



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV AUTOMATIZACE A MĚŘICÍ TECHNIKY

DEPARTMENT OF CONTROL AND INSTRUMENTATION

ŘÍDKÁ REPREZENTACE SIGNÁLŮ

SPARSE REPRESENTATION OF SIGNALS

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. Michaela Mesárošová

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Miroslav Jirgl, Ph.D.

BRNO 2023

Diplomová práce

magisterský navazující studijní program **Kybernetika, automatizace a měření**

Ústav automatizace a měřicí techniky

Studentka: Bc. Michaela Mesárošová

ID: 211162

Ročník: 2

Akademický rok: 2022/23

NÁZEV TÉMATU:

Řídká reprezentace signálů

POKYNY PRO VYPRACOVÁNÍ:

Cílem práce je aplikace metod řídké reprezentace signálů pro analýzu dat v rámci dodaného datasetu a zhodnocení vlastností a použitelnosti těchto metod.

1. Prostudujte problematiku řídké reprezentace signálů.
2. Seznamte se s formátem a strukturou naměřených dat v rámci dodaného datasetu.
3. Aplikujte metody řídké reprezentace na analýzu dat.
4. Porovnejte výsledky získané metodou SRC s dalšími běžně používanými metodami klasifikace.
5. Prověřte možnost kompenzace poruch signálu a zvýšení kvality signálu využitím metod řídké reprezentace.
6. Diskutujte dosažené výsledky a doplňte vhodnými grafickými ukázkami.

DOPORUČENÁ LITERATURA:

[1] MALLAT, S. G. A wavelet tour of signal processing: the sparse way. 3rd ed. Amsterdam: Elsevier/Academic Press, c2009. ISBN 978-0-12-374370-1.

[2] MIHÁLIK, O., T. SÝKORA, M. HUSÁK a P. FIEDLER. In-Bed Posture Classification Based on Sparse Representation in Redundant Dictionaries. IFAC-PapersOnLine [online]. 2022, 55(4), 374-379 [cit. 2022-08-25]. ISSN 24058963. Dostupné z: <https://www.sciencedirect.com/science/article/pii/S2405896322003779>

Termín zadání: 6.2.2023

Termín odevzdání: 17.5.2023

Vedoucí práce: Ing. Miroslav Jirgl, Ph.D.

doc. Ing. Petr Fiedler, Ph.D.
předseda rady studijního programu

UPOZORNĚNÍ:

Autor diplomové práce nesmí při vytváření diplomové práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

ABSTRAKT

Imobilné či dlhohležiace osoby sú rizikovou skupinou na vznik preležanín a zároveň si vyžadujú ďalšiu starostlivosť. Preto je potrebné stav stav takýchto osôb monitorovať a to čo najjednoduchšie a efektívne. V tejto práci sa zameriavame na spracovanie signálu poskytovaného tlakovým matracom so senzorovou mriežkou 30×11 , na ktorom je osoba umiestnená a jeho možnosti využitia po prevedení do koeficientov riedkej reprezentácie. Redundantné slovníky, alebo tiež rámce, umožňujú neortogonálne reprezentovanie signálov, čo vedie práve k riedkemu zastúpeniu koeficientov. Keďže tento prístup poskytuje množstvo výhodných vlastností a je využívaný v rôznych aplikáciách, ako odšumovanie, segmentácia, robustné prevody, kvantová teória a ďalšie, overili sme možnosť klasifikovania pozície ľahu osoby na základe riedkej reprezentácie. Výsledky boli porovnávané s ďalšími tradičnými klasifikačnými metódami, ktoré sa svojou spoľahlivosťou preukázali ako menej vhodné pre klasifikačný problém s najlepším dosiahnutým výsledkom 92,41 % pre CNN, avšak s vysokými nárokmi na čas a návrh a zložitosť. Úspešnosť klasifikácie pomocou riedkej reprezentácie dosiahla 92,76 % s nižšími nárokmi na návrh a implementačnú komplexitu. Preskúmané boli tiež možnosti klasifikácie a rekonštrukcie obrazu obsahujúceho oklúzie, kde sa riedka reprezentácia ukázala ako efektívna metóda na odstránenie týchto porúch.

KLÚČOVÉ SLOVÁ

Riedka reprezentácia, LASSO, regresný model, klasifikácia polohy ľahu, SRC, rekonštrukcia obrazu

ABSTRACT

People who are immobile or lie for long periods are at high risk of developing pressure ulcers and require additional care. Therefore, it is necessary to monitor the condition of such persons as simply and efficiently as possible. In this work, we focus on processing the signals provided by a pressure mattress with a 30×11 sensor grid on which a person lays and the possibilities of its use after conversion into sparse representation coefficients. Redundant dictionaries, also known as frames, enable non-orthogonal representation of signals, which leads to a sparse representation of coefficients. Since this approach provides many advantageous properties and is being used in various applications, such as denoising, segmentation, robust transformations, quantum theory, and others, we verified the possibility of classifying a person's lying position based on a sparse representation. The results were compared with other traditional classification methods, which were found to be less suitable for the classification problem, with the best-achieved result of 92.41 % for CNN, but with high demands on time, design and complexity. The success rate of the classification reached 92.76 %, with fewer demands on design and implementation complexity. The possibilities of classification and reconstruction of an image containing occlusions were also investigated, where the sparse representation proved to be an effective method to remove these defects.

KEYWORDS

Sparse representation, LASSO, regression model, in-bed position classification, SRC, image reconstruction

MESÁROŠOVÁ, Michaela. *Řídká reprezentace signálů*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav automatizace a měřicí techniky, 2023, 67 s. Diplomová práce. Vedúci práce: Ing. Miroslav Jirgl, Ph.D.

Vyhlásenie autora o pôvodnosti diela

Meno a priezvisko autora: Bc. Michaela Mesárošová
VUT ID autora: 211162
Typ práce: Diplomová práca
Akademický rok: 2022/23
Téma záverečnej práce: Řídká reprezentace signálů

Vyhlasujem, že svoju záverečnú prácu som vypracovala samostatne pod vedením vedúcej/cého záverečnej práce, s využitím odbornej literatúry a ďalších informačných zdrojov, ktoré sú všetky citované v práci a uvedené v zozname literatúry na konci práce.

Ako autorka uvedenej záverečnej práce ďalej vyhlasujem, že v súvislosti s vytvorením tejto záverečnej práce som neporušila autorské práva tretích osôb, najmä som nezasiahla nedovoleným spôsobom do cudzích autorských práv osobnostných a/alebo majetkových a som si plne vedomá následkov porušenia ustanovenia § 11 a nasledujúcich autorského zákona Českej republiky č. 121/2000 Sb., o práve autorskom, o právach súvisiacich s právom autorským a o zmene niektorých zákonov (autorský zákon), v znení neskorších predpisov, vrátane možných trestnoprávných dôsledkov vyplývajúcich z ustanovenia časti druhej, hlavy VI. diel 4 Trestného zákonníka Českej republiky č. 40/2009 Sb.

Brno

.....

podpis autorky*

*Autor podpisuje iba v tlačenej verzii.

POĎAKOVANIE

V prvom rade chcem poďakovať svojej rodine za ich neustálu podporu, povzbudenie a dôveru. Osobitná vďaka patrí pánovi Ing. Miroslavovi Jirglovi, Ph.D., a Ing. Ondrejovi Mihálikovi za odborné vedenie, trpezlivosť a rady, bez ktorých by dokončenie tejto práce nebolo možné a zároveň všetkým, ktorí na tejto ceste prispeli svojím časom, vedomosťami a ochotou sa o ne podeliť. V neposlednom rade vďačím Michalovi Hečkovi za podporu a motiváciu počas celého procesu tvorby práce, hlavne v časoch, keď bola najviac potrebná.

Obsah

Úvod	12
1 Matematický aparát	13
1.1 Báza	13
1.2 Rámce	14
1.3 Lineárne regresné modely	15
1.3.1 Metóda najmenších štvorcov	16
1.3.2 LASSO	17
1.3.3 Grafická interpretácia	19
2 Zber a príprava dát	21
2.1 Metodika	21
2.2 Predspracovanie dát	22
2.3 Výber λ parametra	23
3 Aplikácia LASSO regresie na dataset	26
3.1 Programové fungovanie	26
3.2 Koeficienty riedkej reprezentácie	27
4 Klasifikácia	30
4.1 Krížová validácia	31
4.2 Vyhodnotenie riedkych koeficientov	32
4.2.1 MC	32
4.2.2 MSCC	32
4.2.3 MCR	34
4.3 Úspešnosť klasifikácie	36
5 Porovnané klasifikátory	40
5.1 Konvolučné neurónové siete	40
5.1.1 Štruktúra CNN	40
5.1.2 Konvolučná vrstva	40
5.1.3 Aktivačná funkcia	41
5.1.4 Združovacia vrstva	41
5.1.5 Plne prepojené vrstvy	41
5.2 Rozhodovacie stromy	41
5.2.1 Náhodný les	42
5.2.2 Boosting	43
5.3 KNN	43

6	Návrh klasifikátorov	45
6.1	CNN	45
6.2	Rozhodovacie stromy	48
6.3	KNN	50
6.4	Zhodnotenie výsledkov	51
7	Oklúzie	53
7.1	Klasifikácia okludovaných dát	54
7.2	Rekonštrukcia poškodeného obrazu	56
	Záver	61
	Literatúra	63
	Zoznam symbolov a skratiek	66
8	Obsah elektronickej prílohy	67

Zoznam obrázkov

1.1	Vývoj koeficientov pri zmene λ faktoru	18
1.2	Grafická interpretácia regresíí	19
2.1	Fyzická tlaková podložka	21
2.2	Tok dát pri vytváraní datasetu.	22
2.3	Usporiadanie nespracovaných dát	23
2.4	Vizualizácia dát pre odlišné subjekty zoradené po triedach	24
2.5	Závislosť časovej náročnosti a chyby modelu na λ	25
3.1	Reprezentácia merania č. 2 s príslušnosťou k triede 1	27
3.2	Reprezentácia merania č. 5 s príslušnosťou k triede 2	27
3.3	Reprezentácia merania č. 6 s príslušnosťou k triede 3	28
3.4	Reprezentácia merania č. 13 s príslušnosťou k triede 4	29
4.1	Príklad rozdelenia datasetu počas krížovej validácie	31
4.2	Výber maximálnych koeficientov pre merania č. 2, 5, 6, a 13 zhora	33
4.3	Sumácie koeficientov tried pre merania č. 2, 5, 6, a 13 zhora	34
4.4	Hodnoty reziduí každej triedy pre merania č. 2, 5, 6, a 13 zhora	35
4.5	Porovnanie rekonštrukcie signálu	36
4.6	Matice zámien pre MCSS, MC a MCR	37
6.1	Experiment výberu konvolučných filtrov CNN	46
6.2	Štruktúra navrhutej CNN siete	48
6.3	Rozdiely v štruktúre rozhodovacích stromov	49
6.4	Experiment výberu parametra pre náhodný les	50
6.5	Výberu počtu učiacich iterácií pre boosting	51
6.6	Výber susedov KNN klasifikátora	51
7.1	Ukážka výskytu oklúzie v obraze	53
7.2	Rozdiel v poruchách z pohľadu jednorozmerného signálu	56
7.3	Aproximácia merania nepoškodeného obrazu	58
7.4	Rekonštrukcia obrazu s rôznymi poruchami	60

Zoznam tabuliek

4.1	Súhrn výsledných spoľahlivostí klasifikácií na základe TPR hodnôt . . .	38
4.2	Súhrn spoľahlivostí klasifikácie nenormalizovaných dát	39
6.1	Nastavenia parametrov CART modelu	48
6.2	Porovnanie úspešnosti testovaných klasifikátorov	52
7.1	Výsledky klasifikácie s výskytom oklúzií typu 1	54
7.2	Výsledky klasifikácie s výskytom oklúzií typu 0	55
7.3	Experiment zmeny λ parametra	56
7.4	Zhodnotenie potlačenia šumu	59

Úvod

Preležaniny, alebo tiež dekubity, sú častým rizikom pre osoby pripútané na lôžko, či so zníženou schopnosťou mobility. Za účelom zlepšenia kvality a zefektívnenia starostlivosti o tieto osoby, je potrebné ich stav spoľahlivo a jednoducho monitorovať. Preto bolo navrhnuté riešenie vo forme tlakovej podložky vo forme matraca, pomocou ktorého je možné sledovať ako polohu ležiaceho, tak aj dodatočné užitočné informácie o spánkových zvykoch, vlhkosti a potu osoby a iné. Táto práca sa zameriava na porozumenie a aplikovanie riedkej reprezentácie signálov pomocou redundantných slovníkov, ktoré budú využité na klasifikáciu obrazu z datasetu nameraného na tlakocitlivom matraci. Ako vstupy – príznaky – pre zvolenú klasifikačnú metódu poslúžia práve riedke koeficienty, ktoré budú získavané výpočtom LASSO regresie. Tá poskytuje možnosť flexibility zvolenia miery riedkosti, a taktiež odhaduje model vhodný na interpretáciu klasifikovaného pozorovania.

Cieľom práce je dosiahnuť čo najlepšiu mieru úspešnosti klasifikovania z riedkych koeficientov, čo bude závisieť na viacerých faktoroch. Jedným z nich bude výber vhodných parametrov pre LASSO algoritmus. Ďalšie sa týkajú samotnej klasifikácie, konkrétne zvolenia rozdeľovania datasetu pri krížovej validácii a nájdenia spoľahlivých kritérií pre klasifikovanie do jednotlivých tried. Taktiež je cieľom úspešná rekonštrukcia poškodených dát za pomoci redundantného slovníka.

V prvej kapitole sa sústreďíme na predstavenie matematickej myšlienky. Popíšeme koncept rámcov a fungovanie lineárnych regresíí. Následne sa v ďalšej časti pozrieme na štruktúru datasetu, s ktorým budeme pracovať a zdroje, z ktorých bol získaný. Tiež uvádzame prístup voľby vhodného regularizačného parametra. V tretej kapitole ozrejmieme fungovanie navrhnutého programu pre získavanie príznakov a popíšeme dôležité pozorovania z vygenerovaných riedkych reprezentácií. Ďalej porovnáme klasifikačné kritériá a zhrnieme dosiahnuté výsledky. Štvrtá kapitola je zameraná na použité metódy v klasifikácii pomocou riedkej reprezentácie a vyhodnotenie kvality klasifikácie jednotlivých postupov. Nadchádzajúca časť obsahuje teoretickú stránku klasifikátorov, zvolených na účeli porovnania spoľahlivosti klasifikácie. V kapitole šiestej popisujeme návrhy klasifikátorov na dostupný dataset a proces ich testovania. Tiež pojednávame o dosiahnutých výsledkoch. V neposlednom rade sa zameriame na prácu s dátami obsahujúcimi oklúzie a na možnosti klasifikovania a obnovy poškodeného obrazu.

1 Matematický aparát

Na začiatok je vhodné uviesť čitateľa do teórie, ktorá bude v praxi aplikovaná. Kapitola sa zameriava na polozenie teoretických základov a predpokladov do kontextu problematiky práce, spolu s vyjadrením ich matematických opisov.

1.1 Báza

Súbor vektorov $\mathbf{a}, \mathbf{b}, \mathbf{c}, \dots$ tvorí lineárny vektorový priestor \mathbf{V} ak [16]:

- množina je uzavretá pri komutatívnom a asociatívnom sčítaní, takže

$$\mathbf{a} + \mathbf{b} = \mathbf{b} + \mathbf{a}, \quad (1.1)$$

$$(\mathbf{a} + \mathbf{b}) + \mathbf{c} = \mathbf{a} + (\mathbf{b} + \mathbf{c}); \quad (1.2)$$

- množina je uzavretá pri násobení skalárom (akýmkoľvek komplexným číslom), kedy vznikne nový vektor $\gamma\mathbf{a}$, pričom operácia je distributívna aj asociatívna, takže

$$\gamma(\mathbf{a} + \mathbf{b}) = \gamma\mathbf{a} + \gamma\mathbf{b} \quad (1.3)$$

$$(\gamma + \mu)\mathbf{a} = \gamma\mathbf{a} + \mu\mathbf{a}, \quad (1.4)$$

$$\gamma(\mu\mathbf{a}) = (\gamma\mu)\mathbf{a}, \quad (1.5)$$

kde γ a μ sú ľubovoľné skaláry;

- existuje nulový vektor $\mathbf{0} \in \mathbf{V}$ taký, že

$$\mathbf{a} + \mathbf{0} = \mathbf{a} \quad (1.6)$$

pre všetky \mathbf{a} ;

- násobenie zhodou necháva vektor nezmenený

$$1 \times \mathbf{a} = \mathbf{a} \quad (1.7)$$

- každý vektor má korešpondujúci záporný vektor $-\mathbf{a}$, kde

$$\mathbf{a} + (-\mathbf{a}) = \mathbf{0}. \quad (1.8)$$

Vychádzaním z (1.4) s $\gamma = 1$ a $\mu = -1$ vyplýva, že $-\mathbf{a} = (-1) \times \mathbf{a}$.

Lineárny obal množiny, tiež span vektorov $\mathbf{a}, \mathbf{b}, \dots, \mathbf{s}$ je definovaný ako súbor všetkých vektorov, ktoré možno prepísať ako lineárny súčet pôvodnej množiny, t.j. všetky vektory

$$\mathbf{x} = \alpha\mathbf{a} + \beta\mathbf{b} + \dots + \sigma\mathbf{s}, \quad (1.9)$$

ktoré sú výsledkom nekonečného počtu možných skalárnych hodnôt (všeobecne komplexných čísel) $\alpha, \beta, \dots, \sigma$. Ak \mathbf{x} v (1.9) je rovný $\mathbf{0}$ pre niektorý výber $\alpha, \beta, \dots, \sigma$ (nie všetky sú nulové), teda ak

$$\alpha a + \beta b + \dots + \sigma s = \mathbf{0}, \quad (1.10)$$

potom o množine vektorov $\mathbf{a}, \mathbf{b}, \dots, \mathbf{s}$ hovoríme ako o lineárne závislej. Avšak ak (1.10) nie je uspokojená pre žiadny súbor koeficientov (iný ako v triviálnom prípade, kedy sú všetky nulové), sú vektory lineárne nezávislé a žiadny z nich nie je možné zostrojiť lineárnym súčtom ostatných vektorov [16].

Definícia 1.1. *Báza U vektorového priestoru \mathbf{V} je lineárne nezávislý objekt \mathbf{V} [5], ktorý \mathbf{V} generuje. Ak U je bázou pre \mathbf{V} , potom hovoríme, že vektory v U vytvárajú bázu pre \mathbf{V} .*

Teorém 1.1. *Nech \mathbf{V} je vektorový priestor a $U = \{\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_n\}$ je podmnožina \mathbf{V} . Potom U je bázou \mathbf{V} vtedy a len vtedy ak každý $\mathbf{v} \in \mathbf{V}$ môže byť unikátne vyjadrený ako lineárna kombinácia vektorov z U , to je vo forme*

$$\mathbf{v} = \alpha_1 \mathbf{u}_1 + \alpha_2 \mathbf{u}_2 + \dots + \alpha_n \mathbf{u}_n \quad (1.11)$$

pre unikátne skaláry $\alpha_1, \alpha_2, \dots, \alpha_n$

Dôkaz. Nech U je bázou \mathbf{V} . Ak $\mathbf{v} \in \mathbf{V}$, potom $\mathbf{v} \in \text{span}(U)$ pretože $\text{span}(U) = \mathbf{V}$. Teda \mathbf{v} je lineárna kombinácia vektorov z U . Predpokladajme, že

$$\mathbf{v} = \alpha_1 \mathbf{u}_1 + \alpha_2 \mathbf{u}_2 + \dots + \alpha_n \mathbf{u}_n \quad \text{a} \quad \mathbf{v} = \beta_1 \mathbf{u}_1 + \beta_2 \mathbf{u}_2 + \dots + \beta_n \mathbf{u}_n$$

sú dve reprezentácie \mathbf{v} . Odčítaním druhej rovnice od prvej dostaneme

$$0 = (\alpha_1 - \beta_1) \mathbf{u}_1 + (\alpha_2 - \beta_2) \mathbf{u}_2 + \dots + (\alpha_n - \beta_n) \mathbf{u}_n.$$

Keďže U je lineárne nezávislá, znamená to, že $\alpha_1 - \beta_1 = \alpha_2 - \beta_2 = \dots = \alpha_n - \beta_n = 0$. Potom $\alpha_1 = \beta_1, \alpha_2 = \beta_2, \dots, \alpha_n = \beta_n$, a teda \mathbf{v} je unikátne vyjadriteľný ako lineárna kombinácia vektorov bázy U . [5] ■

1.2 Rámce

V každodennom živote sa stretávame s určitou formou nadbytočnosti, inak *redundanciou*. Či už pri opakovanom kontrolovaní zamknutých dverí na aute, alebo opätovnom overení, že všetci vedia kde a o akom čase majú byť prítomní pri dôležitej udalosti. Odbúravame tak neistoty. Podobný princíp eliminácie neistoty či neurčitosti je použitý aj v oblasti skúmania signálov. Bežné formy reprezentácie signálu

sú v tvare koeficientov bázy. Tieto bázy však vo väčšine prípadov vytvárajú neredundantné spektrum, napríklad pri výskyte pulzu v časovom priebehu signálu, ktorý nasledovne transformujeme pomocou Fourierovej transformácie, bude obsahovať nenulové hodnoty rozložené naprieč celým spektrom. V prípade poškodenia alebo straty dát môžeme prísť o dôležité informácie o signále, ktoré boli obsiahnuté v daných koeficientoch.

