutým hlavním podprostorem (HP), dále pro procentuální podíl správně odhadnutých nulových koeficientů (TNR) a pro relativní procentuální podíl vysvětleného rozptylu jednotlivými řídkými náboji a oběma současně (tedy $\sum_{1,2}$ je součet rozptylů vysvětlených pomocí $\tilde{\mathbf{v}}_1$ a $\tilde{\mathbf{v}}_2$ dělený součtem rozptylů vysvětlených pomocí prvních dvou obyčejných HK).

Algoritmus	Úhel [°]			TNR [%]		Rozptyl [%]		
	$\tilde{\mathbf{v}}_1$	$\tilde{\mathbf{v}}_2$	HP	$\tilde{\mathbf{v}}_1$	$\tilde{\mathbf{v}}_2$	$\tilde{\mathbf{v}}_1$	$\tilde{\mathbf{v}}_2$	$\sum_{1,2}$
S-PCA	50.3	55.0	44.4	99.2	99.0	26.1	28.5	27.6
ADA-S-PCA	7.3	9.8	6.1	100.0	100.0	44.5	48.8	46.4
IRLS-S-PCA	0.4	0.6	0.4	100.0	100.0	44.8	49.2	46.9
ST-PCA	69.3	75.9	59.2	99.4	99.2	12.8	12.8	12.7
Greedy-PCA	90.0	90.0	90.0	98.0	98.0	15.7	0.0	8.3

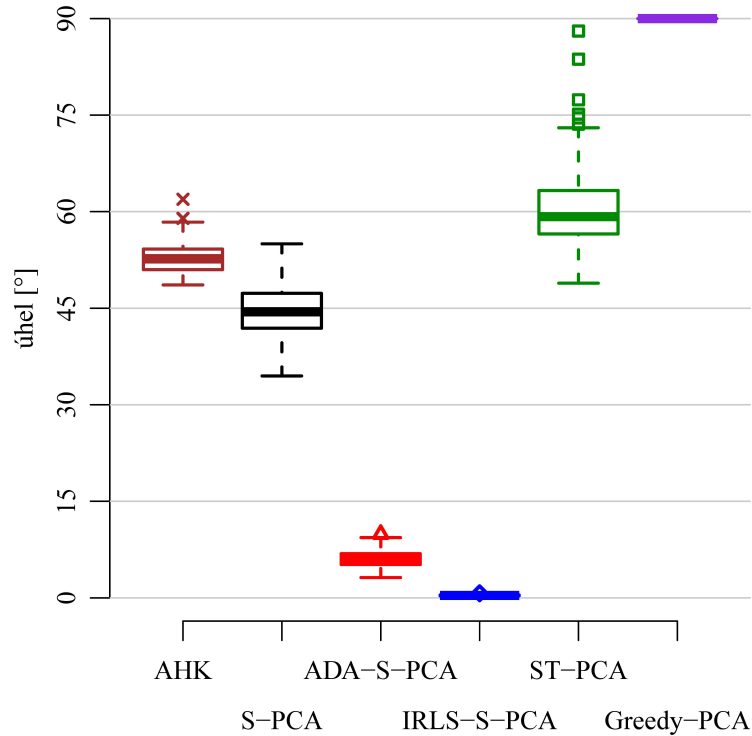
Tento model se liší od modelu z předchozího příkladu v tom, že poměr dimenze p a počtu pozorování n se rapidně zvýšil. Pro vyvážení složitosti problému se zvýšila vlastní čísla příslušná nábojům (síla signálu).

Pro každý tah jsou určeny řídké odhady $\tilde{\mathbf{v}}_1$ a $\tilde{\mathbf{v}}_2$ prvních dvou nábojů pomocí různých algoritmů. Následně jsou spočteny úhly mezi náboji a jejich odhady a úhel mezi skutečným a odhadnutým hlavním podprostorem dimenze dva. Dále uvádíme procentuální podíl správně odhadnutých nulových koeficientů (*true negative rate*, TNR)¹, což je v tomto případě číslo ležící mezi 98% a 100%. Nakonec počítáme množství vysvětleného rozptylu oběma řídkými HK (jako podíl vůči množství rozptylu vysvětleného prvními dvěma obyčejnými HK) jednotlivě i sdruženě. Mediány těchto veličin spočtené ze 100 tahů jsou uvedeny v tabulce 5.2.

Z tabulky 5.2 není patrná následující nepříjemnost, kterou tento příklad disponuje. Přestože první dvě vlastní čísla jsou relativně oddělena (jejich podíl je 4/3), dimenze p je natolik vysoká, že všechny algoritmy (v různé míře) zamění pořadí nábojů \mathbf{v}_1 a \mathbf{v}_2 , tj. $\hat{\mathbf{v}}_1$ je někdy vhodnější odhad \mathbf{v}_2 a naopak. Jelikož jsou v tabulce 5.2 udány pouze mediány a zmíněná záměna přirozeně probíhá v méně než polovině tahů, má tato záměna na výsledky v tabulce 5.2 jen malý efekt. Přesto je třeba si uvědomit, že mnohem důležitější než schopnost odhadnout kvalitně jednotlivé náboje je schopnost jednotlivých algoritmů odhadnout dobře hlavní podprostor dimenze dva. Nejdůležitějším sloupcem tabulky 5.2 je tedy sloupec označený jako HP udávající úhel mezi skutečným a odhadnutým hlavním podprostorem. Proto také na obrázku 5.5 vykreslujeme krabicové diagramy tohoto úhlu pro různé algoritmy. Druhým nejdůležitějším sloupcem tabulky 5.2 je pak poslední sloupec udávající podíl kumulovaného vysvětleného rozptylu (tj. součet rozptylů vysvětlených prvními dvěma řídkými HK dělený součtem rozptylů vysvětlených prvními dvěma obyčejnými HK). Krabicové diagramy této veličiny jsou vykresleny na obrázku 5.6.

Poznámka 5.4. Tabulku podobnou tabulce 5.2 lze nalézt v Shen a Huang (2008, tabulka 3) s tím rozdílem, že autoři neuvádějí vysvětlený rozptyl ani úhel mezi

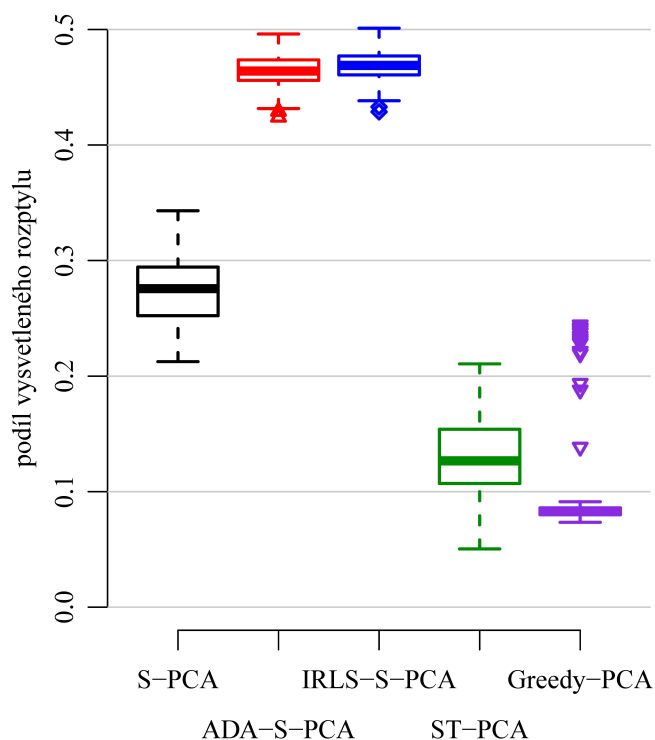
¹Toto je změna oproti minulému příkladu, kdy jsme počítali lépe interpretovatelný podíl správně odhadnutých *nenulových* koeficientů (TPR). Účelem této změny je, aby naše výsledky byly snadněji porovnatelné s těmi původními v Shen a Huang (2008).



Obrázek 5.5: Krabicové diagramy úhlu mezi skutečným a odhadnutým hlavním podprostorem dimenze 2. Menší úhel znamená lepší chování daného algoritmu. Každý diagram je tvořen 100 nezávislými tahy z modelu (5.2). Na rozdíl od tabulky 5.2 je zde uveden i úhel pro obyčejnou AHK (PCA).

podprostory (HP), přestože z diskuze v předchozím odstavci je patrné, že právě HP je nejdůležitější veličinou, kterou měříme. Autoři také uvádějí výsledky pro S-PCA algoritmus, které tudíž mohou sloužit jako pojící prvek pro porovnání našich algoritmů s mnoha dalšími algoritmy uvažovanými v Shen a Huang (2008). Nicméně, naše výsledky obdržené jak pro S-PCA algoritmus tak pro obyčejnou AHK (PCA, viz obrázek 5.5) jsou pesimističtější než výsledky Shena a Huanga. Při našich simulacích jsme dodrželi přesný postup uvedený v Shen a Huang (2008), pro generování matice Σ jsme dokonce použili funkci volně dostupnou na webových stránkách autorů. Jedinou výjimkou je, že Shen a Huang (2008) nemluví v daném příkladě o centrování matice dat. Pokud matici dat nevycentrujeme, implicitně tím využíváme informaci o skutečné střední hodnotě modelu (5.2) a úloha odhadu nábojů se tím zjednodušuje. Skutečně, bez centrování matice dat jsme byli schopni pro obyčejnou AHK a S-PCA obdržet kvalitativně podobné výsledky jako Shen a Huang (2008). Na jedné straně, centrování matice dat je v případě AHK natolik obvyklé, že se v konkrétních simulačních studiích běžně nezdůrazňuje, i když je použito. Na druhé straně, zmíněnou disproporci mezi výsledky jsme i přes značné úsilí nebyli schopni jinak objasnit. \diamond

Nyní okomentujme obdržené výsledky. Zaprvé, hladový algoritmus v aktuálním nastavení zcela selhává, jak je vidět ze všech grafických výstupů. Podobně je na tom jednoduché prahování, které je sice schopné určit poměrně dobře oporu nábojů (viz sloupec TNR v tabulce 5.2), avšak produkuje méně kvalitní odhady než obyčejná AHK (viz obrázek 5.5).



Obrázek 5.6: Krabicové diagramy relativního vysvětleného rozptylu pomocí prvních dvou řádkových HK, tj. součet rozptylů vysvětlených prvními dvěma řádky HK dělený součtem rozptylů vysvětlených pomocí prvních dvou obyčejných HK. Vyšší hodnoty znamenají lepší chování daného algoritmu.

Zadruhé, S-PCA v tomto příkladě dává průměrné výsledky. Z obrázku 5.5 je vidět, že produkuje jen o něco málo lepší úhly mezi skutečným a odhadnutým podprostorem než obyčejná AHK. To však samo o sobě neznamená, že by výsledky S-PCA byly bezcenné. S-PCA stále nabízí o něco málo kvalitnější odhady než obyčejná AHK a proti obyčejné AHK přináší řídkost, čímž zjednodušuje interpretovatelnost obdržovaných výsledků. Otázkou je, nakolik cenná je tato lepší interpretovatelnost v případě, že odhadnutá opora není správná (S-PCA je v tomto ohledu horší než ST-PCA, viz nižší TNR v tabulce 5.2).

Konečně ADA-S-PCA a IRLS-S-PCA nabízejí v daném nastavení obrovské zlepšení proti S-PCA. Ve všech uvažovaných výstupech je ADA-S-PCA algoritmus druhý nejlepší a IRLS-S-PCA algoritmus úplně nejlepší.

Poznámka 5.5. Jistou disproporcí mezi našimi výsledky a výsledky Shena a Huang (2008) lze nejspíš objasnit tím, že my kvůli centrování matice \mathbf{X} řešíme ve skutečnosti složitější úlohu než Shen a Huang, jak jsme diskutovali v poznámce 5.4. Přesto však náš IRLS-S-PCA algoritmus dává lepší výsledky než všechny algoritmy uvažované v Shen a Huang (2008), jak si může zaujatý čtenář porovnat. \diamond

5.3 Přesná kovarianční matice

Nyní diskutujeme poněkud umělý příklad se známou kovarianční maticí převzatý z Zou a kol. (2006), který se stal standardním kritériem pro hodnocení S-PCA algoritmů. Jak uvidíme, kovarianční matice je v tomto příkladu pečlivě sestrojena tak, aby jednoduché prahování dávalo pořádné výsledky vůči složitějším algoritmům pro AHŘK.