Ak k bázam pridáme redundantný doplnok hovoríme o *rámcoch*. V súčasnosti nájdeme ich využitie napríklad v oblasti segmentácie, klasifikácie, kvantovej teórie a výpočtovej technike, odstraňovania šumu a množstvo ďalších [9].

Rámec predstavuje súbor lineárne závislých vektorov $\{\phi_n\}_{n \in \Gamma}$, ktorý charakterizuje ľubovoľný signál z jeho skalárnych súčinov:

$$\{\langle f, \phi_n \rangle\}_{n \in \Gamma}, \quad (1.12)$$

kde Γ môže byť konečná alebo spočítateľne nekonečná množina.

Definícia 1.2. Súbor $\{\phi_n\}_{n \in \Gamma}$ nazývame *rámcom* v Hilbertovom priestore \mathbf{H} [11], ak existujú konštanty $A > 0$ a $B > 0$ také, že pre ľubovoľné $f \in \mathbf{H}$ platí

$$A\|f\|^2 \leq \sum_{n \in \Gamma} |\langle f, \phi_n \rangle|^2 \leq B\|f\|^2 \quad (1.13)$$

Ak $A = B$, rámec nazývame *tesným*.

Konštanty A a B nazývame hranicami a úzko súvisia so stabilitou rekonštrukcie. Ak je uspokojená podmienka z definície 1.2 znamená to, že operátor mapujúci f do transformačných koeficientov musí byť ohraničený, a teda aj výsledná suma v (1.13) bude konečná (ohraničenie zhora), a zároveň žiadne f s $\|f\| > 0$ nebude namapované na hodnotu 0 (ohraničenie zdola). Potom operátor nazývame *rámcový operátor* U . [9][11]

$$\forall n \in \Gamma. \quad Uf[n] = \langle f, \phi_n \rangle \quad (1.14)$$

1.3 Lineárne regresné modely

Regresia využíva súbor matematických a štatistických nástrojov na určenie odhadu vzťahu medzi premennými veličinami. Špecificky môžeme uvažovať všeobecný vzťah medzi nezávislými premennými X , závislými premennými Y a neznámymi parametrami β v tomto tvare:

$$Y = f(X, \beta), \quad (1.15)$$

kde f je predpis regresnej funkcie a parametre β sú hľadané optimalizovaním kvality opisu modelu, vzhľadom k akceptovateľnej chybe (tá je individuálna od problému)[3]. Konkrétne v prípade lineárnej regresie predpokladáme linearitu vstupných parametrov X_1, \dots, X_p . Uvažujme teda vstupný vektor typu $\mathbf{x} = (X_1, X_2, \dots, X_p)^T$ a chceme predikovať výstup Y . Hľadáme takú lineárnu funkciu $f(X)$, ktorá minimalizuje súčet umocnených reziduí z Y . Regresný model má potom tvar

$$\hat{Y} = \hat{\beta}_0 + \sum_{j=1}^p X_j \hat{\beta}_j, \quad (1.16)$$

kde β_0 je v stojovom učení známy ako bias. V prípade, že do \mathbf{x} zakomponujeme konštantu 1 a β_0 zahrnieme do vektora koeficientov β , môžeme vzťah (1.16) preformulovať na

$$\hat{Y} = \mathbf{x}^T \hat{\beta}, \quad (1.17)$$

V uvedenom príklade je modelovaný výstup \hat{Y} skalárna veličina, všeobecne sa však môže jednať o vektor dĺžky K , potom β bude predstavovať maticu, ktorá nadobudne rozmer $p \times K$ [7].

Úspešnosť optimalizácie kriticky závisí od funkcie, ktorú chceme optimalizovať. Všeobecne vieme regresný model interpretovať lepšie ak je jednoduchší, a teda obsahuje menej parametrov, ktoré majú vplyv na výsledok. Z pohľadu predikcie môžu v určitých situáciách prekonávať výkonnosť komplexnejších nelineárnych modelov, obzvlášť v prípadoch, kedy máme k dispozícii len malé množstvo tréningových pozorovaní, riedke dáta, alebo dosahujú nízky pomer signál-šum [7].

1.3.1 Metóda najmenších štvorcov

V štatistike jeden z najčastejšie využívaných regresných modelov je *metóda najmenších štvorcov (MNS)*. Princíp spočíva vo výbere koeficientov $\beta = (\beta_0, \beta_1, \dots, \beta_p)^T$ na základe minimalizácie rezidua súčtu štvorcov nasledovne:

$$\begin{aligned} Q(\beta) &= \sum_{i=1}^N (y_i - f(\mathbf{x}_i))^2 \\ &= \sum_{i=1}^N \left(y_i - \beta_0 - \sum_{j=1}^p x_{ij} \beta_j \right)^2. \end{aligned} \quad (1.18)$$

Zároveň vzťah (1.18) nijak nevytvára o platnosti vytvoreného modelu (1.16), nájde len najlepšie lineárne preloženie dátami.

Predstavme si \mathbf{X} veľkosti $N \times (p+1)$, matica je rozšírená o 1 na prvej pozícii a β_0 je zahrnutá do vektora koeficientov, kde každý riadok predstavuje vstupný vektor,

a podobne nech \mathbf{y} dĺžky N je výstupom. Potom môžeme reziduum sumy štvorcov vyjadriť ako

$$Q(\beta) = (\mathbf{y} - \mathbf{X}\beta)^T(\mathbf{y} - \mathbf{X}\beta) \quad (1.19)$$

Ďalej vyjadríme gradient z (1.19) podľa β .

$$\begin{aligned} \frac{\partial Q(\beta)}{\partial \beta} &= -2\mathbf{X}^T(\mathbf{y} - \mathbf{X}\beta) \\ \frac{\partial^2 Q(\beta)}{\partial \beta \partial \beta^T} &= 2\mathbf{X}^T\mathbf{X} \end{aligned} \quad (1.20)$$

Uvažujme, že \mathbf{X} má plnú stĺpcovú hodnotu, a teda $\mathbf{X}^T\mathbf{X}$ je pozitívne definitné. Potom môžeme prvú deriváciu položiť rovno nule,

$$\mathbf{X}^T(\mathbf{y} - \mathbf{X}\beta) = 0, \quad (1.21)$$

čím získame unikátne riešenie

$$\hat{\beta} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}. \quad (1.22)$$

Predikované hodnoty na vstupnom vektore \mathbf{x}_0 sú dané ako $\hat{f}(\mathbf{x}_0) = (1 : \mathbf{x}_0)^T\hat{\beta}$, potom aproximované hodnoty na tréningový vstup sú

$$\hat{\mathbf{y}} = \mathbf{X}\hat{\beta} = \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}, \quad (1.23)$$

kde $\hat{\mathbf{y}}_i = \hat{f}(\mathbf{x}_i)$. Výsledný odhad $\hat{\mathbf{y}}$ je ortogonálna projekcia \mathbf{y} do stĺpcového priestoru matice \mathbf{X} (ortogonalita je vyjadrená v (1.21)).

Ak stĺpce v \mathbf{X} nie sú lineárne nezávislé, napríklad ak sú dva stĺpce korelované, \mathbf{X} už nemá plnú hodnotu. Potom $\mathbf{X}^T\mathbf{X}$ je singularná a koeficienty $\hat{\beta}$ už nie sú unikátne definované. Výsledné hodnoty $\hat{\mathbf{y}}$ sú stále projekciou \mathbf{y} do podpriestoru, avšak existuje viacero spôsobov, ako túto projekciu interpretovať. Tento prípad sa vyskytuje najčastejšie, ak jeden či viac kvalitatívnych vstupov je zastúpený redundantne.[7]

1.3.2 LASSO

Least absolute shrinkage and selection operator (LASSO), voľným prekladom Operátor najmenšieho absolútneho zmenšenia a výberu, je regresia, ktorej základná vlastnosť je regularizácia modelu. To predstavuje penalizáciu odľahlých hodnôt (robustné voči hodnotám signifikantne vzdialeným od datasetu), ako aj minimalizáciu príznakov, ktoré model popisujú. Táto vlastnosť je zabezpečená pridaním penalizácie pomocou ℓ_1 normy. Ak máme určitý počet predikcií a výstupov systému, usporiadame ich do matice \mathbf{X} a vektoru \mathbf{y} , kde LASSO regresia hľadá vzťah medzi

stĺpcami v matici \mathbf{X} , ktoré budú čo najviac konzistentné so závislou veličinou \mathbf{y} [3]. V pôvodnom texte [23] sa pre LASSO stretávame so vzťahom v nasledovnom tvare:

$$\hat{\beta} = \arg \min_{\beta} \left\{ \sum_{i=1}^N (Y_i - \beta_0 - \sum_{j=1}^p X_{ij} \beta_j)^2 \right\} \quad (1.24)$$

s obmedzením $\sum_j |\beta_j| \leq \tau$

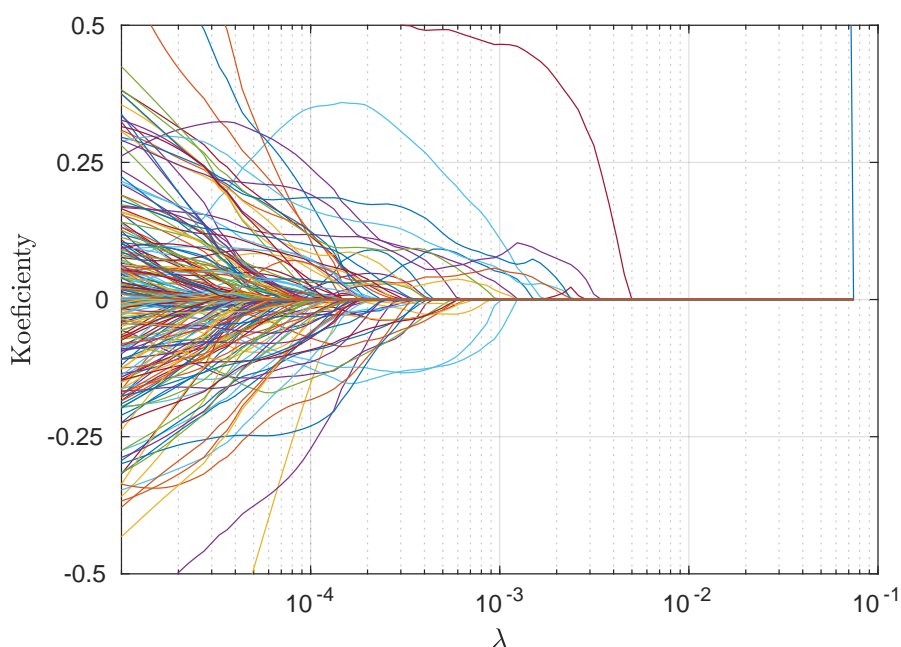
kde τ je parameter ladenia. Po úprave výrazu na jednoduchšie čitateľný formát dostaneme:

$$\arg \min_{\beta} \|\mathbf{y} - \beta_0 - \mathbf{X}\beta\|_2^2 \quad (1.25)$$

s obmedzením $\|\beta\|_1 \leq \tau$

Ekvivalentná formulácia, s ktorou sa najčastejšie stretujeme dnes, nevyjadruje ℓ_1 normu váhového vektora ako obmedzenie, ale ako aditívny parameter. Vzájomný vzťah λ a τ je potom závislý na matici \mathbf{X} a vektore \mathbf{y} .

$$\arg \min_{\beta} \|\mathbf{X}\beta - \mathbf{y}\|_2^2 + \lambda \|\beta\|_1 \quad (1.26)$$

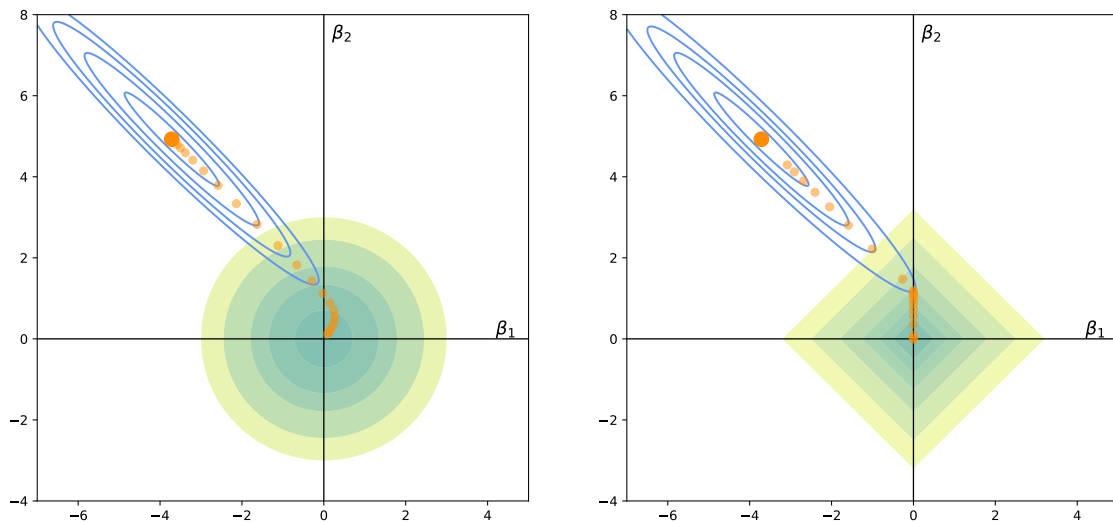


Obr. 1.1: Vývoj koeficientov pri zmene λ faktoru

Obrázok 1.1 demonštruje vplyv λ na celkový odhadovaný model. Zvyšovaním parametru λ obmedzujeme výraz na menší počet výsledných koeficientov, až po dosiahnutie nuly. Naopak jeho zmenšovaním povoľujeme koeficientov viac. Zredukovaním na zanedbateľné číslo dosiahneme výsledok rovnaký, ako pri použití algoritmu najmenších štvorcov. Hodnoty nachádzajúce sa v intervale uvedených extrémov majú za výsledok model riedkej reprezentácie, korešpondujúce s relevantnými pozorovaniami. Ďalšou výhodou oproti MNŠ je predchádzanie preučeniu (angl. overfitting).

1.3.3 Grafická interpretácia

V pôvodnom článku [23] sa stretávame aj s grafickým vysvetlením, prečo LASSO často produkuje nulové koeficienty. Vzťah (1.26) možno po zovšeobecnení vnímať ako minimalizáciu dvoch výrazov: MNŠ + ℓ_q . MNŠ zapísané v tvare (1.19) vytvára eliptické funkcie sústredené okolo odhadu maximálnej pravdepodobnosti (Maximum likelihood), zatiaľ čo výraz ℓ_q vytvára oblasť obmedzenia. Potom riešenie optimalizácie leží v prvom bode prieniku kontúr elipsy a oblasti obmedzenia, kde priesečník sa mení v závislosti od hodnoty λ . Vývoj riešení LASSO regresie porovnáme s hrebeňovou regresiou (angl. ridge regression) s regularizáciou ℓ_2 , čím $\lambda \|\cdot\|_1$ z LASSO regresie nahradíme penalizáciou $\lambda \|\cdot\|_2^2$.



Obr. 1.2: Grafický odhad obrazu hrebeňovej (naľavo) a LASSO regresie (napravo). Modré eliptické oblasti predstavujú kontúry chybovej funkcie MNŠ a plné oblasti sú funkciami obmedzení $\beta_1^2 + \beta_2^2 \leq t$ a $\|\beta_1\| + \|\beta_2\| \leq t$. Oranžové body sú riešeniami regresíí, závislých od λ , kde zväčšený počiatočný bod (stred elipsy) zodpovedá výsledku MNŠ.

Podľa zvolenej normy sa mení tvar oblasti obmedzení. V zjednodušenom prípade uvažujme dva parametre β_1, β_2 , ktoré s normou ℓ_1 formujú diamant $\|\beta_1\| + \|\beta_2\| \leq t$ a pri hrebeňovej regresii vzniká disk $\beta_1^2 + \beta_2^2 \leq t$. Keďže oblasť obmedzenia LASSO priestoru obsahuje vrcholy, existuje pravdepodobnosť, že riešenie bude obsahovať nulové koeficienty. Avšak oblasť pre ℓ_2 normalizáciu je tvorená kruhom a neexistujú vrcholy, ktoré elipsa môže zasiahnuť. Preto sa nulové riešenia budú vyskytovať len veľmi zriedka [7] [20], [23]. Obrázok 1.2, vytvorený na malom názornom súbore hodnôt, jasne zachytáva vývoj koeficientov, ktoré sú v LASSO regresii po doiterovaní na určitú λ potláčané k nule. Zväčšovaním λ hodnoty na nekonečno sa dostávame na riešenie v bode $[0, 0]$, čo je znázornené oranžovými bodmi v grafe a jej zmenšovaním sa približujeme výsledku MNŠ. Na druhej strane hrebeňová regresia síce potláča koeficienty v amplitúde, no pre tvar oblasti obmedzenia sa nedostane do bodu, od ktorého by zostával aspoň jeden koeficient β nulový.

2 Zber a príprava dát

V kapitole sa budeme zaoberať vznikom datasetu, ktorý bude použitý v experimentoch tejto práce, ako aj jeho obsahom a členením. Taktiež popisujeme jeho spracovanie a následný výber regularizačného parametra LASSO regresie, hodiaci sa na pre dáta.

2.1 Metodika

Vytváranie datasetu prebehlo pomocou tlakocitlivej podložky, obrázok 2.1, ktorá sa skladá zo segmentov. Segmenty sú bloky o veľkosti 6×11 snímačov, ktoré je možné prispôbiť ich tvrdosťou na základe subjektívnych preferencií. Celá matica postele tvorí 30×11 snímačov, teda 5 segmentov. Snímače obsiahnuté v matraci fungujú na princípe zmeny odporu pri meniacom sa tlaku, ktorý sa na ne vyvíja. Dáta sú merané po segmentoch, ktoré neskôr vytvárajú ucelený obraz tlakovej mapy.

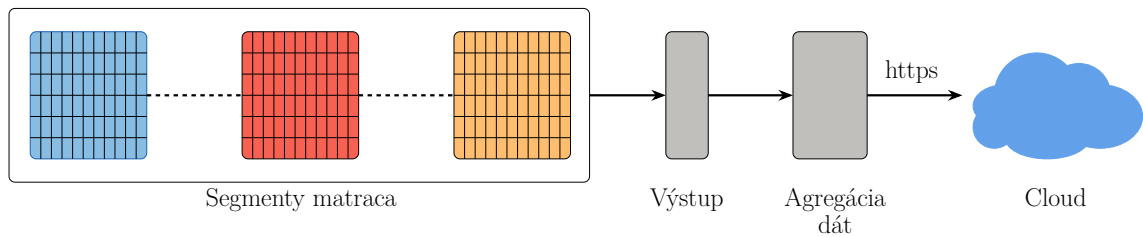


(a) Tlaková podložka

(b) Segment s mriežkou snímačov

Obr. 2.1: Ukážka konkrétneho tlakocitlivého matraca spolu s jeho vnútorným zložením so segmentov.

Anotácia polohy je robená ručne pri zbere dát. Tie sú najskôr ukladané lokálne a následne prostredníctvom https posielené v dávkach na cloud server. V cloude vzniká dataset samotný, do ktorého sú pripájané nasledujúce prichádzajúce dávky nových meraní.



Obr. 2.2: Tok dát pri vytváraní datasetu.

Momentálne figurujú na trhu konkurenčné modely postelových zariadení alebo snímacích podložiek s podobnými funkciami a zameraním, ako napríklad matrace od VistaMedical [1], tlakové podložky od Tekscan [22] alebo prístup na základe merania teploty od Tactilus [8] pomocou termistorov. Keďže cieľom je zlepšiť všeobecnú starostlivosť o imobilných ľudí, či dôchodcov, tieto riešenia sú drahé na širokú distribúciu do nemocničných zariadení, domovov dôchodcov alebo domáce použitie. Preto je snahou vytvoriť dostupnejšie riešenie práve pre tieto účely, čo v konečnom dôsledku vedie aj k efektívnejšej starostlivosti tieto osoby.