Uvažujme následující model. Necht

$$f_1 \sim N(0,290), \quad f_2 \sim N(0,300), \quad f_3 = -0.3f_1 + \gamma f_2 + \epsilon,$$

kde f_1 a f_2 jsou vzájemně nezávislé a obě jsou nezávislé s chybovým členem $\epsilon \sim N(0,1)$. Veličiny f_1, f_2, f_3 jsou nepozorovatelné proměnné. Pozorovatelné veličiny X_1, \dots, X_{10} jsou dány předpisem

$$\begin{aligned} X_i &= f_1 + \epsilon_i, & i &= 1, \dots, 4, \\ X_i &= f_2 + \epsilon_i, & i &= 5, \dots, 8, \\ X_i &= f_3 + \epsilon_i, & i &= 9, 10, \end{aligned}$$

kde $\epsilon_i \sim N(0,1)$, $i = 1, \dots, 10$ jsou chybové členy nezávislé s f_1, f_2, f_3 a ϵ . Potom

$$\mathcal{X} = (X_1, \dots, X_{10})^\top \sim \mathcal{N}(0, \Sigma), \quad (5.3)$$

kde dopočítání matice Σ je jednoduché cvičení.

Poznámka 5.6. Přestože Zou a kol. (2006) nazývají tento model faktorovým modelem o třech faktorech, f_1, f_2 a f_3 nejsou nekorelované, což je základní podmínka faktorového modelu. Model lze snadno přepsat na faktorový model o dvou faktorech ve smyslu (1.11). Je jednoduché ověřit, že model (5.3) pro \mathcal{X} lze ekvivalentně psát jako

$$\mathcal{X} = \mathbf{W}\mathcal{F} + \epsilon \quad (5.4)$$

pro $\mathcal{F} = (f_1, f_2)^\top \sim \mathcal{N}(0, \mathbf{I}_2)$, $\epsilon \sim \mathcal{N}(0, \Psi)$, kde $\Psi = \text{diag}(1, \dots, 1, 2, 2) \in \mathbb{R}^{10 \times 10}$, a konečně

$$\mathbf{W} = \begin{pmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & -0.3 & -0.3 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & \gamma & \gamma \end{pmatrix}^\top \in \mathbb{R}^{10 \times 2}.$$

Jedná se tedy o normální faktorový model, který však není izotropní. Tuto poznámku uvádíme pouze proto, abychom zasadili model (5.3) do kontextu předchozích kapitol. Z hlediska úvah v tomto příkladu je výhodnější vnímat model tak, jak byl popsán nad touto poznámkou. \diamond

Zou a kol. (2006) diskutovali tento model pro $\gamma = 0.925$. Počet autorů, kteří tento jednoduchý model převzali, je překvapivě velký. Z námi citovaných pramenů například d'Aspremont a kol (2007), Shen a Huang (2008), Ma (2011) nebo Lu a Zhang (2012).

Diskutujeme proto nejprve případ $\gamma = 0.925$, jiné hodnoty koeficientu γ budeme uvažovat až později. Rozptyly nepozorovatelných proměnných f_1, f_2 a f_3 jsou po řadě 280, 300 a 283.3. Počty pozorovaných proměnných asociovaných s těmito

Tabulka 5.3: První dva (populační) náboje modelu 5.3 odhadnuté různými metodami: obyčejnou AHK (značeno PCA), jednoduchým prahováním (ST-PCA) a ostatními metodami. Ostatní metody jsou: S-PCA, ADA-S-PCA, IRLS-S-PCA a Greedy-PCA.

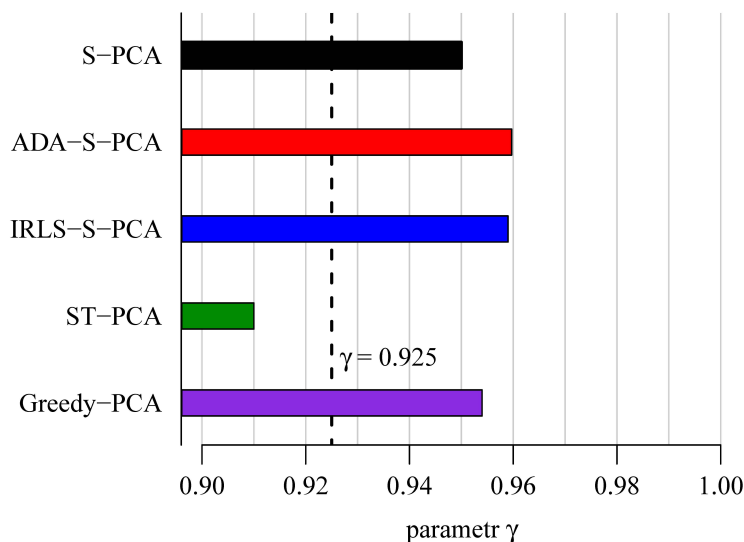
	PCA		ST-PCA		Ostatní	
	1st PC	2nd PC	1st PC	2nd PC	1st PC	2nd PC
X_1	-0.116	-0.478	0	0.5	0	0.5
X_2	-0.116	-0.478	0	0.5	0	0.5
X_3	-0.116	-0.478	0	0.5	0	0.5
X_4	-0.116	-0.478	0	0.5	0	0.5
X_5	0.395	-0.145	0	0	0.5	0
X_6	0.395	-0.145	0	0	0.5	0
X_7	0.395	-0.145	0.497	0	0.5	0
X_8	0.395	-0.145	0.497	0	0.5	0
X_9	0.401	0.010	0.503	0	0	0
X_{10}	0.401	0.010	0.503	0	0	0
vysvětlený rozptyl	60.0	39.6	40.9	39.5	38.8	38.6

proměnnými jsou po řadě 4, 4 a 2. Zároveň, první dvě klasické HK² vysvětlují 96 % celkového rozptylu. Na základě těchto údajů tvrdí Zou a kol. (2006) (a také další autoři uvažující tento model), že pro aproximaci kovarianční struktury by měly být použity první dvě HK, které by ideálně měly být obě řídké: první náboj by měl popsat skrytou proměnnou f_2 za použití proměnných X_5, \dots, X_8 a druhý náboj by měl popsat skrytou proměnnou f_1 za použití proměnných X_1, \dots, X_4 . Přesně tak se chová S-PCA, ADA-S-PCA, IRLS-S-PCA a Greedy-PCA (a také řada dalších algoritmů výše citovaných autorů), které prvními dvěma řídkými HK vysvětlují 80.4 % rozptylu, viz tabulka 5.3. Oproti tomu ST-PCA se nechá zmást pravděpodobně vysokou korelací mezi f_2 a f_3 , a produkuje sub-optimální výsledek, viz tabulka.

Korelace mezi f_2 a f_3 je kontrolována právě parametrem γ , který byl doposud nastaven na 0.925. Vyšší γ samozřejmě znamená vyšší korelaci a tudíž těžší úlohu. Vystává přirozená otázka, nakolik závisí selhání ST-PCA, a naopak úspěch ostatních metod na volbě tohoto parametru. Odhady nábojů jsme vyzkoušeli pro různé volby parametru $\gamma \in (0.9, 1)$. Na obrázku 5.7 je vyznačeno chování námi uvažovaných algoritmů v závislosti na volbě γ pomocí sloupcového diagramu. Jednotlivé pruhy vyznačují, pro které hodnoty γ dávají dané metody „optimální“ výsledky, tj. odhady nábojů jako v posledních dvou sloupcích tabulky 5.3. Pruhy končí v místě, kde daný algoritmus přestává být schopný dobře rozlišit mezi f_2 a f_3 a začne produkovat výsledky podobné výsledkům ST-PCA v tabulce 5.3. Například IRLS-S-PCA se chová „optimálně“ pro $\gamma < 0.958$. Jak vidíme z obrázku 5.7, hodnota $\gamma = 0.925$ volena ve všech člancích citovaných výše odpovídá hodnotě, kdy ST-PCA „selže“, zatímco všechny ostatní algoritmy dají optimální výsledky. Z diagramu je též vidět, že ADA-S-PCA a IRLS-S-PCA jsou nejrobustnější vzhledem k volbě γ .

Na závěr příkladu ještě diskutujeme dosavadní premisu, že „optimální“ odhad nábojů je dán přesně koeficienty v posledních dvou sloupcích matice 5.3. Vzhle-

²V tomto případě populační HK. Jsme v populační situaci, neprovádíme žádný náhodný výběr.



Obrázek 5.7: Chování různých algoritmů v závislosti na parametru γ . Pruhy vyznačují, pro jaké hodnoty γ se dané algoritmy chovají „optimálně“, tj. dávají odhady nábojů přesně tak, jak jsou udány v posledních dvou sloupcích tabulky 5.3.

dem k ekvivalentnímu přepisu modelu z poznámky 5.6, zejména tvaru matice \mathbf{W} , by mohlo být lepší ponechat jednomu z nábojů kardinalitu 6 namísto omezení obou kardinalit na 4. Pokud bychom například povolili druhému náboji kardinalitu 6, první náboj by použil proměnné $X_1, \dots, X_4, X_9, X_{10}$ a druhý proměnné X_5, \dots, X_8 , přičemž toto chování by bylo společné pro všechny metody. Navíc, robustnost všech metod vůči volbě γ by se zvýšila. Pro volbu $\gamma = 0.925$ by S-PCA a jeho modifikace dali totožné výsledky vysvětlující 98.1 % rozptylu prvními dvěma řádky HK, zatímco ST-PCA a Greedy-PCA by při stejném odhadu opory odhadly nenulové koeficienty nábojů trochu jinak, což by vedlo k nepatrně menšímu množství vysvětleného rozptylu. Tyto výsledky nebudeme uvádět.

5.4 Shrnutí

V této kapitole jsme prozkoumali chování čtyř algoritmů představených v kapitole 3 a IRLS-S-PCA algoritmu, který byl představen v kapitole 4, aplikováním těchto algoritmů na tři syntetické příklady. Dva z příkladů se zabývaly náhodným výběrem z normálního špičatého modelu, zatímco ve třetím příkladu jsme se snažili aproximovat známou kovarianční matici. Druhý a třetí příklad zároveň nabízejí možnost porovnání s existující literaturou.

Za souhrnný komentář stojí rozdílné chování algoritmů v prvním a druhém příkladu. Zatímco v prvním příkladě byl poměr počtu pozorování vůči počtu proměnných n/p roven $1/2$, v druhém příkladě byl tento poměr roven $1/10$. Z toho pramenící vyšší složitost druhého příkladu byla kompenzována výrazným zesílením signálu – zvětšením vlastních čísel příslušících nábojům. Zatímco v prvním příkladě nebyly mezi chováním jednotlivých algoritmů propastné rozdíly a zejména ST-PCA algoritmus fungoval poměrně dobře, v druhém příkladě se rozdíly mezi kvalitou výsledků jednotlivých algoritmů výrazně prohloubily a při použití jed-

nodušších metod ST-PCA a Greedy-PCA jsme obdrželi velmi špatné výsledky. Žádáme čtenáře o porovnání obrázků 5.3 a 5.5.

Ve všech příkladech se chovaly nejlépe algoritmy ADA-S-PCA a IRLS-S-PCA, které z hlediska všech uvažovaných měr výrazně překonaly svého předchůdce S-PCA. Přitom IRLS-S-PCA produkoval o něco lepší výsledky než ADA-S-PCA. Opět žádáme čtenáře o porovnání výstupů daných metod na obrázcích 5.3, 5.5, a případně také 5.4.

6. Analýza reálných dat: hlasování poslanců

V této kapitole aplikujeme diskutované algoritmy pro analýzu řídkých hlavních komponent na reálná data, konkrétně hlasování poslanců v Poslanecké sněmovně (dolní komoře) Parlamentu České republiky v roce 2015. Rok 2015 byl vybrán z důvodů jednoduchosti a aktuálnosti. Je jediným rokem vládnutí současné vlády, ve kterém nedošlo k žádné změně v rozložení sil v parlamentu, jako je například střídání poslanců v rámci politické strany, změny příslušnosti poslanců k politickým stranám nebo vznik, zánik, případně rozštěpení politické strany.

V politologii je (jako v mnoha jiných vědních odvětvích) AHK velmi oblíbená zejména z důvodu vizualizace. Čtenáře odkazujeme na Clintona a kol. (2004) pro přehledné shrnutí statistických metod používaných v politologii. Pokud se díváme na jednotlivá hlasování jako na proměnné, vektory hlasování příslušné jednotlivým poslancům mají vysokou dimenzi. Tyto vektory zachycují chování poslanců při hlasováních. Při projekci těchto vektorů do hlavního podprostoru dimenze 2 nebo 3 lze očekávat viditelné shlukování poslanců podle politické příslušnosti (tj. členství v jednotlivých stranách). Pokud budeme předpokládat, že daná projekce věrně zachytává chování jednotlivých poslanců, můžeme pomocí výsledného dvourozměrného obrázku za použití dodatečné informace o politické příslušnosti poslanců odpovědět na zajímavé otázky, jako jsou například:

- Které politické strany jsou homogenní?
- Které politické strany mají podobnou agendu a které naopak stojí proti sobě?
- Kteří poslanci jsou odlehlými pozorováními v rámci politických stran?