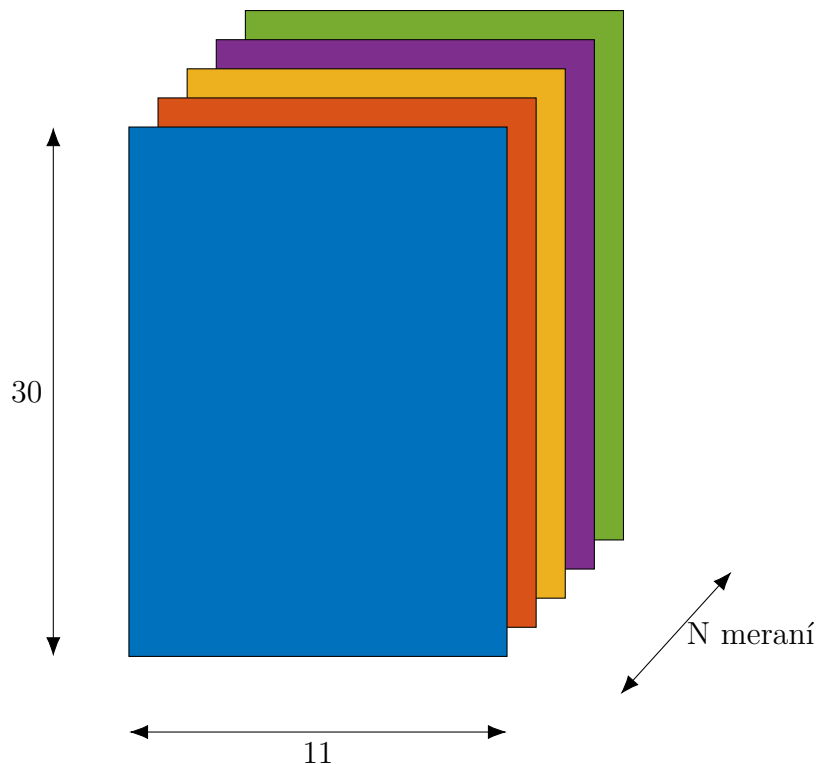
2.2 Predspracovanie dát

Aby bolo možné správne narábať s dátami, je nevyhnutné vedieť ich typ, štruktúru a prípadné dodatočné informácie, ktoré ich sprevádzajú. V našom prípade ide o dáta získané z tlakového matraca, teda tlakové mapy veľkosti 30×11 s hodnotami z intervalu 0 až 1. Každé meranie je vložené do 3D matice spôsobom zobrazeným na obrázku 2.3.

Ďalšia dôležitá informácia ku každému meraniu je údaj o polohe človeka na matraci. Ďalej budeme človeka označovať ako subjekt. Všetky tieto dáta sú kompaktné uložené v skripte `DataCleanPT.mat`. Merania sa zúčastnilo celkom 18 subjektov a celková veľkosť datasetu je 290 meraní. Polohy ľahu, ktoré boli pozorované sú:

- na chrbte,
- na ľavom boku,
- na pravom boku,
- na bruchu

a v rámci každej z nich dochádzalo k miernym zmenám, ako napríklad rôzne polohovanie rúk a nôh a pozícia na matraci – na kraji posteľe, v strede, šikmo, atď. Zmienené polohy budú ďalej označované ako triedy 1 až 4 v poradí, ako sú vypísané a ich ukážka je na nadchádzajúcom obrázku 2.4. Pred prácou s datasetom vykonáme úpravy so zámienkou jednoduchšej manipulácie a zredukovania problému



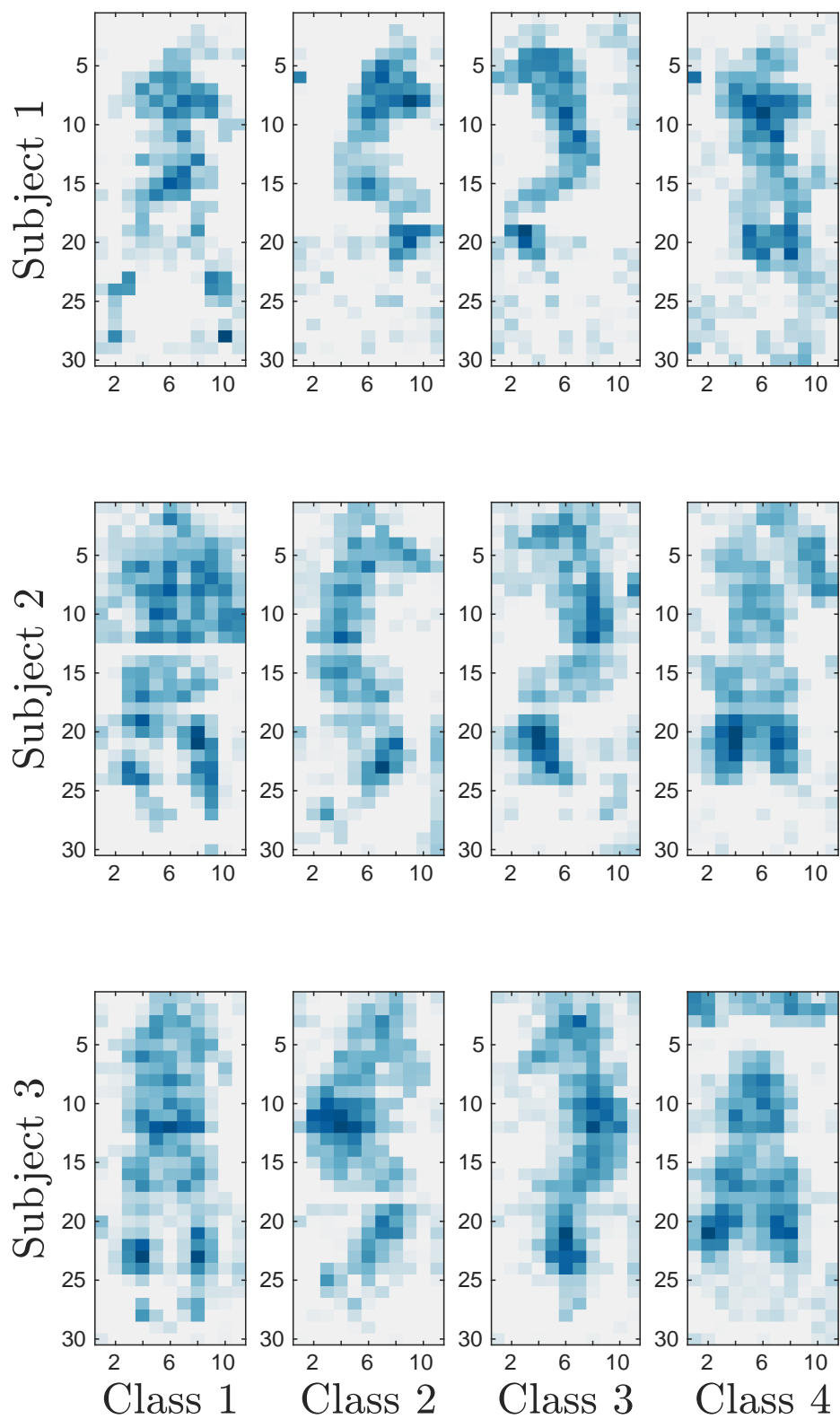
Obr. 2.3: Usporiadanie nespracovaných dát

z 2D priestoru na jednorozmerný. Z tohoto dôvodu pôvodné merania pretransformujeme na stĺpcové vektory a usporiadame do matice. Takto zoradené pozorovania tvoria náš slovník pre budúce výpočty. Aby sme slovník mohli využiť na učenie, je tiež potrebné normalizovať jeho prvky [13].

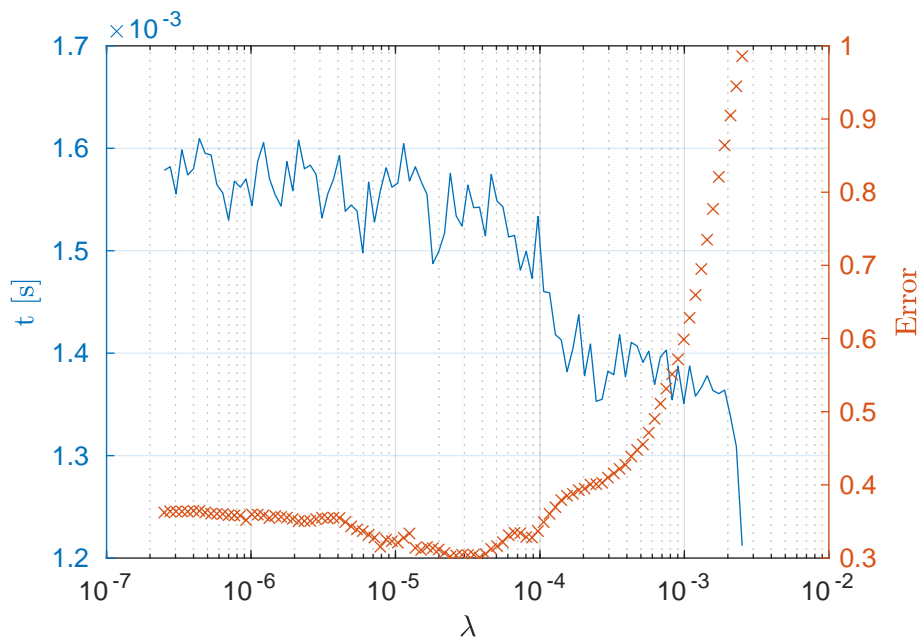
2.3 Výber λ parametra

Prostredie MATLABu poskytuje priamo funkciu `lasso`, ktorá implementuje výpočet vzťahu (1.26). Keďže chceme využiť vlastnosti riedkej reprezentácie, ale zároveň optimalizovať chybu modelu a časovú náročnosť výpočtu, je nevyhnutné zvoliť vhodný parameter λ . Oblasť hodnôt záujmu bola stanovená na základe obrázka 2.5, kde je výsledok iteratívneho zaznamenávania času trvania výpočtu koeficientov pre implicitné λ hodnoty funkcie `lasso`. Zobrazená chyba zodpovedá odchýlke originálnych dát od odhadovaného modelu pre konkrétnu λ a bola vypočítaná pomocou reziduí z kapitoly (4.5).

Ako by sme intuitívne mohli predpokladať, pri znižovaní riedkosti výsledku, a teda zvyšovaní množstva počítaných koeficientov, narastá aj čas behu funkcie. Tento nárast sa však deje len na malom rozmedzí hodnôt s nízkym sklonom, s výnimkou



Obr. 2.4: Vizualizácia dát pre odlišné subjekty zoradené po triedach



Obr. 2.5: Závislosť časovej náročnosti a chyby modelu na λ

konca priebehu. Graf odchýlky má všeobecne opačný trend ako časový priebeh. Ďalšia viditeľná skutočnosť je, že neustálym pridávaním meraní a nedostatočnou penalizáciou sa opäť začne zvyšovať vytvorená chyba medzi originálnym a aproximovaným vstupom. Keďže časové rozdiely sú v relevantnom intervale λ hodnôt zanedbateľne malé, zameriame sa len na priebeh reziduí a jeho oblasť minimálnych výsledkov. Z toho vyplýva, že vhodná λ leží v intervale $< 10^{-4}; 10^{-5} >$.

3 Aplikácia LASSO regresie na dataset

Po definovaní matematických základov a rozboru poskytnutého datasetu je na rade spojenie zmienených aspektov a overenie fungovania v praxi. Na implementáciu bol použitý MATLAB verzie R2022b.

Naším cieľom skúmania je zistiť, či je možné správne reprezentovať, rekonštruovať a klasifikovať aj na základe nízkeho počtu koeficientov za pomoci redundantných slovníkov. Výsledky budú hodnotené pomocou kritérií z kapitoly 4, kde očakávame podobné vyhodnotenia pre jednotlivé merania.

3.1 Programové fungovanie

Pre každý testovací blok boli iteratívne generované koeficienty riedkej reprezentácie patriace vždy jednému konkrétnemu meraniu. Následne sme vyhotovili rekonštrukcie, ktoré slúžili ako na porovnanie s originálnym signálom, tak aj na výpočet chyby modelu pomocou reziduí z (4.6). Uvažujme dataset rozdelený na testovacie merania podľa subjektu č. 2 a tréningové dáta, alebo tiež slovník, tvorený všetkými zostávajúcimi subjektmi (podrobnejšie v kapitole 4). Pre toto rozloženie ukážeme čiastkové výsledky štyroch konkrétnych meraní, každé patriace inej triede, a následne ich klasifikáciu do jednej z tried. Tým sme vytvorili veličiny X a Y v tvare $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_{277}]$ a $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_{13}]$, kde prvky matice reprezentujú stĺpcové vektory, každý obsahujúci dáta obrázku jedného merania. Volanie funkcie a výpočet koeficientov riedkej reprezentácie je v nasledovnej forme:

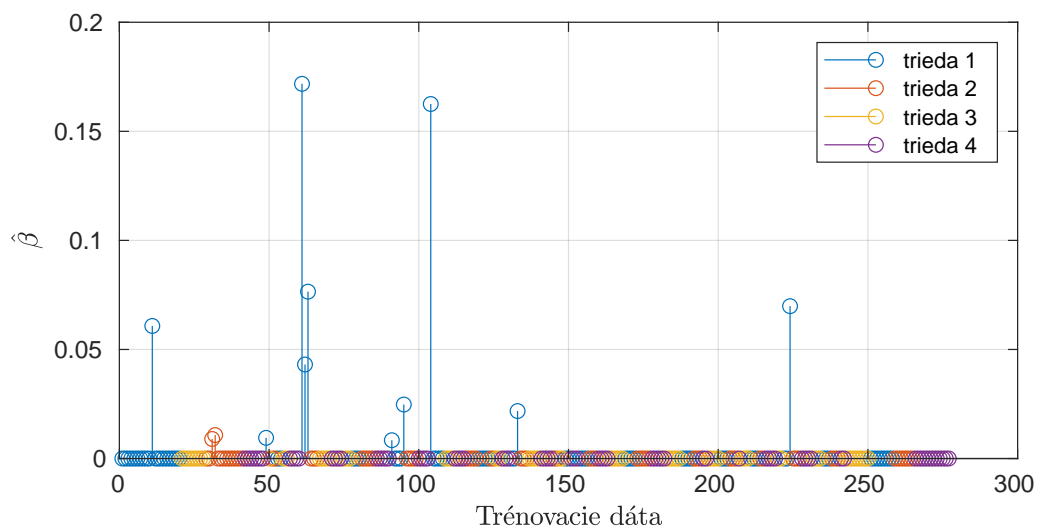
Výpis 3.1: Volanie funkcie `lasso` v prostredí MATLAB

```
1 beta = lasso(train_set, test_set(:,n), ...  
2     "Lambda", lambda, "Intercept", false, ...  
3     "Standardize", false);
```

Parameter λ sme zvolili $7 \cdot 10^{-4}$, vzhľadom na predchádzajúce úvahy v kapitole 2.3. Voliteľný parameter `Intercept` predstavuje bias, alebo β_0 , ktorý sme v našom prípade nezahrnuli do výpočtu, a tak na výsledok majú vplyv len hodnoty obsiahnuté v datasete. Druhý parameter `Standardize` je v tomto prípade, podľa dokumentácie MATLABu, automaticky nastavený na `false` a vo volaní funkcie sa nachádza pre transparentnosť podmienok výpočtu. Po spustení funkcie vektor **beta** obsahuje koeficienty riedkej reprezentácie, ktoré sú v nasledujúcej časti vykreslené pre zobrazenie rozdielov vo výsledkoch medzi triedami.

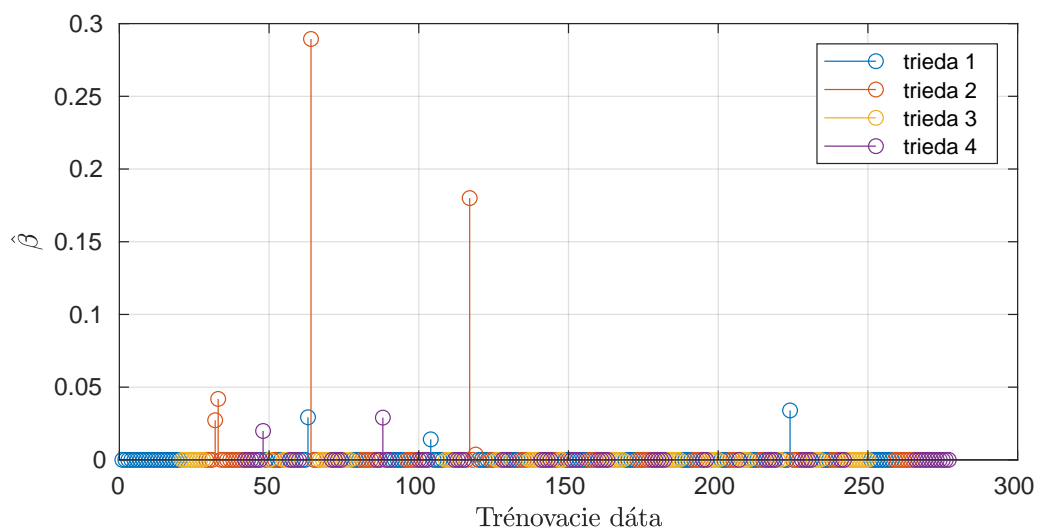
3.2 Koeficienty riedkej reprezentácie

Zvolené štyri reprezentačné merania boli č. 2, č. 5, č. 6 a č. 13 a dôvodom ich výberu sú možnosti poukázania na viaceré skutočnosti medzi koeficientmi riedkej reprezentácie, ktoré budú dôležité pri ďalšom spracovaní a klasifikácii.



Obr. 3.1: Reprezentácia merania č. 2 s príslušnosťou k triede 1

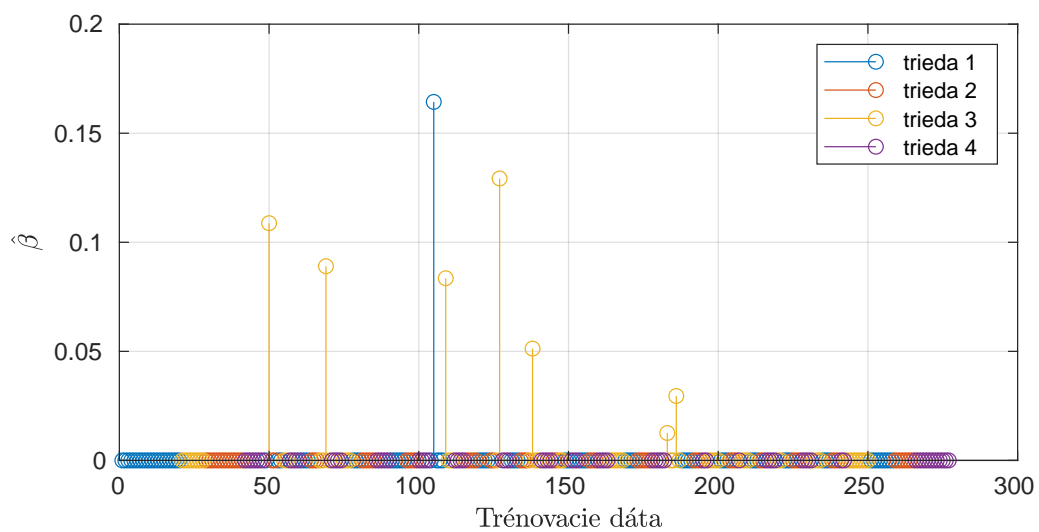
Prvé zobrazenie 3.1 koeficientov riedkej reprezentácie merania č.2 je ukázkovým príkladom, čo by sme mohli očakávať ako priaznivý výsledok LASSO regresie. Trénovacie merania, ktoré reprezentujú testované pozorovanie, majú výrazne prevažujúce



Obr. 3.2: Reprezentácia merania č. 5 s príslušnosťou k triede 2

zastúpenie koeficientov z prvej triedy, čo je zároveň aj trieda zaradenia daného merania. Na druhej strane zvyšné nenulové koeficienty vystupujú len v dvoch meraniach triedy 2 a ich amplitúda sa pohybuje len okolo hodnoty 0,01, čím prispievajú len nepatrným podielom k celkovej aproximácii merania.