Na předchozí otázky dávají odpovědi jak AHK tak AŘHK (za použití libovolného algoritmu). Nicméně, jak uvidíme, odpovědi jsou do jisté míry odlišné v závislosti na použité metodě. Pomocí IRLS-S-PCA algoritmu obdržíme výsledky, které se velmi dobře shodují s obecně známým stavem české politické scény.

Na rozdíl od AHK, AŘHK přirozeně provádí výběr důležitých proměnných, v tomto případě aktů¹, čímž poskytuje dodatečnou informaci. Otázka, kterou jsme si kladli od samého počátku analyzování hlasovacích dat a která do značné míry ovlivnila celkový postoj celé této práce, je následující:

- Lze využít výběr proměnných (hlasování), kterou poskytuje AŘHK, k automatizaci procesu vytváření volebních kalkulaček²?

Předesíláme, že odpověď na zmíněnou otázku je pouze *částečně ano*. Jednou z možných volebních kalkulaček je projekce poslanců do hlavního podprostoru dimenze 2 a následná projekce uživatele do tohoto podprostoru na základě hlaso-

¹Jednomu proběhlému hlasování v parlamentu říkáme akt. Akt se nemusí nutně týkat zákona nebo pozměňovacího návrhu, může jít o různá procedurální hlasování.

²Takzvaná *volební kalkulačka* je souhrnné označení pro nástroj, pomocí něž může uživatel zjistit, jaké politické strany a jací poslanci nejlépe reprezentují jeho názory.

vání uživatele „nanečisto“. Volební kalkulačky jsou velmi populární zejména na sociálních sítích s blízcími se volbami³. Ačkoliv metody AŘHK mohou být nápomocné při výběru „důležitých“ zákonů, samotná tvorba volební kalkulačky se neobejde bez expertního vhledu do problematiky.

6.1 Popis dat

Na webových stránkách Poslanecké sněmovny⁴ jsou dostupná data ve formě různých datových tabulek. Tyto tabulky obsahují informace o poslancích, o jejich členství v politických stranách a různých výborech, jejich hlasování v jednotlivých aktech, příslušnosti aktů ke sněmovním tiskům a schůzím poslanecké sněmovny a mnoho dalších informací. Všechna data jsou dostupná zpětně až do roku 1993, kdy byla založena Česká republika. Data jsou celkově ve velmi dobrém stavu a podle našeho názoru se nemá smysl zabývat případnými chybami.

Pro nás bude nejdůležitější tabulka „hl_poslanec“, ve které jsou ve třech sloupcích (číslo poslance, číslo aktu, jednotlivý hlas) zaznamenány hlasování poslanců v daných aktech. Jedinou dodatečnou informací, kterou využíváme výhradně za účelem grafického zobrazení výsledků, je příslušnost poslanců k politickým stranám. Tuto informaci poskytuje tabulka „poslanec“. Detailní popis tabulek lze nalézt na webových stránkách poslanecké sněmovny.

Původní data jsme transformovali do matice $\mathbf{X} \in \mathbb{R}^{n \times p}$ s prvky

$$x_{ij} = \begin{cases} 1, & \text{pokud } i\text{-tý poslanec hlasoval } \textit{ano} \text{ v } j\text{-tém úkonu,} \\ -1, & \text{pokud } i\text{-tý poslanec hlasoval } \textit{ne} \text{ v } j\text{-tém úkonu,} \\ 0, & \text{jinak.} \end{cases}$$

Počet poslanců v parlamentu je $n = 200$, počet aktů za rok 2015 byl $p = 1837$. Tedy je $\mathbf{X} \in \mathbb{R}^{200 \times 1837}$. Vektor hlasování i -tého poslance je reprezentován vektorem $\mathbf{x}^{(i)} \in \{-1, 0, 1\}^{1837}$, $i = 1, \dots, 200$. Pro potřeby AHK a všech algoritmů pro AŘHK matici \mathbf{X} opět sloupcově centrujeme a jinak neupravujeme.

Pokud poslanec nehlasoval pro nebo proti v daném aktu, kódujeme jeho hlasování nulou. Přitom nerozlišujeme, zda se poslanec formálně zdržel hlasování, nebo nehlasoval z jakékoliv jiné příčiny. Důvody jsou následovné. Zaprvé, od novely jednacního řádu 90/1995 Sb. z roku 2014 není v datech rozlišeno, zda se přítomný poslanec formálně zdržel hlasování, nebo zda z nějakého důvodu nehlasoval, rozlišuje se pouze, zda byl přítomen, nebo nepřítomen (s omluvou, nebo bez omluvy). Zadruhé, většina aktů vyžaduje pro schválení nadpoloviční většinu přítomných poslanců. Z prvního důvodu nelze přistupovat k hlasováním či nehlasováním přítomných poslanců jako ke chybějícím pozorováním. Z druhého důvodu by pak bylo zvláštní přistupovat k případným hlasům chybějících poslanců jako k chybějícím hlasováním. Celkově, poslanci jsou si taxonomie dobře vědomi a v případě důležitých hlasování s ní běžně kalkulují. Nami zvolené kódování se nám tudíž zdá nejrozumnější.

³Tvůrcem kvalitních datových analýz a s tím spojených volebních kalkulaček pro Parlament České republiky je občanské sdružení kohovolit.eu. Členové rady tohoto sdružení, Kamil Gregor a Michal Škop, nám poskytli základní vhled do problematiky, za což jim tímto děkujeme.

⁴<https://www.psp.cz/sqw/hp.sqw?k=1300>

Tabulka 6.1: Politická situace v České republice v roce 2015.

	Strana	Počet poslanců	Politická pozice
<i>Koalice</i>	ČSSD	50	levice-středolevice
	ANO	47	střed-středopravice
	KDU-ČSL	14	střed
<i>Opozice</i>	ODS	16	pravice
	Úsvit	26	pravice
	TOP 09	14	středopravice-pravice
	KSČM	33	levice (krajní)

Politickou situaci v České republice (v daném roce 2015) lze popsat následovně. Koalice složená ze tří politických stran tvoří majoritní vládu, kterou lze charakterizovat jako středovou na pravo-levé škále. Opozice sestávající ze zbývajících čtyř stran se rozděluje na pravicovou a levicovou. Detailnější popis politického rozložení sil je dán v tabulce 6.1.

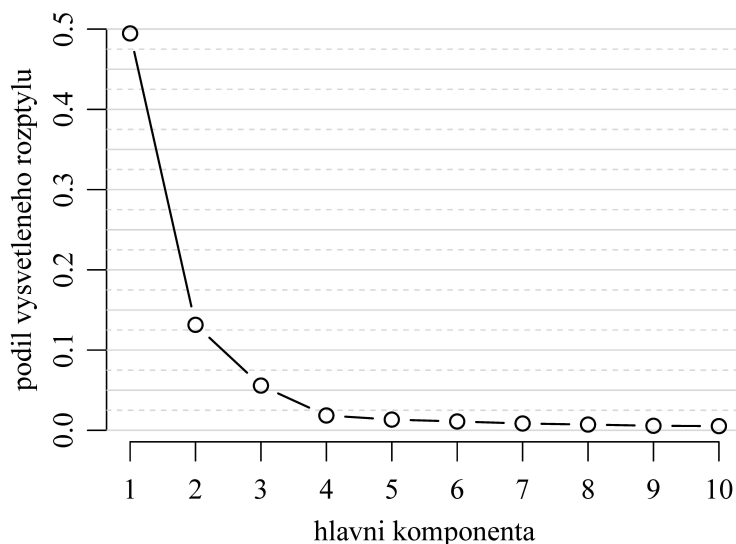
6.2 Volba počtu komponent a kardinalit nábojů

Pro AHK je nutné zvolit počet hledaných nábojů K , tj. dimenzi podprostoru, do kterého chceme naše 1837-rozměrná data projektovat. Pro AŘHK je navíc nutné zvolit kardinality s_1, \dots, s_K pro daný počet nábojů.

Běžným postup ve studiích podobného typu (viz např. Zhang, 2011, pro několik různých příkladů) je následovný. Nejprve je zvoleno K na základě druhé strategie z konce sekce 1.1. Tj. počet komponent je zvolen tak, aby $(K + 1)$ -ní komponenta vysvětlovala „zanedbatelné“ množství rozptylu. Kardinality nábojů jsou následně voleny tak, aby každá řídka HK vysvětlovala relativně vysokou porci rozptylu příslušné obyčejné HK, například 90 %.

Na obrázku 6.1 je zakreslen podíl rozptylu vysvětlený prvními deseti HK. Z obrázku se jasně nabízí volit počet komponent $K = 2$ nebo $K = 3$. S volbou kardinalit tak, aby náboje vysvětlovaly vysokou porci rozptylu příslušné obyčejné HK, však nesouhlasíme z důvodů popsaných v poznámce 3.4. Navíc zmíněná strategie typicky vede k přehnaně vysoké volbě kardinalit. Například my bychom v případě hlasovacích dat, při snaze vysvětlit řídkými HK jen 80 % rozptylu obyčejných HK, skončili s kardinalitami v řádu stovek. Z výše uvedených důvodů zastáváme názor, že volba K a s_1, \dots, s_K by měla být provedena na základě expertního názoru, nikoliv na základě dat.

Uvažujme nyní faktorový model a jeho spojitost s AHK, jak jsme ji diskutovali v sekci 1.4. Tím nemyslíme, že bychom uvažovali pravděpodobnostní model pro naše data, neboť by bylo samozřejmě pochybné uvažovat, že naše diskrétní data mají normální rozdělení. Jak radí Jolliffe (2002), na AHK nenahlížíme jako na pravděpodobnostní model, nýbrž jako na aproximaci dat nízké dimenze. Přesto však volná představa nepozorovaných faktorů v pozadí může být výhodná pro interpretaci.



Obrázek 6.1: Podíl celkového rozptylu vysvětlený každou z prvních deseti obyčejných HK pro data hlasování poslanců.

Z pohledu faktorového modelu, uvažování K hlavních komponent implicitně předpokládá, že chování každého poslance při hlasování může být – až na chybu specifickou pro daného poslance – popsáno jako lineární kombinace dvou skrytých faktorů. Při zamyšlení, jaké faktory mohou mít vliv na to, jak konkrétní poslanec hlasuje v konkrétním aktu, se ihned nabízí dva faktory: příslušnost poslance ke koalici či opozici a pozice poslance na pravo-levé škále. Oba tyto faktory úzce souvisí s příslušností poslance k politické straně. Případný třetí faktor je již méně jasný. Náš osobní (a nutno podotknout laický) názor před provedením analýzy dat byl, že by mohlo jít buď o pro- či proti-evropský postoj, případně o různá křídla soupeřící o vliv uvnitř politických stran. Tyto úvahy nás vedou k uvažování dvou hlavních komponent, výsledky pro trojici HK budeme také diskutovat pro porovnání.

Bez ohledu na počet komponent rozeberme nyní možnou volbu kardinalit s_1, \dots, s_K . Z pohledu faktorového modelu se snažíme popsat faktory v doméně aktů. Otázka tedy zní, kolik aktů je třeba k popsání daných faktorů. Všimněme si, že z tohoto pohledu je předpoklad řídkosti velmi rozumný, jistě není třeba tisíců hlasování, abychom mohli popsat příslušnost poslance k vládě či koalici nebo jeho pozici na pravo-levé škále. Očekáváme, že počty aktů v řádu desítek pro každý faktor by mohly být dostatečné. Vzhledem k tomu, že výroční analýzy hlasování poslanců prováděné experty⁵ typicky zahrnují 20 až 35 aktů daného roku, rozhodli jsme se nakonec pro volby $K = 2$, $s_1 = 20$ a $s_2 = 15$. Z praktického hlediska poslance projektujeme do roviny (tak jak je v politologii zvykem) a počet zákonů potřebných k popisu této roviny volíme jako 35, což je počet parametrů, které lze spolehlivě odhadnout z $n = 200$ pozorování, a zároveň je to číslo, na jehož základě je rozumné vytvořit volební kalkulačku, neboť nechat uživatele hlasovat na nečisto o 35 aktech není přehnané.

⁵Opět odkazujeme na web kohovolit.eu.

6.3 Výsledky

Obrázek 6.2 znázorňuje projekci dat hlasování poslanců do hlavního podprostoru dimenze $K = 2$, který je získaný obyčejnou AHK (panel (a)) a různými algoritmy pro AŘHK (panely (b)-(f)).

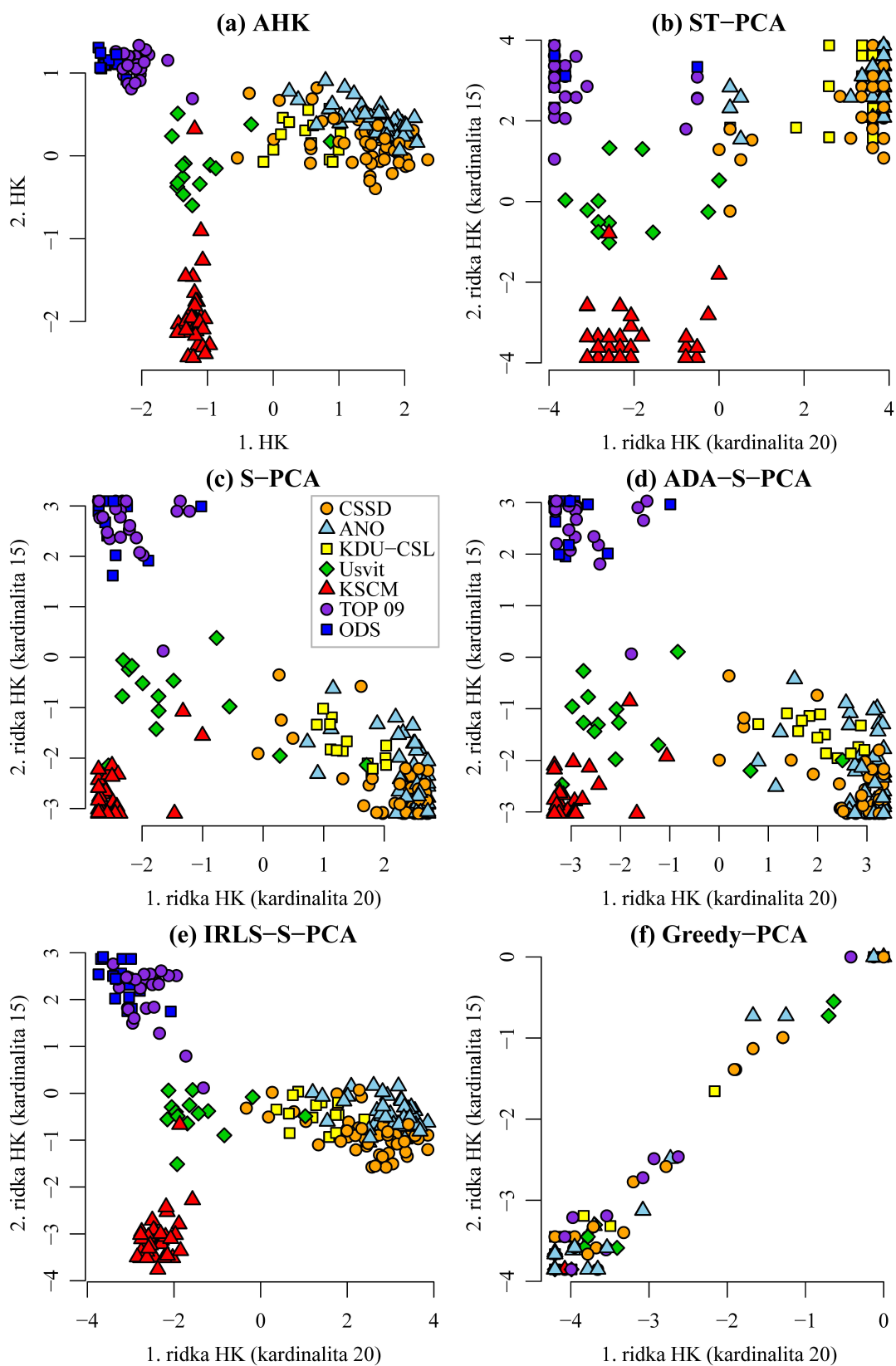
Nejhůře interpretovatelné výsledky dává bezesporu Greedy-PCA, viz panel (f) obrázku 6.2. Vzhledem k tomu, že projektované body leží na diagonále obrázku, se domníváme, že pozorujeme praktický příklad selhání deflačních metod, kdy se první a druhý řídký náboj ve skutečnosti snaží odhadnout stejný směr (obyčejná první HK je v tomto příkladu velmi dominantní, viz obrázek 6.1). Pro srovnání prezentujeme na obrázku 6.3 výsledky obdržené pomocí S-PCA a IRLS-S-PCA sekvenčně. To znamená, že nejprve nalezneme první řídký náboj izolovaně pomocí dané metody, pak provedeme Hotellingovu deflací a následně spočteme druhý náboj, opět izolovaně pomocí dané metody. Všimněme si, že v případě použití deflace je S-PCA algoritmus podobně jako Greedy-PCA zcela neschopný rozlišit druhou komponentu od první, proto je také výsledná projekce diagonální. IRLS-S-PCA se chová jen o něco málo lépe, nicméně jím obdržené výsledky jsou také znehodnoceny deflací. Výsledky obdržené pomocí Greedy-PCA jsou ještě horší než pomocí sekvenční S-PCA, jelikož Greedy-PCA algoritmus není na rozdíl od S-PCA schopen ani pomocí první komponenty odlišit koalici od opozice.

Zbylé algoritmy pro AHŘK, jejichž výsledky jsou zobrazeny v panelech (b)-(e) obrázku 6.2, dávají kvalitativně stejné výsledky jako AHK; všechny tyto výsledky jsou výrazně lepší než výsledky Greedy-PCA. Ve všech případech první HK rozlišuje mezi koalicí a opozicí a druhá mezi pravicí a levicí. Toto je samozřejmě očekávaný a velmi dobře interpretovatelný výsledek. V následujících odstavcích budeme diskutovat výsledky všech metod s výjimkou Greedy-PCA souhrnně, přičemž naše úvahy budou nejlépe viditelné na výsledcích obyčejné AHK a IRLS-S-PCA na panelech (a) a (e) obrázku 6.2.

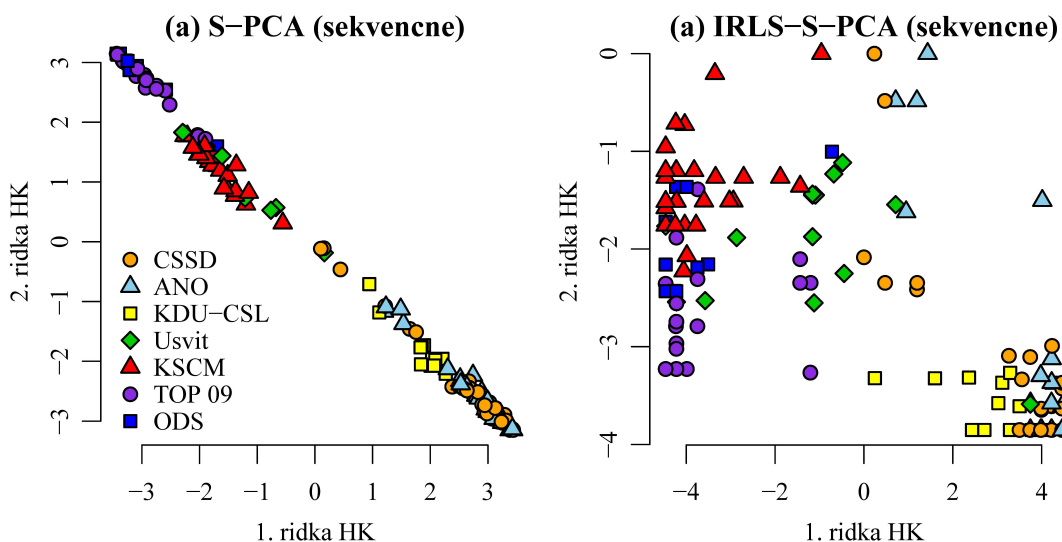
Na základě grafických výstupů lze dělat i další závěry. Strana *Úsvit* leží ve všech případech uprostřed obrázku, trochu blíže opozici, což lze odůvodnit dvěma způsoby. Buď se poslanci této strany chovají oproti členům ostatních stran nekonzistentně nebo se prostě vymykají zažitým vzorcům a bylo by třeba více hlavních komponent k vysvětlení jejich chování. Také to může být způsobeno případnou nízkou účastí poslanců *Úsvitu* při hlasováních. Toto ještě budeme diskutovat později.

Dále, na všech obrázcích je vidět, že *KDU-ČSL* je posunuta nejméně do středu ze tří vládních stran. To samozřejmě souhlasí s tím, že *KDU-ČSL* je nejslabším členem koalice. Naopak vzájemná pozice *ANO* a *ČSSD* naznačuje, že tyto dvě strany jsou rovnocennými partnery a že *ANO* má lehce pravicovější agendu.

Konečně *TOP 09* a *ODS* jsou sice rozlišitelné (alespoň v případě AHK a IRLS-S-PCA), ale s velmi podobnou agendou. Naopak *KSČM* je jakožto strana na kraji politického spektra nejméně izolovaná. K souhrnné interpretaci panelů (a)-(e) obrázku 6.2 ještě poznamenejme, že nejméně středový poslanec *TOP 09*, dva nejméně středoví poslanci *KSČM* a nejméně provládní poslanec *Úsvitu*, jsou shodní na všech panelech.



Obrázek 6.2: Projekce hlasování poslanců do hlavního podprostoru dimenze 2 pomocí AHK a různých algoritmů pro AŘHK.



Obrázek 6.3: Projekce hlasování poslanců do hlavního podprostoru dimenze 2 pomocí S-PCA a IRLS-S-PCA, avšak první a druhý náboj jsou nalezeny sekvenčně s využitím Hotellingovy deflace mezi hledáním prvního a druhého řídkého náboje.

Na druhou stranu, panely (a)-(e) obrázku 6.2 jsou rozhodně odlišné, pokouší-li se čtenář o jemnější rozlišení jednotlivých poslanců. Výsledky ST-PCA, S-PCA a ADA-S-PCA nejsou tak hladké, jako je tomu v případě obyčejné AHK a IRLS-S-PCA, a také nabízejí mnohem horší rozlišení mezi *ČSSD* a *ANO* nebo mezi *ODS* a *TOP 09*.

Celkově vzato, výsledky IRLS-S-PCA zachycené na panelu (e) obrázku 6.2 zachycují obecné povědomí o české politické scéně téměř dokonale. V celkově středové vládě se nachází pravicovější *ANO*, levicovější *ČSSD* a z vládního hlediska lehce upozaděná *KDU-ČSL*. Opozice se jasně rozděluje na pravicovou (*ODS* a *TOP 09*) a levicovou (*KSČM*), zatímco zařazení *Úsvitu* je problematické. Podobnou situaci zachycuje jen o něco málo hůře obyčejná AHK. Žádáme čtenáře o porovnání. Nicméně IRLS-S-PCA jako metoda AŘHK zachycuje situaci s použitím pouze $s_1 + s_2 = 35$ aktů na rozdíl od obyčejné AHK, která využívá všech $p = 1837$ aktů. Získaná řídkost je dodatečnou informací, která může být velice cenná.

Například, pokud by čtenář chtěl zařadit sám sebe do grafu mezi poslance, v případě AHK by musel hlasovat „nanečisto“ o 1837 aktech nebo by musel ručně vybrat nějaké „důležité“ akty a spokojit se s nepřesnou projekcí. Vybrání aktů ručně by samozřejmě dalo nemalou práci. Naopak, v případě IRLS-S-PCA již byly vybrány „důležité“ akty, a pokud by čtenář hlasoval „nanečisto“ o těchto aktech, jeho projekce do roviny by byla přesná. Ačkoliv situace vypadá velice atraktivně z hlediska možné automatizace tvorby volebních kalkulaček, komplikuje se tím, že akty vybrané pomocí metod AŘHK nejsou pro laika snadno interpretovatelné. Uvedme několik příkladů aktů, které byly vybrány všemi metodami AŘHK (s výjimkou Greedy-PCA):

- akt č. 60 403 – hlasování o programu schůze;
- akt č. 61 973 – pozměňovací návrh k vládnímu návrhu zákona o státním rozpočtu ČR na rok 2016;

- akt č. 62 034 – vládní návrh zákona o státním rozpočtu ČR na rok 2016 (finální hlasování).

Je jistě pozitivní, že konečné hlasování o státním rozpočtu bylo vybráno jako důležité všemi AŘHK metodami (kromě Greedy-PCA). Na druhou stranu, v případě aktu č. 60 403 je třeba prohledat stenografické záznamy⁶, aby zjistil, že šlo o návrh poslance Blažka, aby byl jistý bod programu zařazen jako první bod schůze. O jaký bod šlo, je nutné vyčíst ze sněmovních tisků. Zjistit, proč byl právě tento akt důležitý, je však pro laika velmi obtížné. Patrně z těchto důvodů neexistuje v případě Poslanecké sněmovny jednoduchá taxonomie zachycující význam jednotlivých aktů.

Tvorbu volebních kalkulaček tedy nelze zcela automatizovat kvůli různým skrytým agendám na pozadí jednotlivých aktů, k jejichž rozšifrování je potřebná expertní znalost problematiky. Na druhou stranu, zákony vybrané pomocí metod AŘHK mohou sloužit expertovi jako cenné vodítko, na jaké sněmovní tisky má smysl se soustředit. Lze očekávat, že v mnoha případech se expertovi podaří rozšifrovat skrytou agendu za vybranými akty.