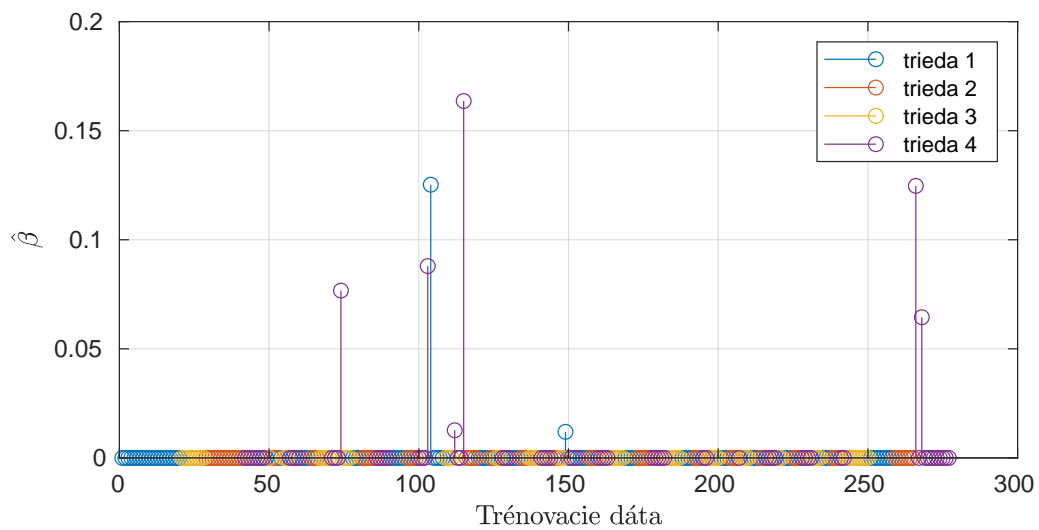
V druhom prípade 3.2 je situácia pestrejšia s mnohorodým zastúpením tried, ktoré koeficienty zostrojujú. Aj napriek početným výskytom inotriednych koeficientov sú merania druhej triedy prevažujúce ako počtom, tak aj vysokou amplitúdou. Trénovacie merania zvyšných tried nepresahujú svojou veľkosťou koeficienty skutočnej príslušnosti merania, avšak už budú tvoriť významnejší podiel na rekonštrukcii vstupu, oproti predchádzajúcemu príkladu z 3.1.



Obr. 3.3: Reprezentácia merania č. 6 s príslušnosťou k triede 3

Graf 3.3 reprezentuje situáciu, kedy množstvo nenulových koeficientov triedy 3 jasne prevyšuje zvyšných zástupcov trénovacích dát. No môžeme vidieť jeden prvok patriaci triede 1, ktorý svojou amplitúdou väčšou ako 0,15 prekonal všetky ostatné zastupujúce merania. Znamená to, že LASSO regresia tento trénovací obraz považovala za najviac podobný testovaciemu vstupu a bol zvolený, ako najviac vhodný resp. dobre rekonštruujúci naše testovacie meranie. Prihliadnuc na rozoberané metódy určovania chýb a klasifikácie v 4 je zjavné, že v takýchto prípadoch je výber klasifikačnej podmienky kľúčovým bodom celkovej úspešnosti klasifikátora.

Posledná ukážka koeficientov riedkej reprezentácie v 3.4 do určitej miery kombinuje vlastnosti z už popísaných grafov, preto predpokladáme, že jeho vysvetlenie nie je nutné rozvádzať. Po rozboře výsledkov `lasso` funkcie sa zameriame na klasifikáciu a rekonštrukciu dát.



Obr. 3.4: Reprezentácia merania č. 13 s príslušnosťou k triede 4

4 Klasifikácia

Klasifikáciu možno definovať ako úlohu zaradenia či rozpoznania prvkov do kategórií na základe ich vlastností, ktoré nazývame *príznamky*. Aby sme mohli určiť príslušnosť nového prvku do triedy, musíme disponovať tréningovou sadou, ktorej roztriedenie je nám už predom známe. Klasifikácia pomocou riedkej reprezentácie sa ukázala ako efektívna v mnohých oblastiach.

Definujme dataset podľa [10] s počtom n vzoriek, kde každá vzorka obsahuje m príznakov. Dataset nazveme slovníkom tréningových dát a definujeme ho ako maticu $\mathbf{X} \in \mathbb{R}^{m \times n}$. Slovník potom obsahuje tréningové dáta zoradené podľa tried C_1, C_2, \dots, C_k . Každý stĺpcový vektor v \mathbf{X} reprezentuje jedno pozorovanie \mathbf{x}_{ij} , kde j je číslo merania v rámci triedy i . Počet prvkov sa môže medzi triedami odlišovať.

$$\mathbf{X} = \begin{bmatrix} | & | & & | & | & & | & | \\ \mathbf{x}_{11} & \mathbf{x}_{12} & \dots & \mathbf{x}_{21} & \mathbf{x}_{22} & \dots & \mathbf{x}_{k1} & \mathbf{x}_{k2} & \dots \\ | & | & & | & | & & | & | \end{bmatrix} \quad (4.1)$$

└───┬───┘
└───┬───┘
└───┬───┘
trieda 1
trieda 2
trieda k

Uvažujme nové meranie v tvare vektora $\mathbf{y} \in \mathbb{R}^m$. Naším cieľom je nájsť taký vektor $\beta \in \mathbb{R}^n$, ktorý pomocou slovníka popíše prvok \mathbf{y} . V predchádzajúcej kapitole sme stanovili, že naším obmedzením riedkosti bude ℓ_1 norma podľa vzťahu (1.26). Potom triedu, do ktorej najpravdepodobnejšie prislúcha nové meranie, hľadáme priradením jeho riedkej reprezentácie k množinám tried slovníka \mathbf{X} . Výsledný lineárny model bude vyzeráť nasledovne:

$$\mathbf{y} = \mathbf{x}_{11}\beta_{11} + \mathbf{x}_{12}\beta_{12} + \dots + \mathbf{x}_{kj}\beta_{kj} \quad (4.2)$$

Tréningové dáta, ktoré dobre reprezentujú \mathbf{y} budú nadobúdať nenulové hodnoty. Na základe tohoto definujeme spôsoby z [10], ktorými budeme určovať príslušnosti k daným triedam.

- Maximálny koeficient (MC): zaradenie prvku bude zhodné s triedou koeficientu patriacemu β , ktorý dosiahol najvyššiu hodnotou.

$$\hat{k} = \arg \max_k (\max_i \beta_{ki}) \quad (4.3)$$

- Maximálny súčet koeficientov v triede (MSCC): predikovaná trieda je určená ako tá, ktorej súčet nenulových koeficientov β je maximalizovaný. Pre každú triedu C_i spočítame súčet z koeficientov β takých, že zodpovedajú prvkom zo

slovníka patriacim tejto triede. Tréningové dáta, ktoré najbližšie reprezentujú \mathbf{y} , by mali pripadať k zhluku elementov vo vektore riedkej reprezentácie.

$$\hat{k} = \arg \max_k \left(\sum_{i=1}^{|C_k|} \beta_{ki} \right) \quad (4.4)$$

- Minimálne reziduum triedy (MCR): nájdením triedy, ktorá minimalizuje reziduum, určíme príslušnosť \mathbf{y} . Reziduum predstavuje chybu medzi testovacím vektorom a zrekonštruovaným vektorom na základe riešenia β :

$$Q = \|\mathbf{y} - \mathbf{X}\beta\|_2 \quad (4.5)$$

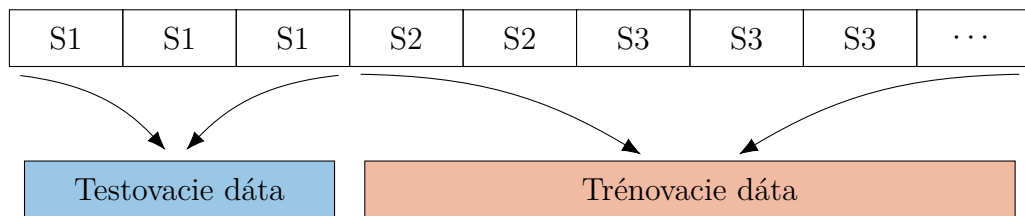
Pre nájdenie triedy potom platí:

$$\hat{k} = \arg \min_k \|\mathbf{y} - \mathbf{X}_k \beta_k\|_2 \quad (4.6)$$

Vyhodnotenie klasifikácie bude ukázané na všetkých vyššie popísaných metódach.

4.1 Krížová validácia

Vždy po vytváraní a trénovaní klasifikátora prichádza na rad otestovanie správnosti jeho pracovania. Tu môžeme naraziť na otázku, aké dáta budú použité na testovanie. Je samozrejmé, že tréningové dáta využiť nemôžeme, pretože nezistíme, ako dobre dokáže klasifikátor generalizovať a taktiež nebudeme schopný zaznamenať pretréningovanie klasifikátora [12]. Na vytvorenie spoľahlivého odhadu spoľahlivosti klasifikátora, obzvlášť pri malom množstve dostupných dát, využívame prístup krížovej validácie. Jedná sa o metódu, v ktorej sú tréningové dáta rozdelené do S skladov / blokov. Potom pre každý sklad $s \in \{1, \dots, S\}$ sa klasifikátor trénuje na všetkých skladoch okrem s -tého a testuje práve na s -tom bloku. Takto sa každý sklad vystrieda v pozícii testovacieho setu [14].



Obr. 4.1: Príklad rozdelenia datasetu počas krížovej validácie

Dataset predstavený v tejto práci bol rozdelený do blokov na základe príslušnosti meraní k subjektom, viď obrázok 4.1. Každý subjekt obsahuje merania zastupujúce

triedy 1–4, teda každá trieda bude spravodlivo zastúpená v každom kole tréningu, menia sa iba celkové počty meraní v rámci subjektu.

4.2 Vyhodnotenie riedkych koeficientov

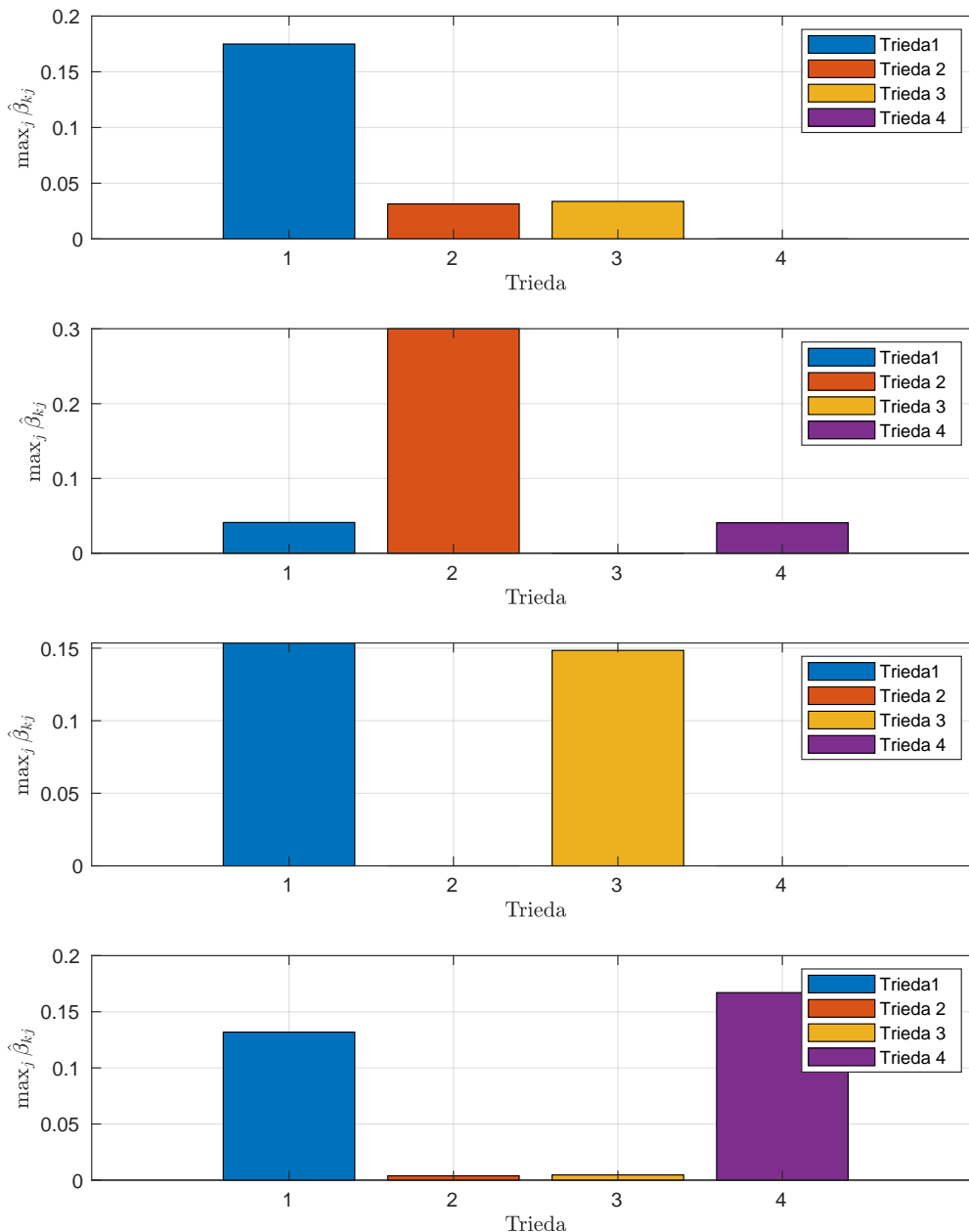
Praktické ukážky rekonštrukcie a výsledkov klasifikácie z riedkej reprezentácie, ďalej aj ako SRC (angl. Sparse representation classification) budú realizované na meraniach, ktorých koeficienty riedkej reprezentácie sme predstavili v predchádzajúcej kapitole. Na záver metódy porovnáme a na základe dosiahnutých výsledkov vyberieme prístup najvhodnejší pre našu aplikáciu.

4.2.1 MC

Ako prvé začneme s najjednoduchším vyhodnotením - výber maximálneho koeficientu MC pre každé meranie. Postupným výpočtom (4.3), nie len z celého spektra, ale aj pre jeho triedy zvlášť, získavame lepši prehľad o výsledku, ktorý je vykreslený v grafoch na obrázku 4.2. Tak, ako sme predpokladali pri rozboře riedkych koeficientov v predchádzajúcej kapitole, sa ukázalo, že pre merania č. 2 a 5 sú klasifikácie jednoznačne určené správnym triedam, keďže všetky ostatné merania druhých tried mali minimálny vplyv na vytváraní modelu a teda aj amplitúdy koeficientov dosahovali veľmi nízkych hodnôt. Na druhej strane pozorovania č. 6 a 13 majú hodnoty MC viditeľne podobné. V prípade č. 13 prebehla klasifikácia do správnej triedy príslušnosti, kdežto č. 6, patriace triede 3, skončilo s priradením do triedy 1. Tu sa prejavila nevýhoda závislosti klasifikácie len na základe jedného maximálneho koeficientu, pretože zo spektra vieme, že napriek veľkej prevahe koeficientov správnej triedy sa môže vyskytnúť iný, ktorý svojou hodnotou zvráti priradenie k príslušajúcej triede.

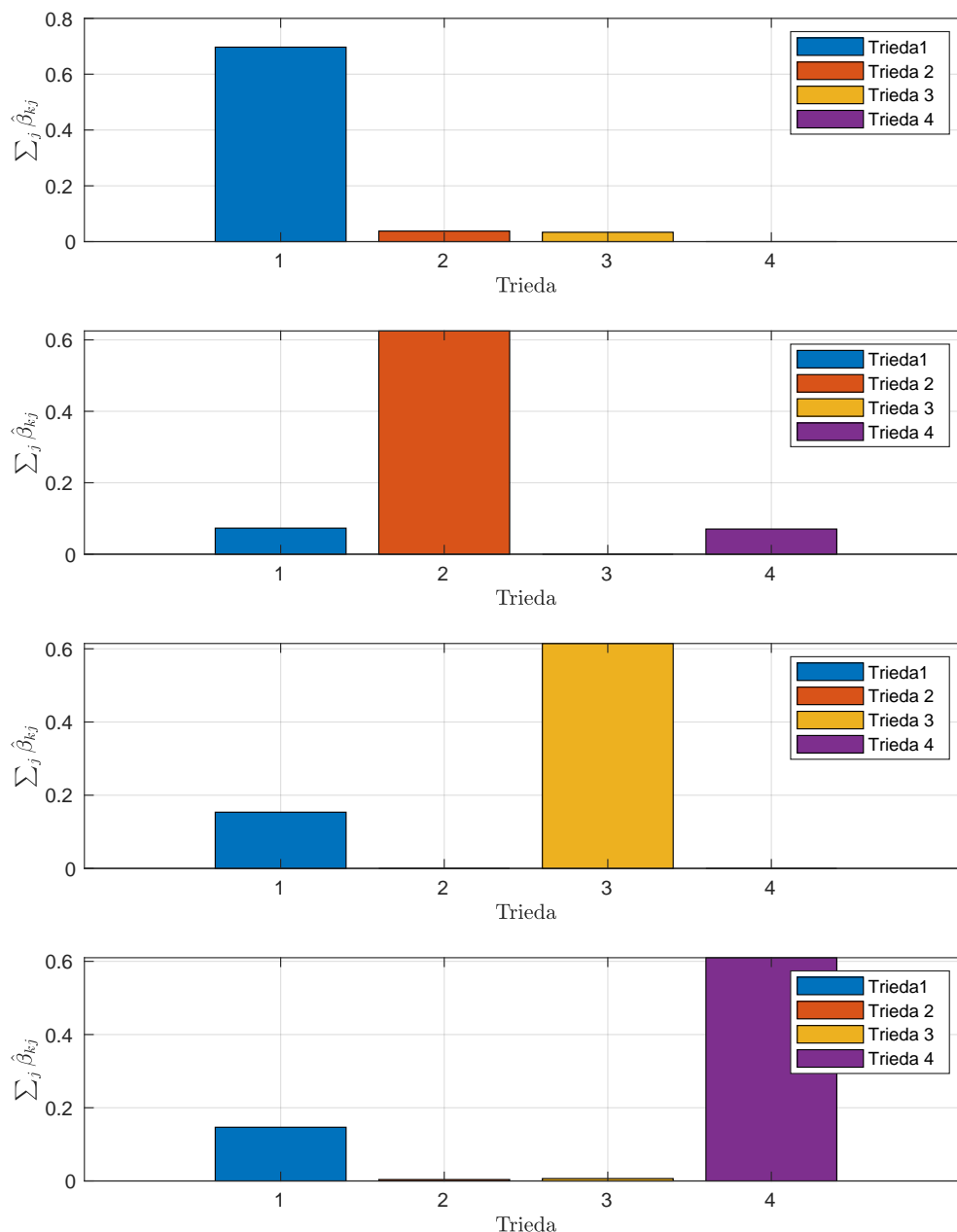
4.2.2 MSCC

Druhým klasifikačným kritériom je porovnanie sumy všetkých koeficientov patriacim k jednej z tried a zaradenie do kategórie s najvyšším výsledkom. Výsledky zodpovedajúce našim príkladným pozorovaniam opäť zobrazíme v grafoch na obrázku 4.3. Je zjavné, že aplikovaním tejto metódy sa nám podarilo eliminovať hlavný problém, ktorý mohol nastať pri použití MC – vplyv vysokých, osamotených koeficientov.