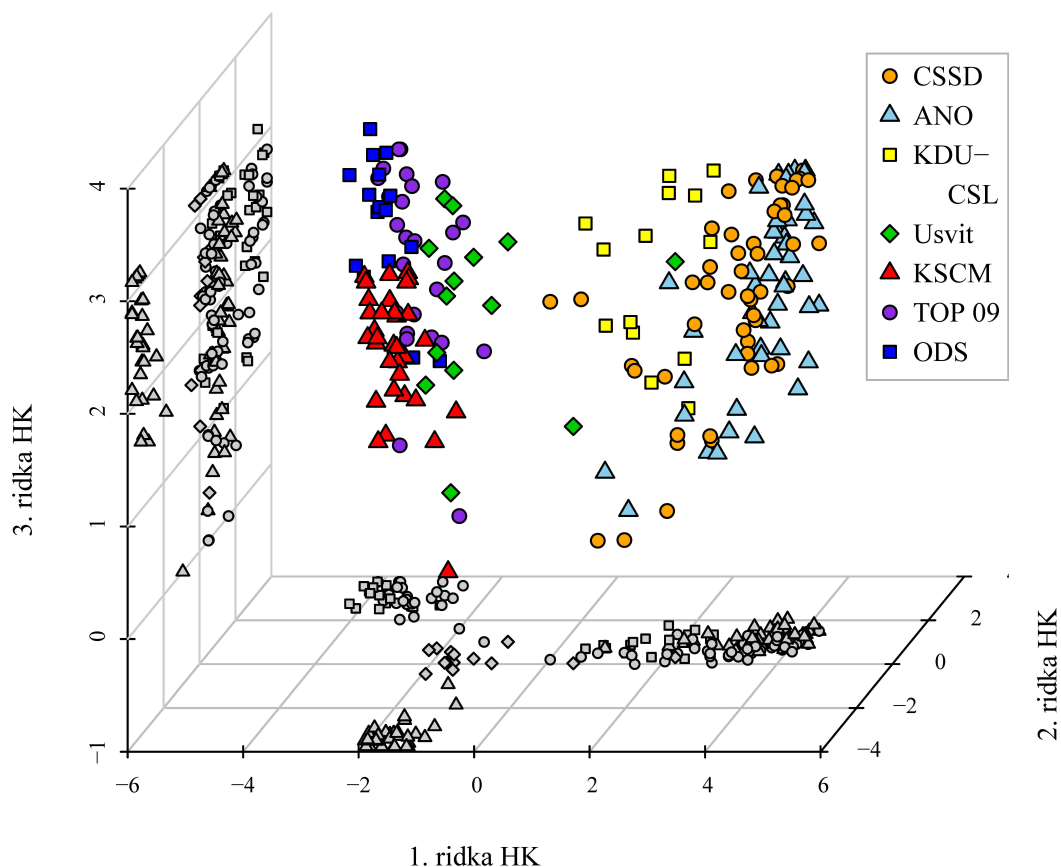
Získaná řídkost může být velmi cenná i z hlediska jiných úvah. Vraťme se například k poslancům *Úsvitu*. Poslanci této politické strany byli považováni za jedny z nejpracovitějších díky jejich vysoké sněmovní docházce. Na druhou stranu, přítomnost ve sněmovně nebo i na hlasování ještě neznámá, že se poslanci aktivně zapojují do významných hlasování. Pouze informativně poznamenejme, že poslanci *Úsvitu* nehlasovali ani pro ani proti danému aktu ve zhruba 40 % případů, což je o 5 % vyšší číslo než celkový sněmovní průměr. Podíváme-li se na důležité zákony vybrané pomocí IRLS-S-PCA, zjistíme, že rozdíl se ještě prohlubuje. V těchto 35 aktech poslanci *Úsvitu* nehlasovali ani pro ani proti ve 42 % případů, zatímco sněmovní průměr byl 42 %. Ačkoliv tedy docházka poslanců *Úsvitu* může být vysoká, v důležitých hlasováních se mnohem častěji než ostatní poslanci zdržují hlasování, případně nehlasují vůbec.

6.3.1 Tři hlavní komponenty

V této sekci ukazujeme pro porovnání výsledky získané uvažováním trojice hlavních komponent. Soustředíme se přitom pouze na výsledky IRLS-S-PCA algoritmu pro $K = 3$ a volbu kardinalit $s_1 = 20$, $s_2 = 15$ a $s_3 = 10$. Tedy na projekci 1837-rozměrných bodů do 3-rozměrného hlavního podprostoru.

3D projekce pomocí IRLS-S-PCA je vyobrazena na obrázku 6.4. Pro umožnění lepší orientace v prostoru zobrazujeme černobíle projekce do rovin tvořených první a druhou a také druhou a třetí komponentou. Projekce ležící ve spodní části obrázku 6.4 je tedy verzí panelu (e) z obrázku 6.2. Zdůrazněme, že tato projekce není přesně stejná jako projekce znázorněná na panelu (e) z obrázku 6.2. Ač se nám to nemusí líbit, obecně nelze čekat, že by se prvních K řídkých HK hledaných samostatně zcela shodovalo s prvními K řídkými HK obdrženy jako část většího

⁶Stenografické záznamy jsou taktéž dostupné na webových stránkách Poslanecké sněmovny. Např. pro akt č. 60 403 na adrese <http://www.psp.cz/eknih/2013ps/stenprot/025schuz/s025151.htm#h80>.



Obrázek 6.4: Projekce hlasování poslanců do hlavního podprostoru dimenze 3 pomocí IRLS-S-PCA algoritmu. Pro lepší přehlednost je graf rozšířen a jsou dány také dvou-dimenzionální projekce do rovin, které jsou tvořeny první a druhou a také druhou a třetí řídkou HK.

počtu $\tilde{K} > K$ řídkých HK. Rozdílnosti konkrétně pro S-PCA algoritmus stručně zkoumali Shen a Huang (2008). Nicméně, v případě dobré metody pro AŘHK by se prvních několik řídkých HK nemělo výrazně lišit v závislosti na tom, kolik HK bylo hledáno. To platí v případě IRLS-S-PCA, jak můžeme ověřit porovnáním dolní části obrázku 6.4 s panelem (e) z obrázku 6.2.

Z 3D projekce na obrázku 6.4 je vidět, že třetí řídka HK se pokouší zachytit rozptýl v rámci politických stran. Těžko odhadovat, čím může být tento rozptýl způsobený. Nicméně, i v 3D projekci tvoří jednotlivé strany homogenní shluky poslanců.

S využitím třetí dimenze lze lépe odpovědět na otázky typu:

- Kteří poslanci jsou odlehlými pozorováními z pohledu svých stran?
- Jaká je pozice lídrů vůči jejich stranám?

Tímto směrem se již nechceme vydávat, ale pro zajímavost se pouze zaměříme na konkrétního poslance *TOP 09*, který má na obrázku 6.4 nejnížší hodnotu třetí řídké HK. Máme na mysli poslance, který je společně s jedním poslancem *Úsvitu* osamocen ve spodní části obrázku 6.4. Tímto poslancem je překvapivě

Karel Schwarzenberg, tehdejší předseda *TOP 09*. Pokud bychom se na 3D projekci dívali i z jiných úhlů, zjistili bychom, že Karel Schwarzenberg je pravděpodobně nejdolehlejším pozorováním mezi všemi poslanci. Zatímco ostatní politické strany jsou více či méně koncentrovány kolem svých předsedů, *TOP 09* je koncentrována spíše kolem svého 1. místopředsedy Miroslava Kalouska. Tyto výsledky v samotné práci neuvádíme, vizualizace však lze získat spuštěním kódu, který je k nalezení v elektronické příloze.

6.4 Shrnutí

V této kapitole jsme analyzovali sadu reálných dat týkající se hlasování poslanců a porovnávali jsme výsledky obdržené různými algoritmy, které v naší práci uvažujeme. Greedy-PCA produkoval jednoznačně nejhorsí výsledky. Do jisté míry je to tím, že při relativně restriktivních požadavcích na kardinalitu nábojů a v případě komplexních dat, jako jsou tato, se ukazují deflační metody jako velmi nespolehlivé. To jsme ukázali porovnáním kvality výstupů získaných použitím klasické S-PCA a IRLS-S-PCA, které samy o sobě deflační metody nepotřebují, a sekvenčních verzí S-PCA a IRLS-S-PCA, které jsou přinuceny deflační metody používat. V sekvenčních verzích používáme dané algoritmy k postupné extrakci řídkých nábojů, přičemž jednotlivé extrakce jsou proloženy deflací. Sekvenční metody daly v tomto příkladě výrazně zhoršené výsledky.

Výsledky získané klasickou metodou AHK a ostatními metodami pro AŘHK (kromě Greedy-PCA) vedly k podobným závěrům. Nicméně výsledky ST-PCA, S-PCA a ADA-S-PCA nebyly dostatečně hladké, a celkově byly mnohem hůře interpretovatelné než výsledky obdržené pomocí klasické AHK či pomocí IRLS-S-PCA.

Přestože uvažovaná data jsou mnohorozměrná s vyšším počtem proměnných než pozorování ($n = 200$ a $p = 1837$), nezdá se, že by obyčejná AHK trpěla prokletím vysoké dimenze způsobem, jaký jsme pozorovali v naší simulační studii v předchozí kapitole. Předpokládaná řídkost tak v tomto případě nemá za následek zlepšení kvality odhadů, ale zlepšení celkové interpretovatelnosti. Díky tomu, že metody AŘHK produkují řídké odhady, dochází implicitně k výběru důležitých zákonů. Toto je cenná informace sama o sobě. Například první HK (obyčejná i řídká) rozlišuje mezi koalicí a opozicí. V případě řídké HK lze zároveň říci, které zákony nejlépe odrážejí kontroverzi mezi koalicí a opozicí. IRLS-S-PCA algoritmus je schopen produkovat výsledky přinejmenším srovnatelné s obyčejnou AHK při poměrně striktní požadované volbě kardinalit řídkých nábojů.

Konečně, zajímali jsme se o to, zda by bylo možné využít výběr proměnných k automatizaci tvorby volebních kalkulaček. Projekce založené na malém počtu zákonů jsou z pohledu volebních kalkulaček jistě žádoucí, jelikož uživatel typicky není ochotný nanečisto hlasovat o nadměrném počtu aktů. Nicméně, ačkoliv hlavní podprostor nalezený pomocí IRLS-S-PCA dobře popisuje politickou situaci, vybrané akty nejsou pouze důležité zákony, ale také různá procedurální hlasování, kterým nelze přiřadit skutečný význam bez dostatečné expertní znalosti problematiky. Tvorbu volebních kalkulaček proto podle našeho názoru nelze plně automatizovat.

Diskuze

Existuje celá řada algoritmů pro AŘHK, které jsme neuvažovali. Chování některých z těchto algoritmů je do značné míry teoreticky prostudováno. Prominentním zástupcem AŘHK algoritmů je z tohoto pohledu SDP algoritmus d'Aspremonta a kol. (2007). My jsme se zaměřili na S-PCA algoritmus, pro který si nejsme vědomi žádných teoretických záruk. Domníváme se, že vzhledem k nekonvexnímu alternujícímu schématu není například důkaz konzistence S-PCA algoritmu dosažitelný v libovolném netriviálním modelu. Tento fakt byl ostatně jednou z našich motivací pro nahrazení konvexní ℓ_1 -penalizace, která je přítomná v S-PCA algoritmu, nekonvexní penalizací v kapitole 4. Ač by bylo jistě vhodné lépe teoreticky prozkoumat řešení dané S-PCA algoritmem, podobné teoretické studie jsou v literatuře dostupné zejména pro metody, které extrahují řídké náboje sekvenčně; k nim patří i SDP algoritmus. Jak jsme však diskutovali a ukázali na praktickém příkladu v kapitole 6, sekvenční algoritmy jsou nuceny využívat jisté formy deflace a např. běžně užívaná Hotellingova deflace může snadno vést ke katastrofálním výsledkům. SDP algoritmus jsme sice nepředstavili, ale v různých simulačních studiích nijak výrazně nepřekonává S-PCA ani v kvalitě prvního nalezeného náboje. Tudíž lze s relativní jistotou očekávat, že jím získané výsledky v reálném příkladu z kapitoly 6 budou stejně špatné jako pro sekvenční S-PCA.

V naší práci jsme se příliš nezabývali volbou ladících parametrů pro studované algoritmy. Všechny metody však umožňují volit přímo požadované kardinality nábojů, což je podle nás velmi žádoucí z praktického hlediska. Na příkladu z kapitoly 6 jsme viděli, že v praxi může mít uživatel s expertním vhladem do dané problematiky, případně předem daným cílem, velmi dobrou představu o tom, jak kardinality volit. Naopak pro takového uživatele může být těžko uchopitelná role samotných ladících parametrů. Přesto však S-PCA algoritmus vyžaduje volbu jednoho ladícího parametru a IRLS-S-PCA vyžaduje volbu dvou, navíc ještě variabilních ladících parametrů. My jsme v našich příkladech vždy zadávali požadované kardinality nábojů a ladící parametry jsme nechávali tak, jak jsou nastaveny defaultně v jednotlivých algoritmech. Pokud bychom chtěli volit parametry lépe než na základě pravidel palce, které se ukázaly být funkční v různých numerických studiích, museli bychom sáhnout k nějakému způsobu převzorkování. Domníváme se, že metody popsané Mazumderem a kol. (2011) by k tomu měly být vhodné.

Jelikož S-PCA algoritmus převádí úlohu AŘHK na iterativní řešení penalizovaných regresních problémů, přirozeně se nabízí vyzkoušet v tomto schématu mnoho dalších způsobů penalizace. Během našich simulačních studií jsme studovali kromě adaptivní lasso metody (vedoucí k ADA-S-PCA) ještě další procedury založené na ℓ_1 -penalizaci, zejména tzv. uvolněné lasso (*relaxed lasso*, Meinshausen, 2007) a vícekrokové adaptivní lasso (Zou a Li, 2008). Výsledky získané těmito metodami nejsou v naší práci uvedeny, budeme je zde jen krátce diskutovat. Uvolněné lasso spočívá v tom, že z obyčejného lasso odhadu je ponechán pouze odhad opory a odhady samotných koeficientů jsou získány MNČ odhadem podproblému s danou oporou, tj. jedná se vlastně o „variační opravu“ popsanou v sekci 3.2. Tu lze samozřejmě použít v případě všech uvažovaných algoritmů. V případě hledání více než jednoho náboje vede „variační oprava“ k mnohým problémům, a my

jsme navíc nepozorovali zlepšení kvality odhadů ani v případě hledání jednoho náboje. Podobně, víceřadkové adaptivní lasso iterativně opakuje proceduru adaptivní lasso metody a vlastně se pokouší aproximovat ℓ_0 -normu pomocí převažování ℓ_1 -normy analogickým způsobem, jako se penalizace ze sekce 4.1 pokouší aproximovat ℓ_0 -normu pomocí převažování ℓ_2 -normy. Opět pro nás bylo překvapením, že jsme touto víceřadkovou procedurou nebyli schopni dosáhnout celkově lepších výsledků než pomocí ADA-S-PCA. Vysvětlujeme si to tím, že převažování ℓ_1 -normy je (vzhledem k vynucování řídkosti v každém kroku) mnohem náchylnější na volbu ladících parametrů než převažování ℓ_2 -normy, a proto námi používaná pravidla palce pro volbu ladících parametrů nefungovala v případě víceřadkové adaptivní lasso metody příliš dobře.

Ohledně reálného příkladu z kapitoly 6 může čtenáře přirozeně napadnout, zda právě kritérium residuálního součtu čtverců, které se snaží minimalizovat všechny naše algoritmy, je tím optimálním kritériem pro diskretní hlasovací data. Přirozeným pravděpodobnostním modelem pro naše data by mohlo být multinomické rozdělení. My jsme se však rozhodli setrvat u součtu čtverců ze dvou důvodů. Zatímco penalizační metody diskutované v kapitole 2 a v sekci 4.1 lze přirozeně rozšířit na obecnou věrohodnost, na AHK je tradičně nahlíženo jako na aproximaci dat v Euklidovském prostoru. Uvažování pravděpodobnostní AHK založené na jiném než normálním rozdělení by tudíž nesedělo do celkového pojetí této práce. Zadruhé, z hlediska aproximace hlasovacích dat prostorem nižší dimenze je Euklidovská metrika zcela přirozená, jelikož vede k tomu, že na každé ose jsou hlasování pro (hodnota 1) a hlasování proti (hodnota -1) stejně vzdálené od nehlasování (hodnota 0). Na druhou stranu, jistě by bylo zajímavé porovnat naše výsledky s výsledky pravděpodobnostní (obyčejné i řídké) AHK založené např. na předpokladu multinomického rozdělení pro naše data. Další alternativou by mohlo být přistupovat k nulám v datech jako k chybějícím pozorováním a snažit se použít logistickou (řídkou) AHK. Poznamenejme, že logistická verze S-PCA algoritmu již byla navržena v Lee a kol. (2010). K této úloze se pravděpodobně vrátíme v budoucnu.

Závěr

V této práci jsme se věnovali řídké modifikaci analýzy hlavních komponent, tedy modifikaci významné statistické a numerické metody, která je navíc základem strojového učení a zpracování velkých dat. Zatímco AHK se snaží pouze maximalizovat rozptyl komprimovaných dat, AŘHK se pokouší najít kompromis mezi maximalizací rozptylu ve směru jistého vektoru (náboje) a malým počtem nenulových koeficientů tohoto vektoru. Důvodem pro zavedení tohoto předpokladu řídkosti je, že snaha o dosažení co nejvyššího rozptylu mnohdy produkuje nežádoucí výsledky v mnohorozměrné situaci, kdy počet proměnných je vyšší než počet pozorování. Předpoklad řídkosti však dělá úlohu výpočetně náročnou (NP-těžkou).

V uplynulé dekádě bylo navrženo hojné množství algoritmů, které se snaží řešit NP-těžkou úlohu AŘHK. Některé z těchto algoritmů jsou teoreticky podložené, ale v praxi nevyužívané kvůli vysokým časovým a paměťovým nárokům nebo nutnosti extrahovat náboje postupně a využívat přitom deflačních metod. Naopak, mnohé algoritmy jsou prakticky dobře využitelné, ačkoliv se jedná spíše o heuristické přístupy k úloze.

My v této práci zaujímáme aplikovaný přístup a soustředíme se na S-PCA algoritmus Zoua a kol. (2006), který je velmi atraktivní z hlediska praktického využití. Jeho atraktivita pramení zejména z toho, že S-PCA nevyžaduje výpočet potenciálně obrovské kovarianční matice, vyhýbá se použití deflačních metod, jelikož počítá všechny požadované náboje současně, a umožňuje uživateli volit přesnou kardinalitu hledaných nábojů. Na druhou stranu, algoritmus se snaží řešit nekonvexní optimalizační úlohu pomocí schématu alternující minimalizace. Toto schéma dělá případné studium vlastností výsledných odhadů velmi obtížné, kvůli čemuž je algoritmus do značné míry heuristickým přístupem k AŘHK.

S-PCA algoritmus zavádí do vnějšího nekonvexního iteračního schématu omezení na ℓ_1 -normu, aby si vynutil řídkost hledaných nábojů. My navrhuje, aby namísto ℓ_1 -penalizace byla použita jiná, nekonvexní penalizace. Naším cílem je zvýšení kvality nalezených nábojů. Rozhodně se nesnažíme tvrdit, že by naše nekonvexní penalizace byla z obecného hlediska vhodnější než slavné lasso. Pouze se domníváme, že z hlediska použití ve vnějším nekonvexním schématu je zcela zbytečné používat konvexní penalizaci a že nekonvexní penalizací může být dosaženo v konkrétním případě AŘHK lepších výsledků. Výsledkem našeho návrhu je nový algoritmus pro AŘHK, který si zachovává stejné, z praktického pohledu atraktivní vlastnosti jako S-PCA algoritmus. Nový algoritmus nazýváme IRLS-S-PCA, jelikož je založený na metodě iterativně vážených nejmenších čtverců. Návrh IRLS-S-PCA je hlavním výpočetním přínosem naší práce.

Pro náš IRLS-S-PCA algoritmus dokazujeme globální konvergenci a podobnými metodami dokazujeme také globální konvergenci původního S-PCA algoritmu. Toto je hlavním teoretickým přínosem naší práce, neboť si nejsme vědomi toho, že by konvergence velmi oblíbeného S-PCA algoritmu byla v minulosti dokázána. Naš důkaz objasňuje důležitost ℓ_2 -penalizace, která je v S-PCA algoritmu přítomna po boku ℓ_1 -penalizace a bývá někdy mylně považována za nadbytečnou.

Dále provádíme simulační studii porovnávající S-PCA a IRLS-S-PCA algoritmy s několika dalšími algoritmy pro AŘHK. Závěrem našich simulací je, že IRLS-S-PCA algoritmus zřetelně překonává S-PCA algoritmus a další uvažované algoritmy. Zdůrazňujeme, že naše simulační studie je díky volbě modelů pro generování dat propojena s mnohými již publikovanými studiemi. Zejména se to týká sekce 5.2, která může být přímo porovnána s metodami navrhanými Shenem a Huangem (2008). Vedlejším výstupem naší simulační studie je zjištění, že hladový algoritmus Greedy-PCA se chová velmi špatně, dokonce mnohem hůře než ten nejjednodušší možný přístup k AŘHK, kterým je jednoduché prahování (ST-PCA). Toto zjištění je poměrně překvapivé nejen kvůli výrazně vyšší časové i paměťové složitosti Greedy-PCA, ale také kvůli tomu, že Greedy-PCA a ST-PCA již byly porovnávány v minulosti a z tohoto soupeření vyšel Greedy-PCA jako vítěz.

Konečně, v poslední kapitole práce se věnujeme analýze reálných dat, jmenovitě datům o hlasování poslanců v Poslanecké sněmovně České republiky. Tato analýza je naším aplikovaným přínosem. AHK je velmi oblíbená metoda i mezi politology a nezdá se, že by v případě hlasovacích dat trpěla prokletím vysoké dimenze. Přesto však AŘHK přináší cennou interpretovatelnost v podobě výběru důležitých hlasování. IRLS-S-PCA algoritmus je schopen získat kvalitativně srovnatelné, ne-li lepší, výsledky než obyčejná AHK, a to za použití pouhých třiceti pěti z celkového počtu více než osmnácti set hlasování. Vzhledem k tomu, že politologové se o výběru důležitých hlasování z různých důvodů často snaží, automatizace této procedury v podobě AŘHK může být cenným přínosem. Bohužel, k detailní interpretaci vybraných proměnných je v případě dat hlasování Poslanecké sněmovny ČR stále potřeba expertního vhledu do problematiky.

Seznam použité literatury

- AMINI, A. A. a WAINWRIGHT, M. J. (2008). High-dimensional analysis of semidefinite relaxations for sparse principal components. In *Information Theory, 2008. ISIT 2008. IEEE International Symposium on*, pages 2454–2458. IEEE.
- ANDERSON, T. W. (1963). Asymptotic theory for principal component analysis. *The Annals of Mathematical Statistics*, **34**(1), 122–148.
- ANTOCH, J., PRCHAL, L., SARDA, P. A KOL. (2010). Nonparametric comparison of roc curves: Testing equivalence. In *Nonparametrics and Robustness in Modern Statistical Inference and Time Series Analysis: A Festschrift in honor of Professor Jana Jurečková*, pages 12–24. Institute of Mathematical Statistics.
- BERTHET, Q. a RIGOLLET, P. (2013a). Complexity theoretic lower bounds for sparse principal component detection. In *COLT*, pages 1046–1066.
- BERTHET, Q., RIGOLLET, P. A KOL. (2013b). Optimal detection of sparse principal components in high dimension. *The Annals of Statistics*, **41**(4), 1780–1815.
- BJÖRCK, K. a GOLUB, G. H. (1973). Numerical methods for computing angles between linear subspaces. *Mathematics of computation*, **27**(123), 579–594.
- BÜHLMANN, P. a VAN DE GEER, S. (2011). *Statistics for high-dimensional data: methods, theory and applications*. Springer Science & Business Media.
- CANDES, E. J., WAKIN, M. B. a BOYD, S. P. (2008). Enhancing sparsity by reweighted ℓ_1 minimization. *Journal of Fourier analysis and applications*, **14**(5), 877–905.
- CHARTRAND, R. a YIN, W. (2008). Iteratively reweighted algorithms for compressive sensing. In *Acoustics, speech and signal processing, 2008. ICASSP 2008. IEEE international conference on*, pages 3869–3872. IEEE.
- CHEN, X. A KOL. (2011). Adaptive elastic-net sparse principal component analysis for pathway association testing. *Stat Appl Genet Mol Biol*, **10**(1), 48.
- CLINTON, J., JACKMAN, S. a RIVERS, D. (2004). The statistical analysis of roll call data. *American Political Science Review*, **98**(02), 355–370.
- D’ASPREMONT, A., EL GHAOUI, L., JORDAN, M. I. a LANCKRIET, G. R. (2007). A direct formulation for sparse pca using semidefinite programming. *SIAM review*, **49**(3), 434–448.
- D’ASPREMONT, A., BACH, F. a GHAOUI, L. E. (2008). Optimal solutions for sparse principal component analysis. *Journal of Machine Learning Research*, **9**(Jul), 1269–1294.
- DAUBECHIES, I., DEVORE, R., FORNASIER, M. a GÜNTÜRK, C. S. (2010). Iteratively reweighted least squares minimization for sparse recovery. *Communications on Pure and Applied Mathematics*, **63**(1), 1–38.