Obr. 4.2: Výber maximálnych koeficientov pre merania č. 2, 5, 6, a 13 zhora

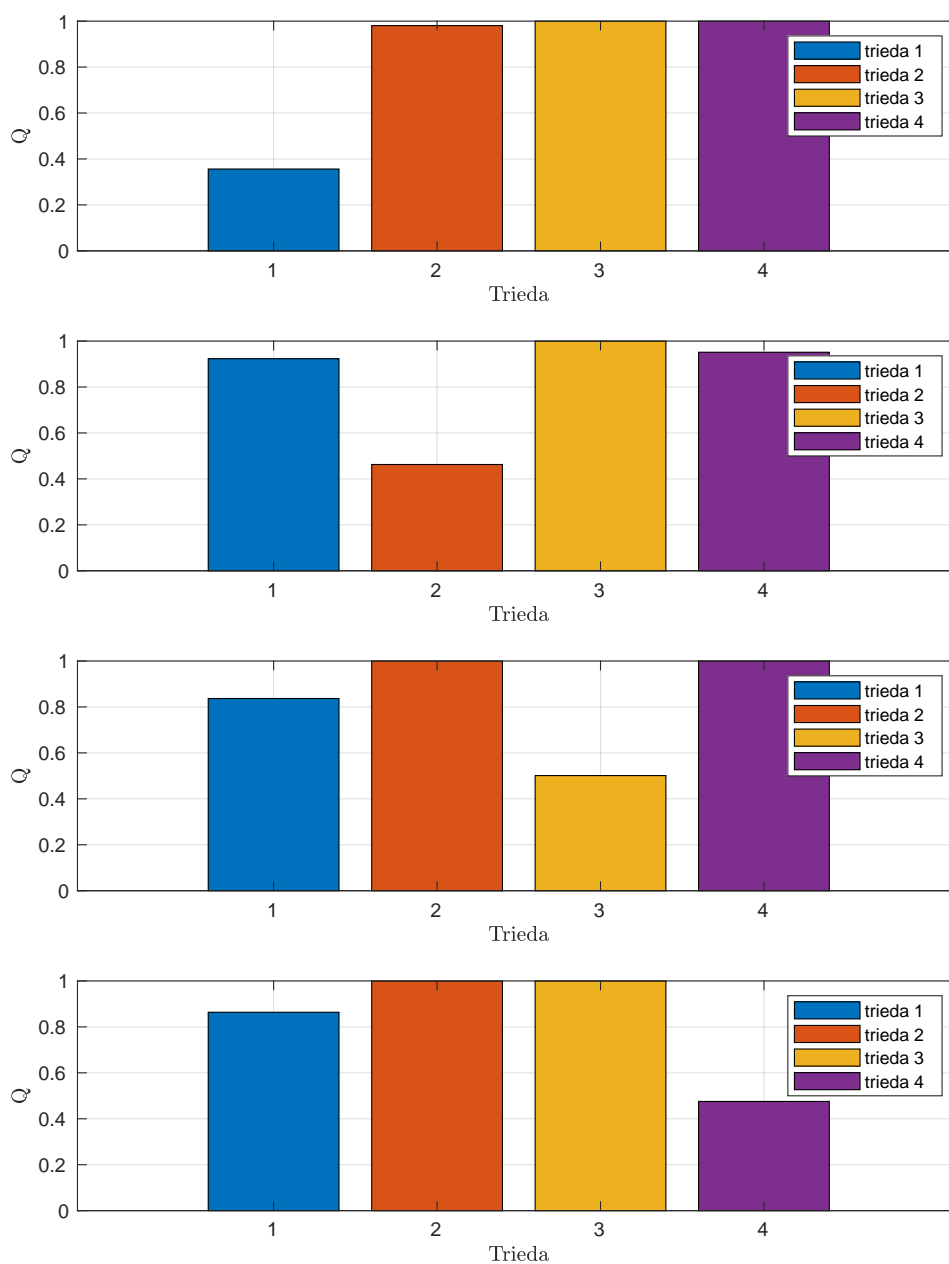
Zatiaľ čo prvé dva grafy sú na prvý pohľad identické s predchádzajúcimi výsledkami, odhliadnuc od číselných hodnôt, pre meranie č. 6 nastala signifikantná zmena v klasifikácii. Pozorovanie bolo zaradené k správnej príslušnosti a toto rozhodnutie bolo, dá sa povedať, istejšie, keďže samotný koeficient prvej triedy bol výraznejšie potlačený súčtom početných koeficientov tretej triedy. V poslednom prípade sa trieda, do ktorej je meranie zaradené nezmenilo, no opäť je viditeľné potlačenie negatívneho efektu vysokých koeficientov rovnako ako pri predchádzajúcom meraní.



Obr. 4.3: Sumácie koeficientov tried pre merania č. 2, 5, 6, a 13 zhora

4.2.3 MCR

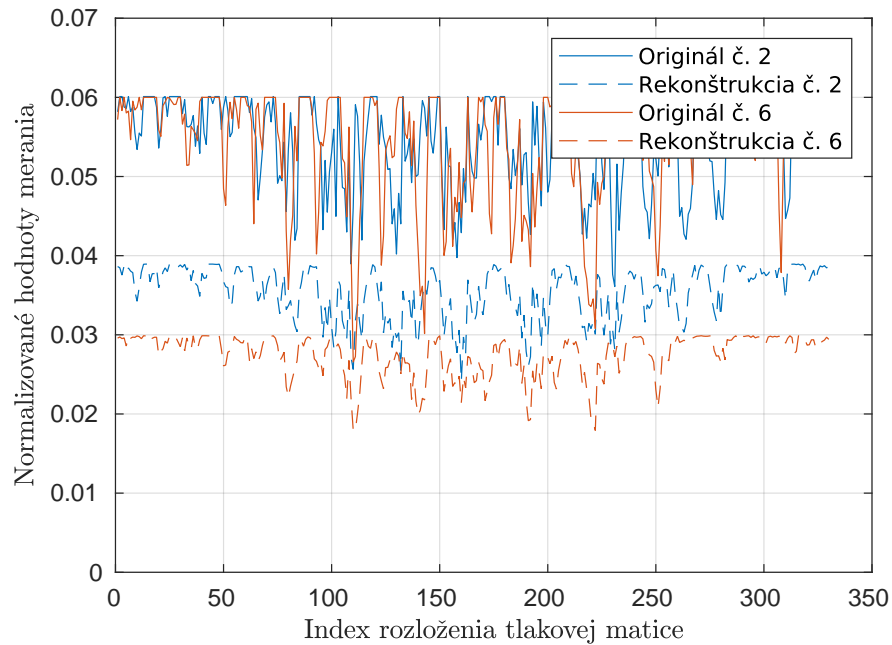
Posledným experimentom je klasifikovanie na základe presnosti rekonštrukcie pozorovania z riedkych koeficientov. Tu sa grafy z 4.4 obrátia, pretože hľadáme minimálne reziduum, a teda model s najmenšou odchýlkou od originálneho merania. Ak má trieda veľkosť rezidua 1, v spektre sa koeficienty z tejto triedy nachádzajú len s nulovými hodnotami.



Obr. 4.4: Hodnoty reziduí každej triedy pre merania č. 2, 5, 6, a 13 zhora

Rovnako ako pri MSCC, klasifikácia na všetkých ukázkach meraní dopadla správne. Pozorovanie 2 je opäť jasne zaradené k svojej správnej triede, kde zvyšné koeficienty výrazne nezasahujú do tvorby celkového predpokladaného modelu. Merania č. 5, 6 a 13 sú ovplyvnené viacerými triedami, ktoré sa podieľajú na výslednom odhade, preto môžeme vidieť aj mierny nárast na hodnotách minimálnych reziduí. Na ucelenie predstavy preto zahrnieme aj ukážku rekonštrukcie meraní 2 a 6, pre kon-

trast medzi najnižšou a najvyššou dosiahnutou minimálnou chybou triedy v rámci analyzovaných príkladov.



Obr. 4.5: Porovnanie rekonštrukcie signálu pomocou koeficientov prvej triedy z merania č. 2, s rekonštruovaným pozorovaním č.6 z koeficientov triedy 3.

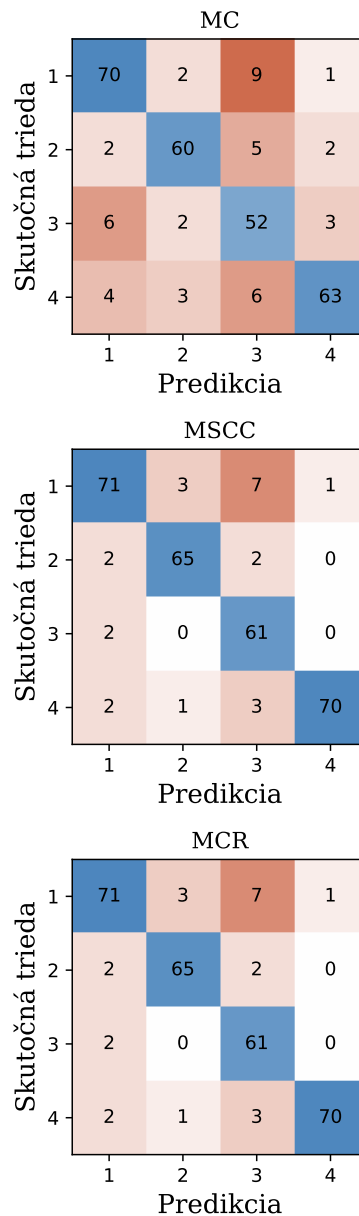
4.3 Úspešnosť klasifikácie

Keď sme vytvorili obraz o tom, ako sa jednotlivé prístupy klasifikácie správajú na reálnych dátach a odhalili ich silné či slabé stránky, prejdeme k vyhodnoteniu celého datasetu. Mieru úspešnosti budeme posudzovať parametrom TPR – true positive rate, ktorý vyjadruje pomer správne klasifikovaných meraní k celkovému počtu klasifikovaných prvkov [4].

$$\text{TPR} = \frac{\text{Korektne klasifikované pozitíva}}{\text{Celkový počet pozitív}} \quad (4.7)$$

Hodnotu TPR pri klasifikovaní do viacerých kategórií vieme výhodne získať z matice zámien (angl. confusion matrix), alebo tiež chybová matica. Táto dvojrozmerná matica predstavuje sumarizáciu úspešnosti klasifikátora s ohľadom na poskytnuté testovacie dáta. Prvá dimenzia (riadky matice) reprezentuje skutočnú triedu, ktorej prvok patrí, druhá (stĺpce matice) zas triedy priradené na základe klasifikácie [24].

Algoritmus spustíme pre všetky dostupné dáta, z čoho vyplýva, že celkovo krížovú validáciu vykonáme 18-krát naprieč datasetom, postupným rozdeľovaním podľa 4.1. Každému meraniu priradíme triedu na základe stanovených klasifikačných kritérií a ich medzivýsledky ukladáme do vektorov predpokladov. Pomocou nich sme schopný vytvoriť matice zámien, ktoré sú na obrázku 4.6. Následne pre všetky metódy vypočítame TPR, na základe ktorého zvolíme najvhodnejší prístup klasifikácie v našej problematike.



Obr. 4.6: Matice zámien pre MSCC, MC a MCR s úspešnosťou klasifikovania pre všetky kategórie

Ak začneme metódou MC zhora, je vidieť, že problém z predošlých úvah v častiach 3.1 a 4.2.1 - zaradenie merania na základe jedného koeficientu - sa prejavil ešte výraznejšie naprieč celým datasetom. Najviac problémov nastalo pri klasifikácii triedy 3 s najnižšou dosiahnutou úspešnosťou 82,5 %, s najčastejšou zámennou za triedu 1. Naopak s hodnotou TPR 87,0 % sa podarilo klasifikovať triedu 2, čo ju robí najpresnejšie klasifikovanou. Čo taktiež možno vyčítať je, že najčastejším problémom bola zámena skutočnej kategórie s triedou tretou. Výsledné TPR sa pohybuje na hodnote $\approx 0,83$, čo po prevedení na percentá dáva klasifikátoru mieru úspešnosti 83,10 %. Celkovo sa tak metóda MC preukázala v našej aplikácii ako menej vyhovujúce kritérium na klasifikáciu.

Ďalej sa zameriame na prístupy MSCC a MCR, ktoré zhrnieme spoločne. Podľa ich chybových matíc sa ich výsledok úplne zhoduje, čo bolo možné očakávať. MCR vieme totiž vnímať ako nadstavbu konceptu MSCC. Ak prvky spektra jednej triedy prispievajú najväčším podielom k aproximácii pôvodného signálu, potom aj rekonštrukcia obrazu pomocou týchto prvkov bude veľmi pravdepodobne najpresnejšia, teda vyprodukuje najmenšiu chybu oproti originálnemu vstupu. Oba prístupy dosiahli rovnakú najmenšiu presnosť priradenia 86,6 % pri triede 1 a najvyššiu pri zaradovaní triedy tretej – 96,8 %. Celková úspešnosť klasifikácie dosiahla až 92,76 %, čím výrazne prekonal predchádzajúce kritérium MC.

Tab. 4.1: Súhrn výsledných spoľahlivostí klasifikácií na základe TPR hodnôt

	TPR [%]	Časová náročnosť [s]
MC	83,10	2,54
MSCC	92,76	2,26
MCR	92,76	2,35

Je jasné, že na základe dvoch identických výsledkov nie je možné vybrať jeden lepší. Preto existuje viac možností, na základe ktorých by pri konkrétnom praktickom prevedení závisel výber najvhodnejšej metódy. Prvým prístupom môže byť stanovená časová náročnosť na výpočet a klasifikáciu. V našom prípade je rýchlejší výpočet MSCC s časom 2,26 s oproti MCR s časom 2,35 s. Takýto časový rozdiel je v podstate zanedbateľný ale pri zväčšení datasetu by mohol byť výraznejší a vhodný na zváženie. Ak nie sme striktné časovo viazaný, je ďalšou alternatívou rozšírenie slovníka, najmä o pozorovania z triedy s najnižšou spoľahlivosťou klasifikácie. Nasledovalo by opätovné porovnanie MSCC a MCR. Ako bolo spomenuté v predchádzajúcom texte, využívali sme na výpočty normalizované dáta. To zahŕňa merania nie len v slovníku – tréningový dataset, ale i dáta privádzané na vstup – validačný dataset. Keďže nemusí byť žiaduce vykonávať dodatočné predspracovanie

nových vstupov, uvádzame aj riešenie bez tohoto medzikroku. Zmena vstupu si vyžaduje taktiež úpravu regularizačného parametru λ , pretože normalizácia znižuje amplitúdu, ktorú dosahujú hodnoty v obraze. To znamená, že LASSO pri výpočte chyby aproximácie z nenormalizovaných dát, ale pôvodnej λ hodnote, bude výsledok penalizovať nesprávne. Po pokusoch dosiahnuť rovnaké alebo podobné výsledky sme sa postupným zvyšovaním hodnoty dopracovali na regularizáciu $\lambda = 1 \cdot 10^{-2}$. Výsledky opäť zhrnieme do tabuľky 4.2.

Tab. 4.2: Súhrn výsledných spoľahlivostí klasifikácie na nenormalizovaných dátach

	TPR [%]	Časová náročnosť [s]
MC	83,79	5,41
MSCC	92,76	5,43
MCR	92,76	5,44

Je evidentné, že nastavením λ na správnu hodnotu sa nám podarilo dosiahnuť rovnako dobrých výsledkov, ako v prípade výpočtov s normalizovanými dátami. Metóda MC si dokonca polepšila svoju spoľahlivosť o necelé percento. Stále tak nedosahuje na výsledky MSCC a MCR, no odráža to skutočnosť, že každá klasifikačná metóda si vyžaduje spracované príznaky na základe jej klasifikačného princípu, alebo podľa podstaty samotných príznakov. Na jednej strane normalizáciou pomáhame eliminovať šum, na druhej strane tak môžeme znižovať alebo úplne odstrániť dôležité informácie a rozdiely medzi príznakmi. Preto všeobecne nie je možné jednoznačne určiť, ktorý prístup je správny a zaručuje priaznivé výsledky. Ďalší upozorovaný jav, ktorý si možno všimnúť je, že čas výpočtov sa dvojnásobne zvýšil oproti hodnotám v tabuľke 4.1. Takýto nárast sa môže prejaviť výraznejšie pri rozšírení datasetu. Keďže sa v našom klasifikačnom probléme normalizácia neprejavuje veľkými zmenami a vieme výsledky optimalizovať nastavením vhodných regularizačných parametrov, budeme aj naďalej pracovať z časových dôvodov s normalizovaným datasetom.

5 Porovnané klasifikátory

Pre objektívne zhodnotenie výsledkov klasifikácie SRC je nutné preskúmať iné, už známe a často používané prístupy strojového učenia a metódy rozpoznávania. Na základe nich bude možné posúdiť vhodnosť SRC pre zvolenú klasifikačnú úlohu.

5.1 Konvolučné neurónové siete

Podkapitola je čerpaná z [15], ak nie je uvedené inak. Aplikácie strojového učenia boli často založené na príznakoch osobitne extrahovaných z dát. Avšak výber vhodných reprezentatívnych príznakov vyžaduje značné odborné znalosti v oblasti a množstvo inžinierskej práce. *Konvolučné neurónové siete* (CNN) patria medzi metódy hlbokého učenia a ich veľká výhoda je schopnosť spracovávať dáta bez predspracovania, ako napríklad obrázky alebo audio signály. Úloha, na ktorú CNN využívame, ovplyvňuje vnútornú štruktúru a usporiadanie vrstiev siete.

5.1.1 Štruktúra CNN

CNN je dopredná sieť, čo znamená, že informácia je v inferenčnej fáze predávaná iba v smere od vstupu k výstupu, bez spätných slučiek. Kľúčovým prvkom, ako možno odvodiť názvu metódy, sú konvolučné vrstvy, ktoré sa učia extrahovať príznaky priamo z dát, využívaním konvolučných operácií. Finálny výstup získavame aplikáciou nelineárnej aktivačnej funkcie na výsledok z konvolučných vrstiev. Združovacie vrstvy (angl. pooling layers) menia veľkosť mapy príznakov vyprodukovanej konvolučnými vrstvami. Plne prepojené vrstvy si môžeme predstaviť ako klasifikátor aplikovaný na extrahované príznaky. Výstupom plne prepojených vrstiev je finálna klasifikácia.

5.1.2 Konvolučná vrstva

Konvolučné vrstvy sú založené na typickej operácii spracovania obrazu, 2D konvolúcia medzi vstupným obrazom \mathbf{O} a konvolučným jadrom \mathbf{K} , ktorý predstavuje maticu váh. Výsledná hodnota pre prvok $\mathbf{X}(m, n)$ je vypočítaná prekrytím jadra cez obrázok so stredom jadra navrchu prvku, prevzatím súčinov jednotlivých prvkov medzi hodnotami jadra a zodpovedajúcimi vstupnými prvkami a sčítaním hodnôt.

$$\mathbf{Y}(m, n) = \sum_{i=1}^K \sum_{j=1}^K \mathbf{K}(i, j) \mathbf{X}(m - 1 + i, n - 1 + j) \quad (5.1)$$

V spracovaní signálov (5.1) zodpovedá 2D krížovej korelácii, konvolúcia zahrňuje aj otočenie jadra, čo v prípade CNN spôsobí iba naučenie siete na otočený obraz.

5.1.3 Aktivačná funkcia

Aktivačné funkcie možno považovať za zhodnotenie, či by sa neurón mal alebo nemal aktivovať pri príchode špecifického vstupu. Najjednoduchšou funkciou je vypínač, ktorý sa aktivuje pri splnení určitých podmienok. Tento druh aktivácie však nie je optimálny pre zložitejšie prípady, ako je klasifikácia do viacerých tried, kde je výhodné porovnávať úroveň aktivácií v rôznych neurónoch. Druhy aktivačných funkcií:

- Jednotkový skok
- Sigmoid
- Tanh
- Rectified Linear Unit (ReLU)
- leaky ReLU
- ELU

5.1.4 Združovacia vrstva

Združovacie vrstvy znižujú priestorovú veľkosť máp príznakov extrahovaných konvolučnými vrstvami. Znižujú sa tým nároky na výpočtovú náročnosť a umožňujú sa nasledujúcej konvolučnej vrstve extrahovať príznaky v inej mierke. Dva hlavné prístupy k združovaniu sú maximálne združovanie (angl. max pooling) a priemerné združovanie (angl. average pooling). Maximálne združovanie vyberá maximálnu hodnotu v receptívnom poli združovacieho jadra, zatiaľ čo priemerné združovanie berie priemer všetkých hodnôt v oblasti.

5.1.5 Plne prepojené vrstvy

Plne prepojené vrstvy pozostávajú zo štandardných neurónov a spájajú každý vstupný prvok s každým výstupným prvkom so samostatnou váhou. Mnoho architektúr CNN má na konci siete niekoľko plne prepojených vrstiev a považujú sa za klasifikátory využívajúce príznaky extrahované konvolučnými vrstvami. Výstupná vrstva je plne prepojená vrstva, ktorej výstupom je predikcia siete pre daný vstup. Výstupný formát závisí od aktuálnej úlohy.

5.2 Rozhodovacie stromy

Rozhodovací strom, alebo aj CART (angl. classification and regression trees), je klasifikátor vyjadrený ako rekurzívne rozdelenie priestoru inštancií [17]. Rozhodovací strom pozostáva z koreňa a uzlov. Strom, ktorý nemá žiadne prichádzajúce hrany do koreňa nazývame riadeným stromom. Všetky ostatné uzly majú presne jednu

vstupnú hranu. Uzol s odchádzajúcimi hranami sa nazýva interný alebo testovací uzol, všetky ostatné uzly sú listami. V rozhodovacom strome každý vnútorný uzol rozdeľuje priestor inštancií na dva alebo viac podpriestorov, podľa určitej diskkrétnej funkcie hodnôt vstupných príznakov. V najjednoduchšom a najčastejšom prípade každý test zohľadňuje jeden príznak, takže priestor inštancií je rozdelený podľa hodnoty príznaku. V prípade číselných príznakov sa podmienka vzťahuje na rozsah. Každý list je zaradený do jednej triedy, ktorá predstavuje najvhodnejšiu cieľovú hodnotu. Alternatívne môže list obsahovať pravdepodobnostný vektor označujúci pravdepodobnosť, že cieľový príznak má určitú hodnotu. Inštancie sa klasifikujú tak, že sú navigované od koreňa stromu až po list podľa výsledkov testov pozdĺž cesty. Podľa [2] zložitosť stromu má zásadný vplyv na jeho presnosť.