- DONOHO, D. L. (2006). Compressed sensing. *IEEE Transactions on information theory*, **52**(4), 1289–1306.
- ECKART, C. a YOUNG, G. (1936). The approximation of one matrix by another of lower rank. *Psychometrika*, **1**(3), 211–218.
- EFRON, B., HASTIE, T., JOHNSTONE, I., TIBSHIRANI, R. a KOL. (2004). Least angle regression. *The Annals of statistics*, **32**(2), 407–499.
- FAN, J. a LI, R. (2001). Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American statistical Association*, **96**(456), 1348–1360.
- FAN, J. a LI, R. (2006). Statistical challenges with high dimensionality: Feature selection in knowledge discovery. *arXiv preprint math/0602133*.
- FOUCART, S. a RAUHUT, H. (2013). *A mathematical introduction to compressive sensing*. Springer.
- FRIEDMAN, J., HASTIE, T. a TIBSHIRANI, R. (2001). *The elements of statistical learning*, volume 1. Springer series in statistics Springer, Berlin.
- GOLUB, G. H. a VAN LOAN, C. F. (2012). *Matrix computations*, volume 3. JHU Press.
- GU, Q., WANG, Z. a LIU, H. (2014). Sparse pca with oracle property. In *Advances in neural information processing systems*, pages 1529–1537.
- HOERL, A. E. a KENNARD, R. W. (1970). Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, **12**(1), 55–67.
- HOTELLING, H. (1933). Analysis of a complex of statistical variables into principal components. *Journal of educational psychology*, **24**(6), 417.
- HUNTER, D. R. a LI, R. (2005). Variable selection using mm algorithms. *Annals of statistics*, **33**(4), 1617.
- JENATTON, R., OBOZINSKI, G. a BACH, F. R. (2010). Structured sparse principal component analysis. In *AISTATS*, pages 366–373.
- JOHNSTONE, I. M. (2001). On the distribution of the largest eigenvalue in principal components analysis. *Annals of statistics*, pages 295–327.
- JOHNSTONE, I. M. a LU, A. Y. (2009). On consistency and sparsity for principal components analysis in high dimensions. *Journal of the American Statistical Association*, **104**(486), 682–693.
- JOLLIFFE, I. (2002). *Principal component analysis*. Wiley Online Library.
- JOLLIFFE, I. T. (1995). Rotation of principal components: choice of normalization constraints. *Journal of Applied Statistics*, **22**(1), 29–35.
- JOLLIFFE, I. T., TRENDAFILOV, N. T. a UDDIN, M. (2003). A modified principal component technique based on the lasso. *Journal of computational and Graphical Statistics*, **12**(3), 531–547.

- JOURNÉE, M., NESTEROV, Y., RICHTÁRIK, P. a SEPULCHRE, R. (2010). Generalized power method for sparse principal component analysis. *Journal of Machine Learning Research*, **11**(Feb), 517–553.
- LAI, M.-J., XU, Y. a YIN, W. (2013). Improved iteratively reweighted least squares for unconstrained smoothed ℓ_q minimization. *SIAM Journal on Numerical Analysis*, **51**(2), 927–957.
- LANGE, K. (2013). Optimization, vol. 95 of springer texts in statistics.
- LANGE, K., HUNTER, D. R. a YANG, I. (2000). Optimization transfer using surrogate objective functions. *Journal of computational and graphical statistics*, **9**(1), 1–20.
- LEE, S., HUANG, J. Z. a HU, J. (2010). Sparse logistic principal components analysis for binary data. *The annals of applied statistics*, **4**(3), 1579.
- LENG, C. a WANG, H. (2009). On general adaptive sparse principal component analysis. *Journal of Computational and Graphical Statistics*, **18**(1), 201–215.
- LIU, J. a YE, J. (2010). Efficient ℓ_1/ℓ_q norm regularization. *arXiv preprint arXiv:1009.4766*.
- LU, Z. a ZHANG, Y. (2012). An augmented lagrangian approach for sparse principal component analysis. *Mathematical Programming*, pages 1–45.
- MA, S. (2011). Alternating direction method of multipliers for sparse principal component analysis. *Journal of the Operations Research Society of China*, **1**(2), 253–274.
- MA, Z. A KOL. (2013). Sparse principal component analysis and iterative thresholding. *The Annals of Statistics*, **41**(2), 772–801.
- MACKEY, L. W. (2009). Deflation methods for sparse pca. In *Advances in neural information processing systems*, pages 1017–1024.
- MAZUMDER, R., FRIEDMAN, J. H. a HASTIE, T. (2011). Sparsenet: Coordinate descent with nonconvex penalties. *Journal of the American Statistical Association*, **106**(495), 1125–1138.
- MEINSHAUSEN, N. (2007). Relaxed lasso. *Computational Statistics & Data Analysis*, **52**(1), 374–393.
- MEINSHAUSEN, N. a BÜHLMANN, P. (2006). High-dimensional graphs and variable selection with the lasso. *The annals of statistics*, pages 1436–1462.
- MOGHADDAM, B., WEISS, Y. a AVIDAN, S. (2006). Spectral bounds for sparse pca: Exact and greedy algorithms. *Advances in neural information processing systems*, **18**, 915.
- MURPHY, K. P. (2012). *Machine learning: a probabilistic perspective*. MIT press.
- PAUL, D. (2007). Asymptotics of sample eigenstructure for a large dimensional spiked covariance model. *Statistica Sinica*, pages 1617–1642.

- PAUL, D. a JOHNSTONE, I. M. (2012). Augmented sparse principal component analysis for high dimensional data. *arXiv preprint arXiv:1202.1242*.
- PEARSON, K. (1901). Liii. on lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, **2**(11), 559–572.
- R CORE TEAM (2017). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. URL <http://www.R-project.org>. ISBN 3-900051-07-0.
- RICHTÁRIK, P., TAKÁČ, M. a AHİPAŞAOĞLU, S. D. (2012). Alternating maximization: Unifying framework for 8 sparse pca formulations and efficient parallel codes. *arXiv preprint arXiv:1212.4137*.
- ROWEIS, S. (1998). Em algorithms for pca and spca. *Advances in neural information processing systems*, pages 626–632.
- SHEN, D., SHEN, H. a MARRON, J. (2012). A general framework for consistency of principal component analysis. *arXiv preprint arXiv:1211.2671*.
- SHEN, D., SHEN, H. a MARRON, J. S. (2013). Consistency of sparse pca in high dimension, low sample size contexts. *Journal of Multivariate Analysis*, **115**, 317–333.
- SHEN, H. a HUANG, J. Z. (2008). Sparse principal component analysis via regularized low rank matrix approximation. *Journal of multivariate analysis*, **99**(6), 1015–1034.
- SIGG, C. D. a BUHMANN, J. M. (2008). Expectation-maximization for sparse and non-negative pca. In *Proceedings of the 25th international conference on Machine learning*, pages 960–967. ACM.
- TIBSHIRANI, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288.
- TIBSHIRANI, R. J. A KOL. (2013). The lasso problem and uniqueness. *Electronic Journal of Statistics*, **7**, 1456–1490.
- TIKHONOV, A. N. (1963). Solution of incorrectly formulated problems and the regularization methods. In *Soviet Mathematics Doklady*, volume 4, pages 1035–1038.
- TIPPING, M. E. a BISHOP, C. M. (1999). Probabilistic principal component analysis. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, **61**(3), 611–622.
- WANG, T., BERTHET, Q., SAMWORTH, R. J. A KOL. (2016). Statistical and computational trade-offs in estimation of sparse principal components. *The Annals of Statistics*, **44**(5), 1896–1930.
- WIPF, D. a NAGARAJAN, S. (2010). Iterative reweighted ℓ_1 and ℓ_2 methods for finding sparse solutions. *IEEE Journal of Selected Topics in Signal Processing*, **4**(2), 317–329.

- WITTEN, D. M., TIBSHIRANI, R. a HASTIE, T. (2009). A penalized matrix decomposition, with applications to sparse principal components and canonical correlation analysis. *Biostatistics*, page kxp008.
- ZANGWILL, W. I. (1969). Nonlinear programming: a unified approach.
- ZHANG, Y. a KINGSBURY, N. (2010). Fast l0-based sparse signal recovery. In *Machine learning for signal processing (mlsp), 2010 ieee international workshop on*, pages 403–408. IEEE.
- ZHANG, Y. (2011). Sparse principal component analysis: Algorithms and applications. [PhD. dissertation] UC Berkeley.
- ZOU, H. (2006). The adaptive lasso and its oracle properties. *Journal of the American statistical association*, **101**(476), 1418–1429.
- ZOU, H. a HASTIE, T. (2005). Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, **67**(2), 301–320.
- ZOU, H. a HASTIE, T. (2012). *Elastic-Net for Sparse Estimation and Sparse PCA*. URL <http://www.stat.umn.edu/~hzou>. version 1.1.
- ZOU, H. a LI, R. (2008). One-step sparse estimates in nonconcave penalized likelihood models. *Annals of statistics*, **36**(4), 1509.
- ZOU, H. a ZHANG, H. H. (2009). On the adaptive elastic-net with a diverging number of parameters. *Annals of statistics*, **37**(4), 1733.
- ZOU, H., HASTIE, T. a TIBSHIRANI, R. (2006). Sparse principal component analysis. *Journal of computational and graphical statistics*, **15**(2), 265–286.
- ZVÁRA, K. (1989). *Regresní analýza*, volume 1. Praha: Academia.

Dodatky

A. Poznámky ke značení

Uvažujme matici $\mathbf{X} \in \mathbb{R}^{n \times p}$. Sloupce této matice značíme $\mathbf{x}_1, \dots, \mathbf{x}_p \in \mathbb{R}^n$ a řádky této matice značíme $\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(n)} \in \mathbb{R}^p$. Tedy je

$$\mathbf{X} = (\mathbf{x}_1 | \dots | \mathbf{x}_p) = (\mathbf{x}^{(1)} | \dots | \mathbf{x}^{(n)})^\top. \quad (\text{A.1})$$

Je-li $S \subset [p]$, \mathbf{X}_S značí matici, která vznikne z matice \mathbf{X} vypuštěním sloupců, které nenáleží do S . Je-li např. $S = [r]$ pro $r < p$, můžeme psát $\mathbf{X} = (\mathbf{X}_S | \mathbf{X}_{S^c})$, kde $S^c = \{r + 1, \dots, p\}$.

Prvek matice \mathbf{X} na pozici (i, j) značíme jako x_{ij} . Dále i -tý prvek vektoru $\mathbf{v} \in \mathbb{R}^m$ je značen jako v_i . Pokud má již samotný vektor index, oddělujeme index prvku ve vektoru čárkou, tj. $v_{j,i}$ je i -tý prvek vektoru \mathbf{v}_j . Všimněme si, že s tímto značením pro prvky matice \mathbf{X} platí $x_{ij} = x_{j,i}$.

Všechny vektory v této práci jsou sloupcové. Například i -tý řádek matice \mathbf{X} , tj. $\mathbf{x}^{(i)}$, je sám o sobě sloupcový vektor, viz rovnost (A.1).

Vektor $\mathbf{v} \in \mathbb{R}^m$ někdy zapisujeme jako $[v_i]_{i=1}^m$. Tuto konvenci užíváme zejména v případě (částí) gradientu. Je-li $f(\mathbf{x}, \mathbf{y})$ funkce dvou vektorových proměnných $\mathbf{x} \in \mathbb{R}^p$ a $\mathbf{y} \in \mathbb{R}^q$, značíme

$$\nabla_{\mathbf{x}} f(\mathbf{x}, \mathbf{y}) = \left[\frac{\partial}{\partial x_i} f(\mathbf{x}, \mathbf{y}) \right]_{i=1}^p$$

Ještě poznamenejme, že v případě gradientu mírně zneužíváme notaci, když symbolem $\nabla_{\mathbf{x}} f(\mathbf{x}_0, \mathbf{y}_0)$ rozumíme část gradientu příslušnou proměnné \mathbf{x} vyhodnocenou v bodě $(\mathbf{x}_0, \mathbf{y}_0) \in \mathbb{R}^{p+q}$.

K mírnému zneužívání notace dochází také tehdy, je-li matice $\mathbf{L} \in \mathbb{R}^{m \times m}$ diagonální. V tom případě značíme její prvek na pozici (i, j) jako l_i , což může být také i -tý prvek vektoru \mathbf{l} . Vztah mezi diagonální maticí a vektorem diagonálních hodnot je však velmi intuitivní, jelikož

$$\mathbf{L} = \text{diag}(l_1, \dots, l_m) = \text{diag}(\mathbf{l})$$

Zatímco velká tučná písmena značí matice a malá tučná písmena vektory, náhodné veličiny označujeme netučnými velkými písmeny (např. X nebo Z) a náhodné vektory označujeme kroucenými symboly (např. \mathcal{X} , \mathcal{Z}). Výjimkou je náhodný vektor chyb, který značíme tradičně ϵ .

Řešení různých optimalizačních úloh, jako např. úlohy

$$\arg \min_{\beta \in \mathbb{R}^p} \|\mathbf{y} - \mathbf{X}\beta\|_2^2 \quad \text{s.t.} \quad \rho(\beta) \leq t, \quad (\text{A.2})$$

kde $t > 0$ je pevné a $\rho : \mathbb{R}^p \rightarrow \mathbb{R}$ je známá funkce, nemusí být jednoznačně určeno. Předpis (A.2) definuje formálně množinu takových $\beta \in \mathbb{R}^p$, která splňují omezující podmínku a pro která se hodnota účelové funkce shodně rovná minimální

dosazitelné hodnotě na množině dané podmínkou. Přesto však píšeme

$$\hat{\boldsymbol{\beta}} = \arg \min_{\boldsymbol{\beta} \in \mathbb{R}^p} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 \quad \text{s.t.} \quad \rho(\boldsymbol{\beta}) \leq t,$$

čímž rozumíme, že z množiny všech řešení (A.2) je vybráno jedno náhodné. Symbolem $\hat{\boldsymbol{\beta}}$ rozumíme řešení aktuálně uvažované optimalizační úlohy v proměnné $\boldsymbol{\beta}$, tj. odhad neznámé aktuálně uvažované veličiny $\boldsymbol{\beta}$. Výjimkou z této konvence jsou řídké odhady nábojů: i -tý řídký náboj značíme $\tilde{\mathbf{a}}_i$, abychom jej odlišili od i -tého obyčejného (hustého) výběrového náboje $\hat{\mathbf{a}}_i$, který je běžným odhadem skutečného i -tého populačního náboje \mathbf{a}_i . Tento populační náboj ve skutečnosti může, ale nemusí být řídký.

B. Užitá maticová algebra

V této sekci připomeneme některé pojmy z lineární algebry a pravidla pro počítání s maticemi. Naší terminologií se přitom snažíme kopírovat knihu Goluba a Van Loana (2012), která je v tomto kontextu velmi vlivnou publikací. Stejně jako v celém zbytku této práce, i zde pracujeme pouze s reálnými maticemi.

Definice B.1. Systém vektorů $\{\mathbf{v}_1, \dots, \mathbf{v}_q\} \in \mathbb{R}^m$, kde $q \leq m$ se nazývá *ortogonální*, pokud platí

$$\mathbf{v}_i^\top \mathbf{v}_j = 0 \quad \text{pro} \quad i, j \in [q], i \neq j,$$

respektive *ortonormální*, pokud platí

$$\mathbf{v}_i^\top \mathbf{v}_j = \mathbb{1}_{[i=j]} \quad \text{pro} \quad i, j \in [q].$$

Matice \mathbf{V} se nazývá *ortogonální*, resp. *ortonormální*, pokud její sloupce tvoří ortogonální, resp. ortonormální, systém.

Lemma B.2. (Golub a Van Loan, 2012, věta 2.1.1.)

Nechť $r < m$ a $\mathbf{V} \in \mathbb{R}^{m \times r}$ je ortonormální. Pak existuje ortonormální matice $\mathbf{V}_c \in \mathbb{R}^{m \times (m-r)}$ taková, že

$$(\mathbf{V} | \mathbf{V}_c)^\top (\mathbf{V} | \mathbf{V}_c) = \mathbf{I}_m. \quad (1)$$

Věta B.3. (Spektrální rozklad.)

Pro symetrickou matici $\boldsymbol{\Sigma} \in \mathbb{R}^{p \times p}$ existují ortonormální matice $\mathbf{A} \in \mathbb{R}^{p \times p}$ a diagonální matice $\mathbf{L} \in \mathbb{R}^{p \times p}$ s prvky $l_1 \geq \dots \geq l_p$ na diagonále takové, že platí

$$\boldsymbol{\Sigma} = \mathbf{A} \mathbf{L} \mathbf{A}^\top.$$

Diagonální prvky matice \mathbf{L} z předchozí věty se nazývají *vlastní čísla* matice $\boldsymbol{\Sigma}$ a sloupcové vektory matice \mathbf{A} se nazývají *vlastní vektory* matice $\boldsymbol{\Sigma}$. Pro vlastní čísla a vlastní vektory zřejmě platí pro libovolné $i, j \in [p]$, že

$$\begin{aligned} \boldsymbol{\Sigma} \mathbf{a}_j &= l_j \mathbf{a}_j, \\ \mathbf{a}_i^\top \boldsymbol{\Sigma} \mathbf{a}_j &= l_j \mathbb{1}_{[i=j]}. \end{aligned}$$

Věta B.4. (*Singulární rozklad.*)

Nechť $\mathbf{X} \in \mathbb{R}^{n \times p}$, $\text{rank}(\mathbf{X}) = r$. Potom existují matice $\mathbf{U} \in \mathbb{R}^{n \times r}$, $\mathbf{D} \in \mathbb{R}^{r \times r}$ a $\mathbf{V} \in \mathbb{R}^{p \times r}$ takové, že

1. \mathbf{D} je diagonální matice s prvky $d_1 \geq \dots \geq d_r > 0$ na diagonále;
2. \mathbf{U} a \mathbf{V} jsou ortonormální;
3. $\mathbf{X} = \mathbf{UDV}^\top = \sum_{i=1}^r d_i \mathbf{u}_i \mathbf{v}_i^\top$.

Diagonální prvky matice \mathbf{D} se nazývají *singulární čísla* matice \mathbf{X} , sloupcové vektory matice \mathbf{U} se nazývají *levé singulární vektory* matice \mathbf{X} a sloupcové vektory matice \mathbf{V} se nazývají *pravé singulární vektory* matice \mathbf{X} . Levé singulární vektory tvoří ortonormální bázi sloupcového prostoru matice \mathbf{X} , tedy podprostoru v \mathbb{R}^n , a pravé singulární vektory tvoří ortonormální bázi řádkového prostoru matice \mathbf{X} , tedy podprostoru v \mathbb{R}^p .

Poznámka B.5. Singulární rozklad matice \mathbf{X} bývá alternativně zaveden (viz např. Golub a Van Loan, 2012, věta 2.4.1) pomocí matic $\widetilde{\mathbf{U}} \in \mathbb{R}^{n \times n}$, $\widetilde{\mathbf{D}} \in \mathbb{R}^{n \times p}$ a $\widetilde{\mathbf{V}} \in \mathbb{R}^{p \times p}$, pro které platí stejné podmínky jako pro matice \mathbf{U} , \mathbf{D} a \mathbf{V} ve větě B.4 s tou výjimkou, že v podmínce (i) je $\widetilde{d}_{11} \geq \dots \geq \widetilde{d}_{mm} \geq 0$, kde $m = \min(n, p)$, a v podmínce (iii) je horní mez sumy m . Pro matice $\widetilde{\mathbf{U}}$ a $\widetilde{\mathbf{V}}$ platí

$$\begin{aligned}\widetilde{\mathbf{U}} &= (\mathbf{U} | \mathbf{U}_c), \\ \widetilde{\mathbf{V}} &= (\mathbf{V} | \mathbf{V}_c),\end{aligned}$$

kde \mathbf{U} a \mathbf{V} jsou z věty B.4 a \mathbf{U}_c a \mathbf{V}_c jsou ortonormální doplňky k \mathbf{U} a \mathbf{V} z lemmatu B.2. Matice $\widetilde{\mathbf{D}}$ vznikne z matice \mathbf{D} doplněním o nulové sloupce a řádky tak, aby měla správnou dimenzi. Pak je zřejmě $\mathbf{UDV}^\top = \sum_{i=1}^r d_i \mathbf{u}_i \mathbf{v}_i^\top = \sum_{i=1}^m d_i \mathbf{u}_i \mathbf{v}_i^\top = \widetilde{\mathbf{U}} \widetilde{\mathbf{D}} \widetilde{\mathbf{V}}^\top$, jelikož $d_{r+1} = \dots = d_m = 0$. \diamond

„Plný“ singulární rozklad z poznámky B.5 je tedy ekvivalentní „úspornému“ singulárnímu rozkladu z věty B.4. Při výpočtech samozřejmě preferujeme úsporný singulární rozklad, jelikož tím šetříme paměť. Kdykoliv řekneme „nechť $\mathbf{X} = \mathbf{UDV}^\top$ je singulární rozklad matice \mathbf{X} ,“ vždy máme na mysli libovolný rozklad splňující podmínky věty B.4, který implicitně zahrnuje dimenze příslušných matic a tím i hodnot matice \mathbf{X} , která je označena jako r .

Mezi singulárním a spektrálním rozkladem symetrické matice $\mathbf{\Sigma}$ je úzký vztah. Lze snadno ověřit, že pravé singulární vektory matice \mathbf{X} jsou vlastní čísla matice $\mathbf{X}^\top \mathbf{X}$ a levé singulární vektory matice \mathbf{X} jsou vlastní čísla matice $\mathbf{X} \mathbf{X}^\top$. To souvisí s jednoduše ověřitelným faktem, že je-li $\mathbf{X} = \mathbf{UDV}^\top$ je singulární rozklad matice \mathbf{X} , pak $\mathbf{VD}^2 \mathbf{V}^\top$ je spektrální rozklad matice $\mathbf{X}^\top \mathbf{X}$. Tuto vlastnost v podstatě užíváme v důkazu lemmatu 1.9. Nezapomínejme však, že singulární ani spektrální rozklady matice nemusejí být jednoznačné.

Definice B.6. Pro matici $\mathbf{X} \in \mathbb{R}^{n \times p}$ definujeme její *Frobeniovu normu* jako

$$\|\mathbf{X}\|_F = \left(\sum_{i=1}^n \sum_{j=1}^p x_{ij}^2 \right)^{\frac{1}{2}}.$$

Pro čtvercovou matici $\mathbf{A} \in \mathbb{R}^{p \times p}$ definujeme její *stopu* jako součet diagonálních prvků matice, tj.

$$\text{tr}(\mathbf{X}) = \sum_{i=1}^p a_{ii}.$$

Poznámka B.7. Prostým rozepsáním, případně využitím singulárního rozkladu, lze ověřit, že pro Frobeniovu normu a stopu matice platí následující vlastnosti:

1. Pro $\mathbf{X} \in \mathbb{R}^{n \times p}$ se singulárním rozkladem $\mathbf{X} = \mathbf{U}\mathbf{D}\mathbf{V}^T$ platí

$$\|\mathbf{X}\|_F = \sum_{i=1}^n \|\mathbf{x}^{(i)}\|_2^2 = \sum_{j=1}^p \|\mathbf{x}\|_2^2 = \text{tr}(\mathbf{X}^T \mathbf{X}) = \sum_{i=1}^r d_i^2.$$

2. Frobeniova norma je invariantní vůči rotacím, tj. je-li $\mathbf{X} \in \mathbb{R}^{n \times p}$ a $\mathbf{A} \in \mathbb{R}^{n \times n}$ a $\mathbf{B} \in \mathbb{R}^{p \times p}$ jsou ortonormální matice, platí

$$\|\mathbf{X}\|_F = \|\mathbf{X}\mathbf{B}\|_F = \|\mathbf{A}\mathbf{X}\|_F$$

3. Pro $\mathbf{A} \in \mathbb{R}^{n \times p}$ a $\mathbf{B} \in \mathbb{R}^{n \times p}$ platí

$$\text{tr}(\mathbf{A}^T \mathbf{B}) = \text{tr}(\mathbf{B}^T \mathbf{A}),$$

z čehož indukcí plyne, že stopa matice je invariantní vůči cyklickým permutacím matic v argumentu.

◇

Věta B.8. (*Eckartova-Youngova-Minského věta, Eckart a Young, 1936.*)

Nechť $\mathbf{X} \in \mathbb{R}^{n \times p}$, $\mathbf{X} = \mathbf{U}\mathbf{D}\mathbf{V}^T$ je singulární rozklad matice \mathbf{X} a necht $K \in [r]$. Potom řešení úlohy

$$\widehat{\mathbf{X}}_* = \arg \min_{\mathbf{X}_* \in \mathbb{R}^{n \times p}} \|\mathbf{X} - \mathbf{X}_*\|_F \quad \text{s.t.} \quad \text{rank}(\mathbf{X}_*) \leq K$$

je dáno částečným singulárním rozkladem matice \mathbf{X} jako $\widehat{\mathbf{X}}_* = \sum_{i=1}^K d_i \mathbf{u}_i \mathbf{v}_i^T$.