Medzi veľké výhody CART patrí ich jednoduchá interpretovateľnosť, schopnosť pracovať so zmiešaným spojitým a diskrétnym vstupom, taktiež relatívna robustnosť voči odlahlým hodnotám a ich dobrá škálovateľnosť na veľké datasety. Vedia sa tiež vysporiadať s chýbajúcimi vstupmi. Avšak CART modely majú aj rad nevýhod, primárne ich nižšia presnosť predikcie oproti iným modelom. Prídruženým problémom je nestabilita stromov, pre ktoré aj malá zmena vo vstupných dátach má rozsiahly vplyv na štruktúru stromu, vzhľadom na jeho hierarchické usporiadanie v procese rastu. Častá terminológia označuje CART ako odhad s vysokým rozptylom. V nasledujúcej časti sa zameriame na možné eliminácie tohoto faktu prostredníctvom zoskupení (angl. ensembles) [17].

5.2.1 Náhodný les

Jednou z možností zredukovania rozptylu odhadu je spriemerovanie mnohých odhadov. Ako príklad môžeme natrénovať N rôznych stromov na rozdielnych podmnožinách datasetu a vypočítať výsledné zoskupenie ako

$$f(x) = \sum_{n=1}^N \frac{1}{N} f_n(x), \quad (5.2)$$

kde f_n predstavuje n -tý strom. Táto technika sa nazýva bagging, odvodené z bootstrap agregácie. Opätovné spúšťanie rovnakého učiaceho algoritmu môže viesť k vysoko korelujúcim prediktorom, čo limituje mieru redukcie rozptylu. Preto sa metóda náhodných lesov snaží dekorovať základných žiakov (angl. learners) učením stromov na náhodne vybranej podmnožiny vstupných dát, ako aj náhodne vybranej podmnožiny prípadov/tried dát. Takéto modely majú často veľmi dobrú úspešnosť predikcií a sú používané v mnohých aplikáciách [17].

5.2.2 Boosting

Boosting patrí k hladným algoritmom, ktorý vytvára kolekciu slabých žiakov s úspešnosťou len o málo lepšou ako náhodný generátor. Ich spojením do zoskupenia vzniká klasifikátor s dobrou presnosťou. Hlavným algoritmom je Adaboost, z ktorého neskôr vznikali ďalšie odvodené variácie. Adaboost využíva inováciu priradzovania váh dátam na základe toho, aké ťažké bolo pre predchádzajúce klasifikátory určiť správnu triedu. Váhy sú predávané klasifikátoru ako súčasť vstupu počas tréovania. Konceptne je Adaboost veľmi jednoduchý. V každej iterácii je tréovaný nový klasifikátor na tréovacím vstupe, na ktorý sú aplikované váhy, kde každý bod vstupu je upravený individuálne na základe jeho predošlej klasifikácie. Počiatočné hodnoty váh majú rovnakú hodnotu $1/N$, kde N je počet prvkov v tréningových dátach. Potom je po každej iterácii spočítaná chyba ϵ ako suma váh nesprávne klasifikovaných prvkov, a váhy týchto prvkov znásobíme hodnotou α :

$$\epsilon = \sum_{n=1}^N \omega_n I(k_n \neq \hat{k}_n) \quad (5.3)$$

$$\alpha = \log \left(\frac{1 - \epsilon}{\epsilon} \right), \quad (5.4)$$

kde ω_n je váha pripadajúca n -tému prvku a $I(k_n \neq \hat{k}_n)$ je indikátorová funkcia s návratovou hodnotou 1, ak sa predikovaná a skutočná trieda nezhodujú a hodnotou 0, ak zhoda nastala. Váhy nesprávne klasifikovaných vstupov sa následne znormalizujú, čím efektívne redukuje vplyv korektné zaradených prvkov. Ako podmienku ukončenia tréovania možno využiť viacero možností. Buď stanovíme maximálny povolený počet iterácií / učiacich cyklov, alebo po úspešnom zaklasifikovaní všetkých prvkov, či nakoniec ak jeden prvok obsahuje váhu vyššiu ako polovicu dostupných váh [12].

5.3 KNN

KNN (angl. k nearest neighbours), alebo tiež *k najbližších susedov*, patrí do skupiny bezparametrových klasifikátorov. Jeho prístup je jednoduchý v princípe, kedy nemáme k dispozícii model popisujúci dáta. Ďalšia najlepšia možnosť je pozrieť sa na podobné dáta a zaradiť sa do ich triedy [12]. Rozhodnutie závisí na K bodoch $x_{(i)}$, $i = 1, \dots, K$ v tréningovom sete, ktoré sú vzdialenosťou najbližšie testovaciemu vstupu y . Sčítaním počtu členov každej triedy v tejto skupine a výberom najvyššieho zastúpenia zaradíme vstup do prislúchajúcej triedy. Elementom na zváženie je spôsob počítania vzdialenosti tréovacích dát od vstupu, najčastejšie využívanou metrikou je Euklidovská vzdialenosť [7], [14]

$$d_{(i)} = \|\mathbf{x}_{(i)} - \mathbf{y}\|. \quad (5.5)$$

Výber vhodnej hodnoty K hrá dôležitú rolu. Pri nízkom počte uvažovaných bodov je klasifikátor citlivý na šum, zatiaľ čo prílišným zväčšením K berieme do úvahy aj nadmieru vzdialené body, ktoré svojím príspevkom znižujú presnosť klasifikácie [12].

6 Návrh klasifikátorov

V tejto časti sa zameriame na konkrétne návrhy klasifikátorov kapitoly 5, uvádzame všetky postupy a úvahy pri ich vytváraní. Všetky dosiahnuté výsledky klasifikácie sú zhrnuté v tabuľke 6.2, spolu s porovnaním troch metód klasifikácie SRC. Klasifikátory boli navrhované v prostredí MATLAB 2022b.

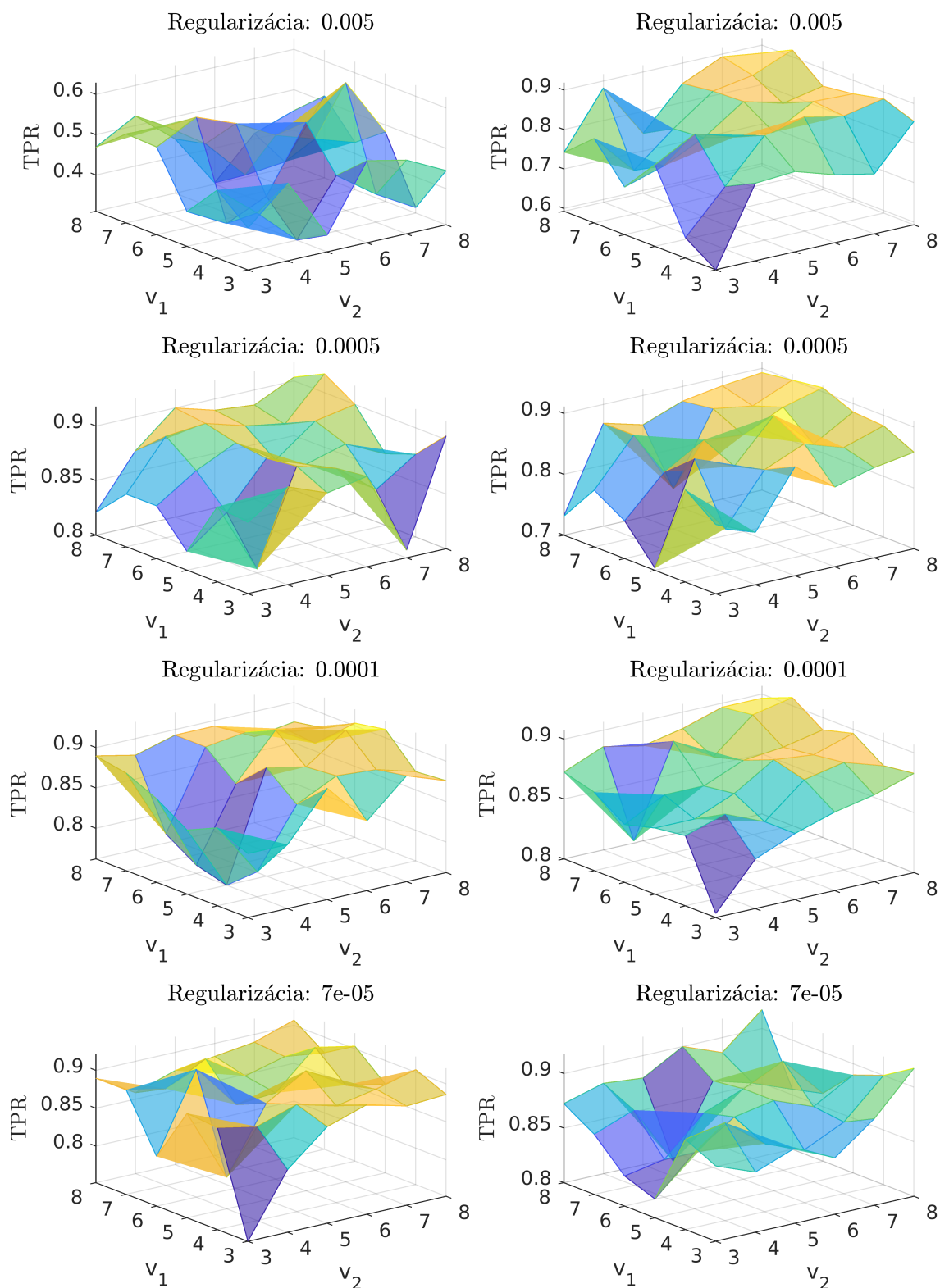
6.1 CNN

Konvolučná neurónová sieť je najzložitejším klasifikátorom na návrh z vybraných v kapitole 5. Ponúka veľmi rozsiahle množstvo nastaviteľných parametrov, kde treba myslieť na ich veľkosť, počet, typ a aj umiestnenie v konvolučnej sieti. Taktiež v metódach umelej inteligencie neexistuje univerzálna cesta, ktorá by objasňovala správny prístup a postup návrhu CNN. Postup sa totiž môže výrazne meniť pri každom druhu klasifikačného problému, pri rôznych typoch vstupných dát, ich počtu a druhu a aj celkovej aplikácie. Naša úloha sa líši od klasických CNN aplikácií aj veľkosťou dostupného datasetu a rozmerov obrazu - 290 meraní veľkosti 30×11 , zatiaľ čo táto metóda tradične využíva databázy obsahujúce tisíce až desiatitisíce prvkov. Keďže otestovanie všetkých možností siete nie je reálne splniteľná úloha, budeme pracovať s optimálnym riešením, ktoré sa nám podarilo nájsť v našich experimentoch. Na začiatok sme stanovili cieľ zhotoviť sieť s čo najjednoduchšou štruktúrou, preto prvý parameter, ktorý na počiatok zvolíme, sú dve konvolučné vrstvy. Tradične sa v nich využívajú filtre veľkosti 3×3 alebo 5×5 , ktoré budeme reprezentovať množinou $\{3, 5\}$. Konvolučné vrstvy potom vieme zapísať ako dvojicu parametrov siete (v_1, v_2) , kde v_i predstavuje konvolučný filter v i -tej konvolučnej vrstve s rozmerom $v_i \times v_i$, pre $i \in \{1, 2\}$. Otestované boli nasledovné kombinácie:

$$(v_1, v_2); \quad v_1, v_2 \in \{3, 5\}, \quad (6.1)$$

Testovanie prebehlo na regularizačných parametroch $5 \cdot 10^{-3}$, $2 \cdot 10^{-3}$, $1 \cdot 10^{-3}$, $5 \cdot 10^{-4}$, $2 \cdot 10^{-4}$, $1 \cdot 10^{-4}$, $7 \cdot 10^{-5}$ a $5 \cdot 10^{-5}$. Ukázalo sa, že všetky zoradenia filtrov najčastejšie dosahujú TPR hodnôt v intervale 80 – 90 %, viď obrázok 6.1. Rozhodujúcim faktorom v tomto prípade bude nájsť najvyššie TPR v čo najstabilnejšej oblasti a zároveň v konfigurácii s čo najnižším počtom potrebných filtrov v každej vrstve. Po porovnaní rôznych kombinácií sme vypozerovali, že túto možnosť ponúka usporiadanie $(v_1, v_2) = (3, 3)$.

Ďalším ladiacim prvkom, ktorý sme podrobili testovaniu, boli aktivačné funkcie. Vyskúšané funkcie boli sigmoid, tanh, leaky ReLU a ELU, ktoré sme aplikovali v rôznych usporiadaniach a pároch. Ako prvú funkciu sme z potenciálnej konvolučnej



Obr. 6.1: Výsledky TPR v ľavom stĺpci pre kombináciu konvolučných filtrov $(v_1, v_2) = (3, 3)$ a v pravom stĺpci $(v_1, v_2) = (3, 5)$, pri meniacej sa miere regularizácie siete.

siete vyradili sigmoid, spôsobujúci drastický pokles v správnosti klasifikácie na 24 %. Taktiež patrí k výpočtovo náročnejším funkciami, spolu s tanh, oproti jednoduchým leaky ReLU a ELU. Pre zvyšné tri funkcie sa TPR pohybovala v malom intervale 89 – 91 %, s malým náskokom pre leaky ReLU po prvej konvolučnej vrstve a ELU za druhou konvolučnou vrstvou s úspešnosťou 92 %.

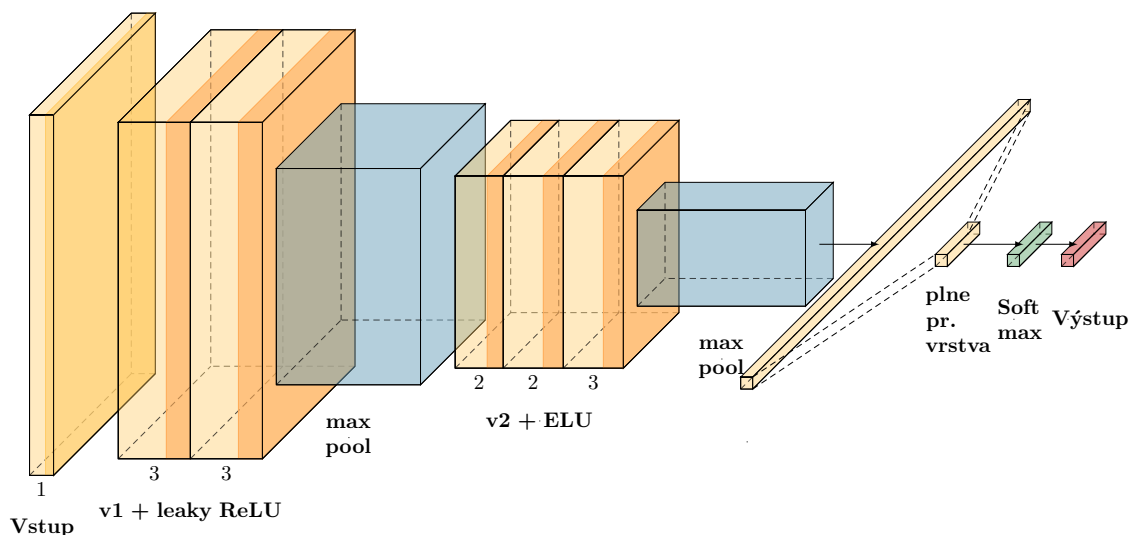
Posledná možnosť, ktorú sme testovali v zaradení do štruktúry, sú podvzorkovacie vrstvy – maximálne a priemerné združovanie. Po uvážení veľkosti vstupného obrazu sme zvolili rozmery združovacích vrstiev 2×2 a 3×3 , $\{2, 3\}$, a veľkosti kroku posunutia okna 1 a 2. Jednotlivé konfigurácie porovnávané v experimentoch je potom možné popísať vektorom parametrov (z_1, p_1, z_2, p_2) , kde z_i značí, že i -ta združovacia vrstva má rozmer $z_i \times z_i$ a p_i určuje veľkosť kroku okna, pre $i \in \{1, 2\}$. Testovanie prebehlo pre tieto štvorice:

$$(z_1, p_1, z_2, p_2) \in (\{2, 3\} \times \{1, 2\})^2 \quad (6.2)$$

Ako nevhodný sa ukázal prístup priemerovania, ktorý dosahoval pre všetky nastavenia 6.2 horšie výsledky oproti výberu maxim a to ≈ 88 %. Preto budeme ďalej podvzorkovacou vrstvou uvažovať maximálne združovanie. V jej prípade sa rozdiely v presnosti klasifikácie rôznych konfigurácií líšili len nepatrne, kolísali o ± 2 %. Je nutné si uvedomiť, že znižovanie kroku posunu zvyšuje výpočtovú náročnosť. Pri využití kroku 1 čo i len v jednom podvzorkovaní sme zdvojnásobili trénovací čas na 117s, v porovnaní s časom 66s pri $(p_1, p_2) = (2, 2)$. Najlepšej spoľahlivosti sme dosiahli práve s nastavením $(z_1, p_1, z_2, p_2) = (3, 1, 2, 2)$. Vzhľadom na veľkosť našej CNN siete a rozlíšenie vstupného obrazu, nie sú tieto časové hodnoty tak závažné, no v prípade zmeny datasetu (množstvo meraní alebo typ dát), je to faktor, ktorý bude potrebné zohľadňovať.

Po prihliadnutí na všetky medzivýsledky a odskúšané parametre sme dospeli k nasledovnej konfigurácii CNN:

1. 6 konvolučných filtrov 3×3 ,
2. leaky ReLU,
3. maximálne združovanie 3×3 ,
4. 7 konvolučných filtrov 3×3 ,
5. ELU,
6. maximálne združovanie 2×2 ,
7. plne prepojená vrstva,
8. softmax.



Obr. 6.2: Výsledná štruktúra navrhnujej CNN siete. Čísla pod blokmi značia počet kanálov danej vrstvy.

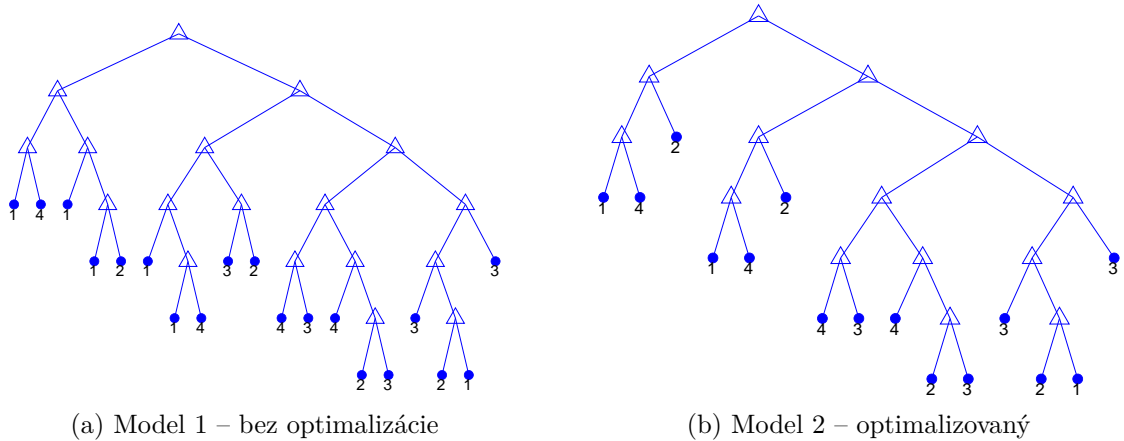
6.2 Rozhodovacie stromy

Otestovali sme viaceré metódy, menovite binárne rozhodovacie stromy a zoskupenia stromov. Pre CART sme využili funkciu `fitctree` a vyskúšali sme dva varianty tréovania. Prvá bola založená len na vstupných dátach a ich zaradení, bez špecifikácie iných parametrov, nastavené zostali implicitné hodnoty funkcie. V druhom variante sme zvolili možnosť funkcie optimalizovať hyperparametre, menovite: maximálny počet delení stromu (`MaxNumSplits`), rozdeľovacie kritérium (`SplitCriterion`), minimálny počet pozorovaní, ktoré list musí zaklasifikovať (`MinLeafSize`) a počet náhodne vybraných prediktorov použitých pre jednotlivé delenia (`NumVarToSample`). Po tréovaní s povolenou optimalizáciou boli ako optimálne parametre vybrané odlišné hodnoty od implicitných, tieto modely porovnávame v tabuľke 6.1.

Tab. 6.1: Nastavenia parametrov CART modelu

optimal. param.	MinLeafSize	MaxNumSplits	SplitCriterion	NumVarToSample
nie	1	256	Gini index	330
áno	4	38	Gini index	330

Hlavným rozdielom medzi dvoma modelmi je veľkosť, ktorú môže strom nadobudnúť vďaka obmedzenému počtu delení. Ďalej je viac pravdepodobné, že druhý



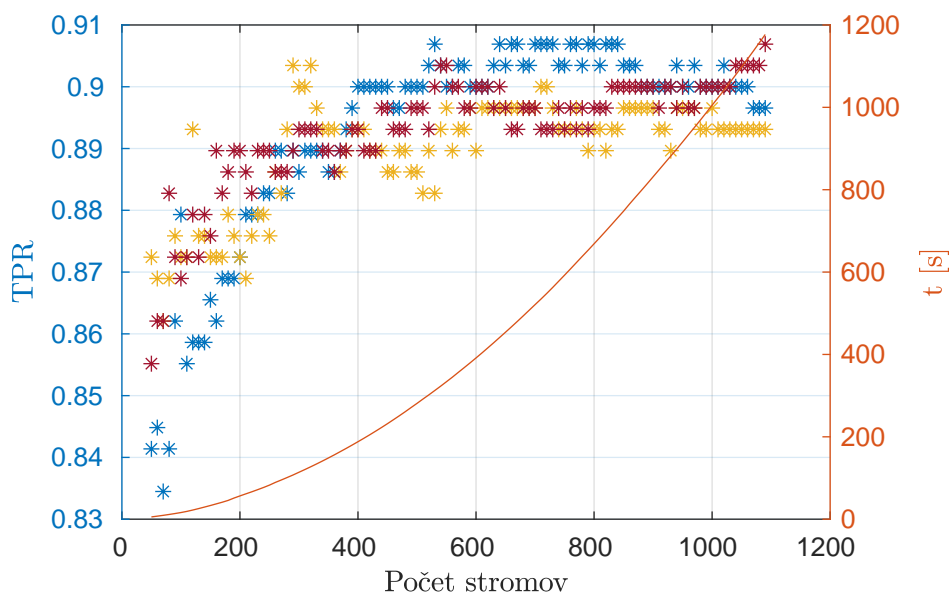
Obr. 6.3: Rozdiely v štruktúre rozhodovacích stromov na základe ich zvolených hyperparametrov.

model bude obsahovať menej listov vďaka orezávaniu ciest klasifikujúcich menej ako 4 vstupy, čo môžeme pozorovať aj na obrázku 6.3. Značná nevýhoda v optimalizácii sa vyskytla v čase potrebnom na tréningovanie, zatiaľ čo sa výsledok klasifikácie zmenil len zanedbateľne, viď Tab. 6.2.

Keďže binárny strom sa nepreukázal ako vhodný na takýto druh vstupných príznakov, otestovali sme taktiež náhodný les za pomoci funkcie `TreeBagger` a boosting s funkciou `fitcensamble`. V prípade náhodných lesov sme mali možnosť ladit parameter počtu stromov, ktoré budú tvoriť výsledný les. Keďže výber podmnožiny je v algoritme náhodný, aj výsledky sú odlišné pre rovnaké nastavenia. Preto sme rovnaký experiment spustili niekoľkokrát pre interval hodnôt 50 – 1090 stromov s krokom 10, aby sme overili prípadnú konvergenciu hodnoty spoľahlivosti a odhalili čo najvhodnejší potrebný počet stromov.

Ako je zrejmé z obrázka 6.4, TPR klasifikátora sa od 200 stromov pohybuje v relatívne malom okolí výsledkov a svojho maxima dosahuje na $\approx 90\%$. Ako najpriaznivejší interval sa javí 550 až 650 stromov, kde sa pohybuje v blízkosti maximálnej TPR pre tento klasifikátor. Zahrnutie vyššieho počtu stromov nepovažujeme za vhodný, pretože aby sme sa opäť dostali do stabilnejšej oblasti výsledkov, veľkosť lesa sa zväčší dvojnásobne. Tým nie len zvyšujeme komplexitu modelu, ale aj časovú náročnosť na výpočet, čo je cena za minimálne a nezaručené, prípadne žiadne zlepšenie. Preto na finálny výpočet uvedený v tabuľke 6.2 využijeme 620 stromov.

Čo sa týka stromov učných technikou Adaboost, otestovaná variácia hodnôt ovplyvňovala maximálny počet povolených učiacich cyklov. Tento parameter potom slúžil

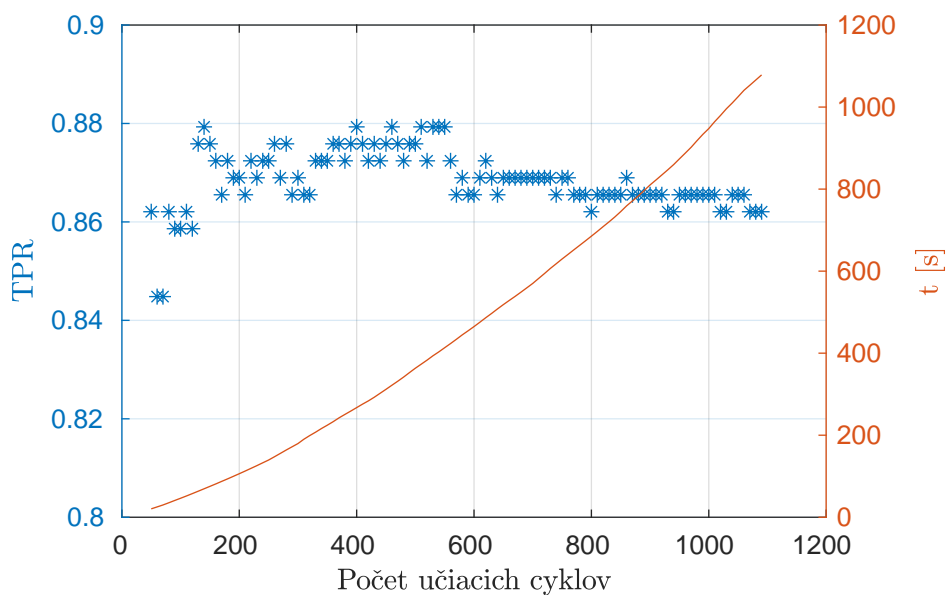


Obr. 6.4: Výsledky TPR pre opakovaný experiment výberu počtu stromov do zoskupenia náhodného lesa.

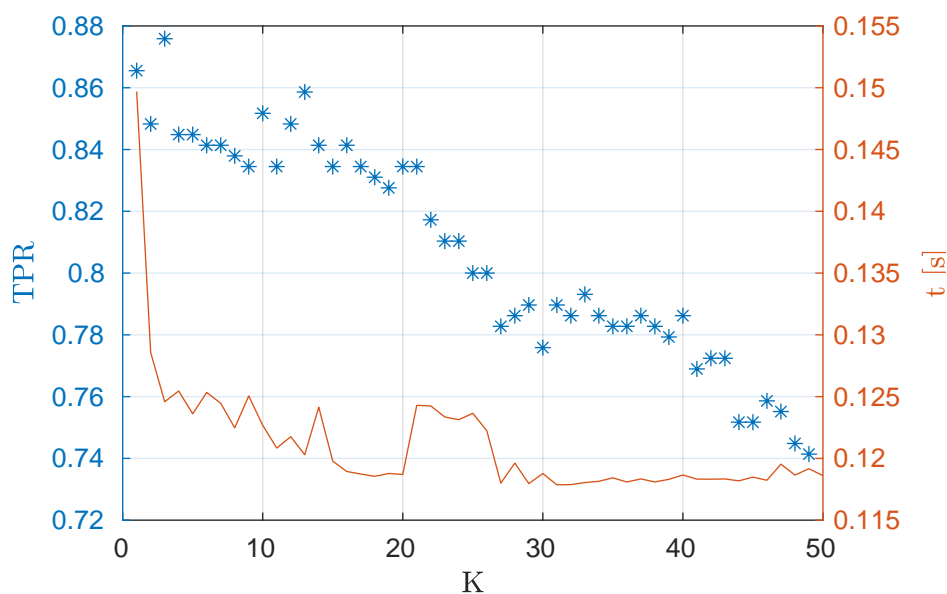
aj ako ukončovacia podmienka v prípade, že sa všetky vstupné prvky nepodarilo klasifikovať do správnej triedy. Testovaný interval zhŕňal 50 – 1090 iterácií, viď obrázok 6.5. Keďže sa spoľahlivosť klasifikácie po prekročení približne 500 cyklov začala zhoršovať, nemá význam uvažovať tieto nastavenia, čo je vyhovujúce aj z pohľadu času potrebného na tréning vysokého množstva stromov. Ako porovnávací výsledok bude slúžiť model s maximálne 250 iteráciami.

6.3 KNN

U klasifikátora KNN bol variabilný parameter počet susedov, ktorí mali vplyv na výslednú spoľahlivosť klasifikácie. Jeho výber prebehol v sérii experimentov pre hodnoty $K = 1, \dots, 50$, kde opäť rozhodujúci faktor bola celková hodnota TPR po krížovej validácii. Časová náročnosť bola vo všetkých prípadoch nízka a líšila sa minimálne, preto sme ju nemuseli zohľadňovať s takou váhou, ako v predchádzajúcich podkapitolách. Trend výsledných TPR mal klesajúci charakter, čo značí, že zohľadňovanie väčšieho množstva príznakov nie je vhodný prístup pri tomto type dát. Najpriaznivejších výsledkov sa dosiahlo v prípade $K = 3$, po ktorom nastal výrazný pokles v presnosti a tento model je tiež použitý v súhrnnej tabuľke výsledkov 6.2.



Obr. 6.5: Výsledky TPR boosting metódy pri zvyšovaní maximálneho povoleného počtu učiacich iterácií.



Obr. 6.6: Priebeh testovania vhodného počtu susedov KNN algoritmu spolu s ich časom výpočtu.

6.4 Zhodnotenie výsledkov

Z výsledkov uvedených klasifikátorov možno ako prvé prehlásiť, že binárny rozhodovací strom nie je vhodnou alternatívou na tento typ klasifikačného problému. So svojou TPR hodnotou len 60 % je tak ďaleko pod spoľahlivosťou ostatných menova-

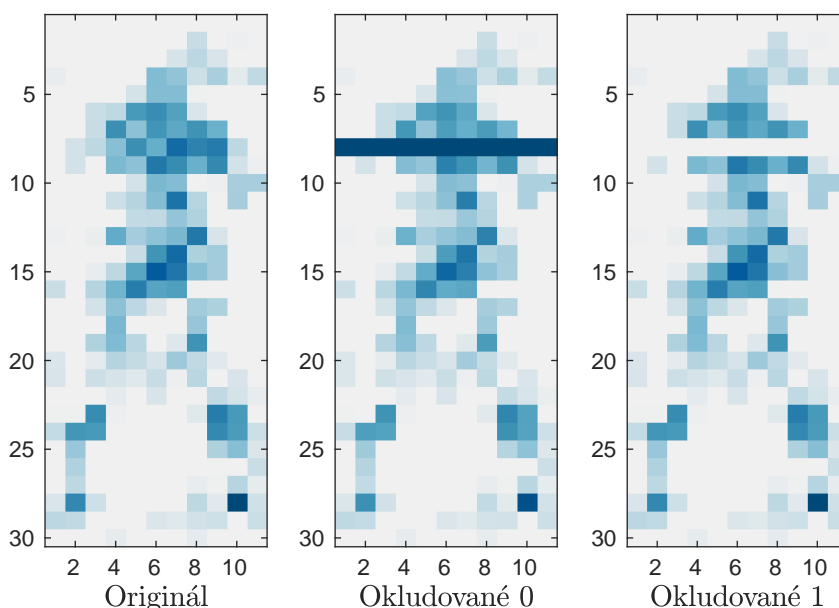
ných klasifikátorov. Na rovnakej úrovni skončili boosting a KNN modely z pohľadu ich presnosti, no výpočtová náročnosť sa u nich diametrálne odlišuje, kde KNN získava náskok a to pred všetkými metódami. O najlepšiu výkonnosť sa delí SRC s CNN. U nich však nenastáva len výrazný rozdiel v tréningovom čase, ale aj v celkovej zložitosti návrhu. CNN má množstvo parametrov, ktoré ovplyvňujú jeho finálnu spoľahlivosť. Preskúmanie čo i len malého podpriestoru ich kombinácií zabralo desiatky hodín, preto uvedené TPR je lokálne optimum, ktoré sa nám podarilo dosiahnuť. Na druhej strane SRC vyžaduje ladenie jediného regularizačného parametra, ktorého vhodnú hodnotu je možné nájsť rýchlo a jednoducho.

Tab. 6.2: Úspešnosti testovaných klasifikátorov a doba výpočtu predikcie

Klasifikačná Metóda	TPR [%]	Časová náročnosť [s]
SRC, MC	83,10	2,54
SRC, MSCC	92,76	2,26
SRC, MCR	92,76	2,35
CNN	92,41	132,94
Náhodný les	90,00	62,92
KNN, $K = 3$	87,59	0,18
Boosting klasifikačné stromy	87,24	97,49
CART	60,69	3,32
CART, optim. hyperparametre	60,34	198,21

7 Oklúzie

Vo všetkých reálnych systémoch, aplikáciách a prístrojoch sa môžu vyskytnúť nečakané chyby či poruchy. Tie vo všeobecnosti vedia nadobúdať rôzny charakter v závislosti od situácie. V našom prípade zlyhanie jedného z tlakových senzorov matracu spôsobí výpadok celého riadka tlakovej matice, čo ovplyvní výstupný obraz, ako aj obtiažnosť jeho následnej klasifikácie. Vznik porúch je často nepredvídateľný a môže zasiahnuť akúkoľvek časť obrazu. Simuláciu jednej z možností výskytu oklúzie ilustrujeme na obrázku 7.1.



Obr. 7.1: Ukážka výskytu oklúzie v obraze spôsobenej zlyhaním senzoru v riadku 8.

Keďže oklúzie prevažne porušia len malý podiel obrazu (jeden riadok predstavuje približne 3,3 % z jeho celkovej veľkosti), preto možno povedať, že jeho reprezentácia v LASSO koeficientoch je riedka. Potom na spracovanie dát, v ktorých možno očakávať aj poškodené merania, aplikujeme rozšírený slovník [25]. Ten obsahuje varianty predpokladaných porúch vo forme matice \mathbf{O} , kde každý stĺpec matice reprezentuje odhadovanú možnú oklúziu (výpadok dát v riadku). Nie je však potrebné vytvárať zastúpenie v rámciach pre kombinácie pôvodných obrazov a poškodených riadkov, s tým sa vysporiadava riedka reprezentácia [13]

$$\mathbf{X}^* = [\mathbf{X} \ \mathbf{O}] = [\mathbf{x}_1 \ \mathbf{x}_2 \ \dots \ \mathbf{x}_n \ \mathbf{o}_1 \ \dots \ \mathbf{o}_{30}] \quad (7.1)$$

Hodnoty možných oklúzií budeme uvažovať 0 alebo 1 v celkom riadku. Takýmto rozšírením datasetu nadobudne jeho veľkosť 320 meraní. V prípade ostatných klasifikátorov však nestačí iba pridať samotný obraz šumu, pretože nevedia kombinovať

vstupy ako LASSO regresia, ktorá priradí mieru zastúpenosti prvku v obraze. Preto je potrebné vytvoriť kombinácie pôvodných obrazov a pridaného šumu, čím rozšírime dataset na 8990 meraní.

7.1 Klasifikácia okludovaných dát

Všetky klasifikátory sme ponechali s nastaveniami z kapitoly 6 a vykonali sme sériu testov, v ktorých sme pozorovali zmeny úspešnosti klasifikácie v závislosti od výskytu oklúzií v tréningových alebo testovacích dátach.

Tab. 7.1: Výsledky klasifikácie s výskytom oklúzií typu 1

Klasifikačná Metóda	TPR [%]		
	trénované bez oklúzií	trénované s oklúziami	
	okludované testovacie dáta	pôvodné testovacie dáta	okludované testovacie dáta
SRC, MC	83,33	83,10	83,33
SRC, MSCC	91,81	92,76	91,81
SRC, MCR	91,81	92,76	91,81
CNN	91,12	92,41	93,40
KNN, $K = 3$	85,58	86,21	86,08
Náhodný les	88,88	92,07	91,35
Boosting stromy	85,66	91,03	89,73
CART	58,69	61,38	60,99
CART optim. hyperparam.	53,38	64,14	61,41

V prvom stĺpci si môžeme všimnúť najviac kritickú situáciu, s ktorou by sme sa mohli stretnúť a to predložiť klasifikátoru poškodené dáta, na ktoré nebol pri tréningu pripravený. Ako prvé pozorujeme, že napriek zavedeniu šumu do slovníka pre SRC, nie je na tom závislá kvalita jeho klasifikácie, napriek tomu, že bolo očakávané zlepšenie. Vieme z toho ale vyvodiť, že na rovnako dobrú klasifikáciu nám postačí aj pôvodný dataset, čím skracujeme výpočtový čas regresie. Na druhej strane vidíme výrazné zlepšenie pre modely zhlukov stromov, kde v prípade náhodných stromov dosahujeme dokonca TPR len o niečo nižšie, ako u SRC. Avšak toto zlepšenie bolo sprevádzané vysokým nárastom tréningovej doby na 50 minút, z pôvodnej 1 minúty. Tento nárast sme pozorovali pri boosting metóde. Len v jednom prípade sa podarilo výsledkom prekonať SRC, a to pri tréningu i testovaní CNN modelu s oklúziami. Ak

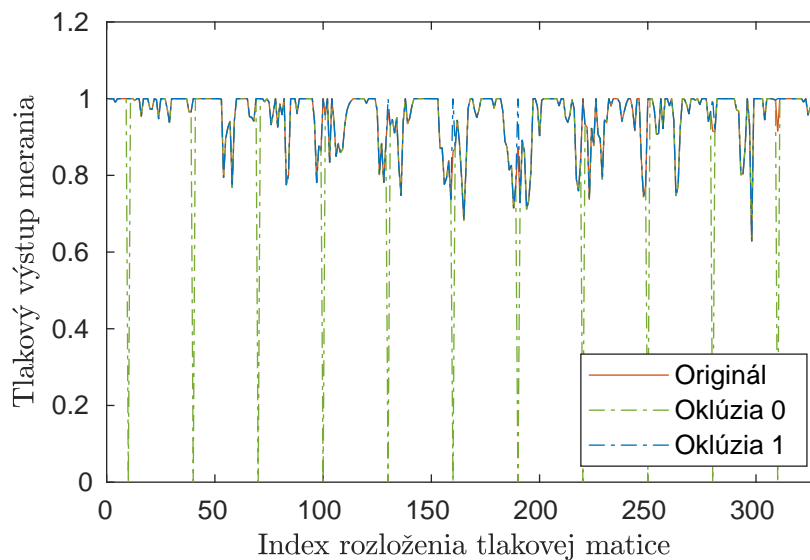
budeme uvažovať možnosť, kedy chceme model odolný voči poruchám a porovnáme SRC s iným najlepším klasifikátorom, vyberieme CNN sieť. Keďže SRC má rovnakú presnosť klasifikácie aj bez tréningu s oklúziami, pracujeme so slovníkom veľkosti 290 meraní a testovacím datasetom 8990 meraní. V prípade CNN je potrebných 8990 učiacich i testovacích dát. Dôsledkom toho sú veľké rozdiely v dobe tréningu, pre SRC 67 s a pre CNN 1 hodina, so ziskom v presnosti o 1,59 %.

Tab. 7.2: Výsledky klasifikácie s výskytom oklúzií typu 0

Klasifikačná Metóda	TPR [%]		
	trénované bez oklúzií	trénované s oklúziami	
	okludované testovacie dáta	pôvodné testovacie dáta	okludované testovacie dáta
SRC, MC	70,08	83,10	70,67
SRC, MSCC	76,26	92,76	76,23
SRC, MCR	76,26	92,76	76,23
CNN	60,85	92,41	90,65
KNN, $K = 3$	72,48	86,21	85,73
Náhodný les	87,86	87,59	87,05
Boosting stromy	85,07	90,00	86,89
CART	56,78	65,17	60,51
CART optim. hyperparam.	59,53	69,66	59,88

Z prvého stĺpca tabuľky 7.2 vidíme, že všetky prístupy klasifikácie, okrem zhlukov rozhodovacích stromov, nevedeli dobre zvládnuť situáciu výskytu oklúzie nulami. Keďže 0 reprezentuje úplné stlačenie tlakovej podložky, teda signál obsahuje krátke pulzy, vytvárame tým omnoho výraznejšiu chybu ako v prípade oklúzie jednotkou, viď obrázok 7.2. SRC si však pri regularizácii $\lambda = 7 \cdot 10^{-4}$ nevie s týmto problémom dobre poradiť ani po zaradení porúch do učiaceho slovníka a jej výsledky prekonávajú ostatné modely, okrem CART, s veľkým náskokom. Pri tréningu a testovaní na okludovaných dátach si dokonca pohoršila o 0,03 %, preto sme vykonali dodatočný experiment pre zistenie optimálnej hodnoty regularizácie pri práci s týmto typom oklúzie, tabuľka 7.3. Ukázalo sa, že malou zmenou λ na hodnotu 10^{-4} vieme stále dostatočne penalizovať výsledný model LASSO regresie, a zároveň výrazne zlepšiť jeho schopnosť klasifikácie. Síce sa klasifikácia v ostatných prípadoch znížila o 2 %, získali sme všeobecne stabilný klasifikátor, ktorý vie dobre pracovať ako s poškodenými tak aj s originálnymi dátami, s oklúziami zahrnutými v slovníku. Ďalším

zmenšováním λ sme už však pozorovali zhoršovanie vo všetkých výsledkoch. V takom prípade je výhodné, ak máme znalosť o charaktere vyskytujúcich sa oklúzií, a teda vieme zvoliť vhodnú alternatívu nastavení ladiacich parametrov.



Obr. 7.2: Rozdiel v poruchách z pohľadu jednorozmerného signálu

Tab. 7.3: Zmena regularizačného parametra za účelom zlepšenia klasifikácie s oklúziami typu 0

Klasifikačná metóda	TPR [%]			
	trénované bez oklúzií		trénované s oklúziami	
	pôvodné testovacie dáta	okludované testovacie dáta	pôvodné testovacie dáta	okludované testovacie dáta
SRC, MC	84,14	69,87	84,14	82,53
SRC, MSCC	90,69	75,94	90,69	90,59
SRC, MCR	90,69	75,95	90,69	90,58

7.2 Rekonštrukcia poškodeného obrazu

Kvalita klasifikácie SRC závisela na druhu poruchy, ktorú obraz obsahoval, preto preskúmame tiež možnosť obnovy obrazu v týchto situáciách. V tomto prípade však

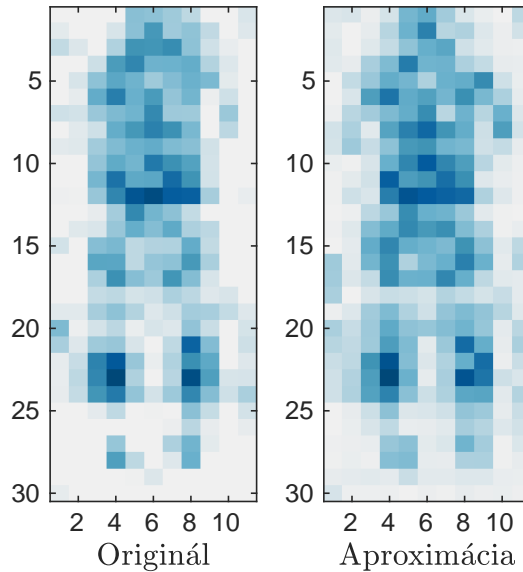
nebudú koeficienty rozdeľované do tried a výsledný obraz zložíme zo všetkých meraní, ktoré budú vyhodnotené ako relevantné. Keďže výsledkom LASSO regresie je aproximácia vstupného merania na základe dostupného učiaceho slovníka, sú v koeficientoch zahrnuté aj tie merania, ktoré reprezentujú samotné oklúzie v obraze. Preto je pri rekonštrukcii poškodených dát potrebné odstrániť koeficienty, ktoré tieto oklúzie reprezentujú, čím zabezpečíme, že vytvorená rekonštrukcia už poruchy obsahovať nebude.

Keďže v predchádzajúcej časti sa výrazne odlišovali výsledky pre typ oklúzie, obrázok 7.4 znázorňuje aproximáciu oboch prípadov. Kvalita rekonštrukcie obrazu je vyjadriteľná pomocou viacerých spôsobov. My ju budeme posudzovať prostredníctvom pomeru signál-šum, ďalej ako SNR (angl. signal-to-noise ratio) medzi originálnym a aproximovaným obrazom, kde zvyšovaním pomeru sa zlepšuje kvalita signálu a šum je viac potláčaný. Existuje viacero pohľadov, na základe ktorých sa dá SNR definovať. V prípade fotografií a digitálneho obrazu sa stretávame s pomerom počtu elektrónov a korešpondujúcim šumom signálu dopadajúcich na senzor kamery [18], pri účeloch obrazovej dekonvolúcie figurujú vo vzájomnom vzťahu smerodajná odchýlka obrazu a šumu [19]. V prípade signálovej analýzy a odšumovania nájdeme vyjadrenie pomocou energií signálu a šumu [11]. Keďže môžeme uvažovať o našich meraniach ako o obrazoch, tak aj o jednorozmerných signáloch je potrebné vzhľadom na situáciu a charakter meraní zvoliť vhodný SNR vzťah. Taktiež je potrebné si uvedomiť, že SNR je bezrozmerná veličina, ktorá sama o sebe nemá veľkú výpovednú hodnotu a treba k zhodnoteniu výsledku uvažovať situáciu, v ktorej vznikala a k čomu je výsledok relevantný [6]. Preto každá aplikácia môže uvádzať rozdielne hodnoty SNR vyjadrujúce výbornú alebo nízku kvalitu signálu. Po uvážení týchto faktorov sme SNR uvažovali v nasledujúcej forme, ako ju uvádza [11].

$$\text{SNR}_{\text{dB}} = 10 \log_{10} \left(\frac{\|\mathbf{Y}\|^2}{\|\mathbf{Y} - \hat{\mathbf{Y}}\|^2} \right) \quad (7.2)$$

Dodatočným hodnotiacim prvkom bolo zvolené reziduum 4.5, na základe ktorého bolo stanovené orientačné zhodnotenie SNR výsledkov. Zároveň sme preskúmali možnosť obnovy obrazu obsahujúceho viac ako jednu riadkovú poruchu. Číselné vyhodnotenie je spísané v tabuľke 7.4. Na experiment bolo využité meranie č. 67, obrázok 7.3 (subjekt č. 3, poloha na chrbte), pre snahu použiť obraz s čo najnižším postranným šumom okolo postavy. Grafické znázornenie experimentu zachytáva obrázok 7.4.

Pre oba typy poruchy sme pridávali počet vyskytujúcich sa oklúzií, konkrétne 1, 3 až 5 chybných riadkov, ktoré sa postupne objavili na riadkoch 4, 11, 15, 20 a 27. Umiestňované boli so zámerom postihnúť dôležité miesta obrazu, ako hlava, ramená, oblasť drieku a panvy a chodidlá, ktoré majú veľkú výpovednú hodnotu.



Obr. 7.3: Rekonštrukcia merania č. 66 a jeho porovnanie s nepoškodeným vstupom.

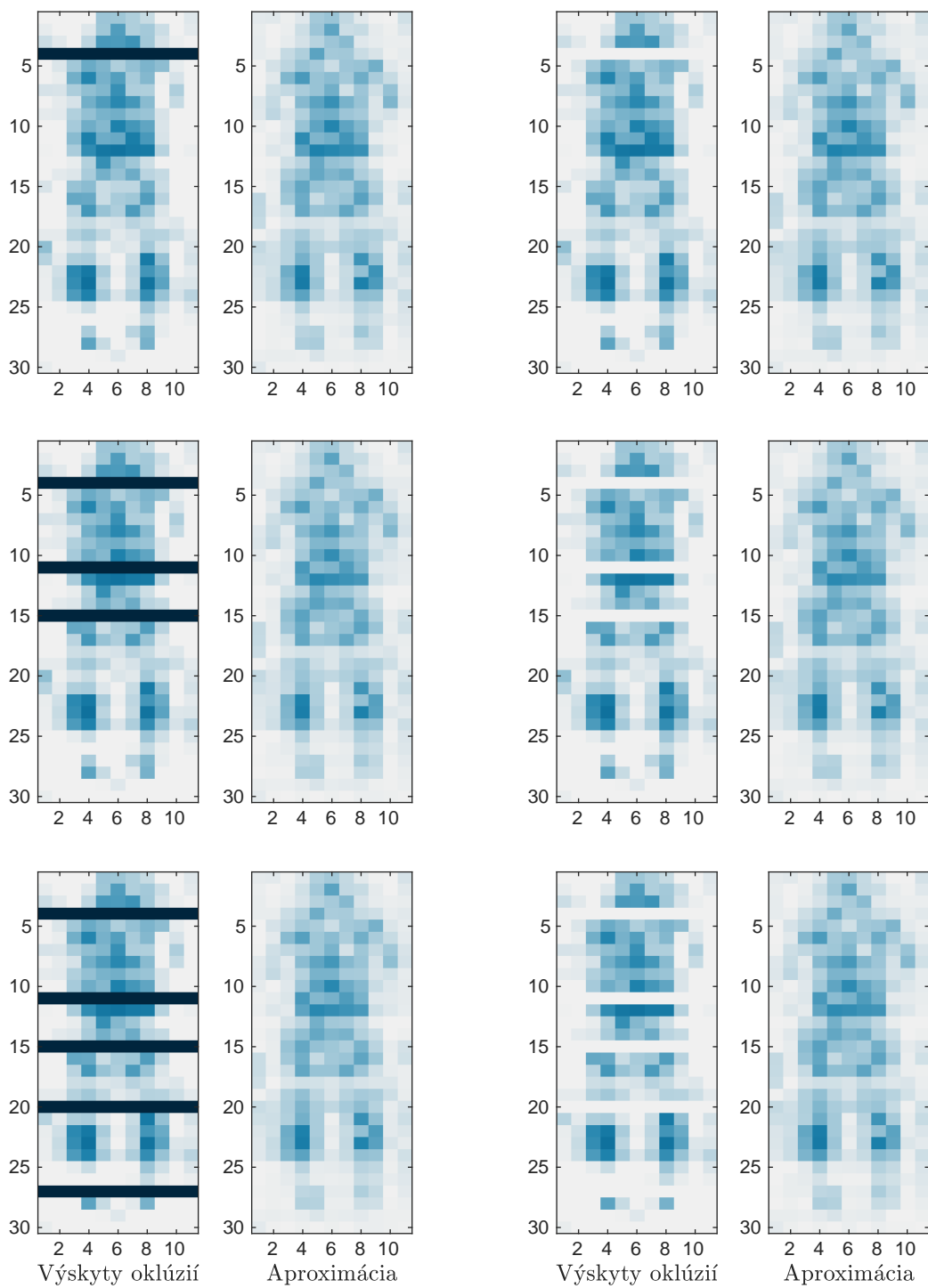
Ako prvé sme vyjadrili SNR pre aproximáciu z originálneho obrazu, aby sme vedeli zhodnotiť ostatné prípady, pretože vytváranie rekonštrukcie tiež vnáša zašumenie do obrazu, viditeľné v 7.3. Keďže vstupné meranie obsahuje okolitý šum mimo osoby, ako aj merania, ktoré sú súčasťou slovníka, sú to hodnoty, ktoré sú tiež do určitej miery aproximované. Jeho zvyšovanie nastáva, ak subjekty, ktoré dobre aproximujú vstupný obraz, obsahujú aj šum nachádzajúci sa mimo pozície v originálnom obraze a po svojom skombinovaní môžu vytvárať výraznejšie zašumený priestor na matraci. SNR aproximácie originálu nadobudlo hodnoty 25,56 dB a reziduum medzi pôvodným a rekonštruovaným obrazom je $Q = 0,06$ a teda aproximáciu považujeme za dobrú.

Po vyčíslení SNR pre zvyšné situácie sa ukázalo, že typ oklúzie má opačný vplyv na rekonštrukciu v porovnaní s klasifikáciou. Keďže pridanie jednotiek nemusí byť vyhodnotené ako poškodenie, berúc do úvahy tiež pozadie okolo subjektu, na ktoré sa nevyvíja tlak a teda obsahuje hodnoty blízke alebo rovné 1, a teda v porovnaní s originálnym meraním sa nachádza poškodenie len tam, kde prechádza cez telo postavy. Na druhej strane nulové hodnoty predstavujú v každom prípade závažnú zmenu. Toto sa odráža aj na dosiahnutých výsledkoch. Ak porovnáme oklúzie zasahujúce 1 riadok, vidíme výrazný pokles v kvalite signálu pre poškodenie 0 oproti poškodeniu 1, kde dokonca došlo k zlepšeniu kvality signálu v porovnaní s aproximáciou z obrázka 7.3. Odstránenie oklúzií bolo v oboch prípadoch veľmi úspešné, hodnoty SNR poklesli len o jeden decibel, taktiež rozdiel aproximácie od originálneho obrazu zostal na hodnote $Q \approx 0,06$. Zvyšné prípady poruchy nulou rýchlo klesali na hodnotu SNR a pre 5 okludovaných riadkov stúpila hodnota reziduí sedemkrát, no aj

Tab. 7.4: Zhodnotenie potlačenia šumu v rôznych prípadoch poškodenia obrazu s referenčnou hodnotou SNR pre aproximovaný čistý obraz bez porúch.

	SNR [dB]	
Bez oklúzií	25,56	
Oklúzia 0	pred potlačením šumu	po potlačení šumu
1 riadok	15,10	24,42
3 riadky	10,23	24,15
5 riadkov	7,78	24,06
Oklúzia 1		
1 riadok	28,64	24,72
3 riadky	24,05	18,76
5 riadkov	23,17	18,61

napriek tomu sa regresný model vedel vysporiadať s touto situáciou a udržal kvalitu potlačenia šumu na takmer rovnakej úrovni. Naproti tomu spodná polovica tabuľky 7.4 má o niečo nižšiu úspešnosť udržania si kvality signálu, stále však nemožno hovoriť o zlej kvalite aproximácie, pretože pre 3 a 5 poškodených riadkov sa pohybuje reziduum v okolí $Q \approx 0,14$. Ako môžeme pozorovať na 7.4, medzi aproximovanými signálmi voľným okom nie sme schopní rozlíšiť rozdiely. Preto bude závisieť aj od zámeru obnovy obrazu, čo považovať za kvalitný signál a čo už nie. Je zrejmé, že na použitie v klasifikácii budú na SNR a Q kladené vyššie nároky, oproti práci, kde by bolo potrebné manuálne spracovanie. Preto treba dbať na fakt, že SNR je hodnota subjektívna v rôznych situáciách. Taktiež je nutné zohľadniť, že hodnoty uvedené v tabuľke 7.4 nie sú platné všeobecne naprieč celým datasetom. V takom prípade bude tiež závisieť od rovnomernosti zastúpenia rôznych pozícií, ktoré sú k dispozícii v učiacom slovníku, čím je priamo ovplyvňovaná kvalita aproximácie. Ak však disponujeme kvalitne zhotoveným datasetom, vieme povedať, že LASSO regresia je schopná veľmi dobre pracovať s okludovanými dátami a ich spätnej rekonštrukcie.



(a) Poškodenie typu 0

(b) Poškodenie typu 1

Obr. 7.4: Obnovenie obrazu LASSO regresiou pre rôzne stupne porušenia nulovými a jednotkovými riadkami.

Záver

V práci sme sa zamerali na aplikovanie riedkej reprezentácie pomocou redundantných slovníkov na dataset obsahujúci merania polohy ležiaceho človeka na tlakocitlivom matraci. Na základe tejto reprezentácie sme otestovali možnosť klasifikácie do tried označujúcich polohy človeka. Po uvedení do teoretických aspektov problematiky v prvej kapitole, sme predostreli kľúčové rozdiely medzi často využívanou MNŠ a metódou LASSO. Najdôležitejšie vlastnosti, ktorými LASSO disponuje, sú robustnosť a nižší počet relevantných koeficientov, čo v konečnom dôsledku znamená aj menej komplexný model a lepšiu interpretovateľnosť.

Následne sme sa v ďalšej kapitole zamerali na oboznámenie s dostupnými meraniami a nájdenie vhodného parametru λ , ktorý určil mieru riedkosti, s ktorou sme pracovali v rámci výpočtového algoritmu. λ sme stanovili na základe pomeru veľkosti chyby modelu a časovej náročnosti získania koeficientov.

V programovom prevedení sme vytvorili na základe nadobudnutých poznatkov algoritmus, ktorý dataset postupne rozdeľuje na tréningové a testovacie skupiny, z ktorých sme postupne získavali odhadované modely pomocou LASSO funkcie a výsledky sme ukladali na neskoršiu klasifikáciu. Taktiež sme si predstavili niekoľko riedkych reprezentácií konkrétnych meraní, kde jasne vidieť nízky počet koeficientov a poslúžili aj na ujasnenie predstavy o fungovaní klasifikátorov.

V štvrtej kapitole sme pristúpili k samotnému roztriedovaniu do tried, na základe troch klasifikačných kritérií - Maximálny koeficient (MC), Maximálna suma koeficientov triedy (MSCC) a Minimálne reziduum triedy (MCR). Tu sme odhalili nevhodnosť využitia MC, ktoré sa ukázalo ako najviac nespoľahlivé s mierou úspešnosti len 83,1%. Na druhej strane sa MSCC a MCR preukázali ako rovnocenné metódy, s mierou úspešnej klasifikácie až 92,8%. Na záver sme tiež predostreli možnosti výberu jednej z prístupov na základe aplikáciu stanovených podmienkach.

Nasledujúce kapitoly predstavili klasifikátory a postup pri ich návrhu, ktoré sme porovnávali so spoľahlivosťou klasifikácie riedkej reprezentácie. Ako najúspešnejšia sa preukázala SRC a hneď po nej CNN. Po uvážení komplexnosti návrhu CNN, rýchlosti tréningovania a validácie a taktiež výpočtovej náročnosti, sme zvolili SRC ako ďaleko vhodnejšieho kandidáta, vďaka svojim nižším nárokom na návrh, a výpočet a taktiež zrozumiteľnosť a interpretovateľnosť modelu.

Posledná kapitola sa zameriavala na prácu s oklúziami, kde sme opäť preskúmali schopnosť klasifikovať tieto dáta. Pri oklúziách jednotkou sa ukázalo, že vo všetkých prípadoch okrem jedného zostala SRC neporazeným klasifikátorom. CNN zlepšila svoj výsledok o 1,5 %, no za cenu 31-krát zväčšeného datasetu a 60-krát vyššiemu času výpočtu oproti SRC. Na problémy narazila SRC metóda pri poškodení nulami a najvhodnejším otestovaným klasifikátorom bola CNN sieť. Túto situáciu sa

nám podarilo zlepšiť zvolením vhodnejšej regularizácie, čím sme vytvorili klasifikátor so všeobecne stabilnou presnosťou. V závere práce sme preskúmali tiež možnosť rekonštrukcie dát obsahujúcich oklúzie, kde sme odhalili dobrú schopnosť riedkej reprezentácie aproximovať signály aj s vyšším stupňom porušenia obrazu.

Celkovo sa redundantné slovníky a riedka reprezentácia signálu ukázala ako vhodná cesta v prípade klasifikácie obrazu tohto typu, ako kvôli dosiahnutým výsledkom, tak aj svojej jednoduchosti návrhu. V ďalšej práci by sme chceli preskúmať širšie možnosti aplikácie LASSO regresie a to ako metódu predspracovania dát pred klasifikáciou, alebo na hľadanie a určovanie vhodných príznakov ako vstup pre iné klasifikátory.

Literatúra

- [1] BODITRAK. *ReST*. Citované: 12.5.2023. Dostupné z: https://www.boditrak.com/?fbclid=IwAR3dWr3ZelAFsDt_AY1BtxhIpWk4isruOMRwUD2KwcnCMk48XP0Ga2ntUx4.
- [2] BREIMAN, L., FRIEDMAN, J. H., OLSHEN, R. A. a STONE, C. J. *Classification and Regression Trees*. 1st. ed. Routledge, 1984. ISBN 978-0-421-04841-8.
- [3] BRUNTON, S. a KUTZ, J. N. *Data-Driven Science and Engineering: Machine Learning, Dynamical Systems, and Control*. Cambridge: Cambridge University Press, 2019. ISBN 978-1-108-42209-3.
- [4] FAWCETT, T. An introduction to ROC analysis. *Pattern Recognition Letters*. 2006, zv. 27, č. 8, s. 861–874. DOI: <https://doi.org/10.1016/j.patrec.2005.10.010>. ISSN 0167-8655. ROC Analysis in Pattern Recognition.
- [5] FRIEDBERG, S., INSEL, A. a SPENCE, L. *Linear Algebra*. 4th ed. Pearson Education, 2003. Featured Titles for Linear Algebra (Advanced) Series. ISBN 978-0-13-008451-4.
- [6] GREGSON, P. *How is SNR calculated in images?* Január 2017. Citované: 14.5.2023. Dostupné z: https://www.researchgate.net/post/How_is_SNR_calculated_in_images/5873bd4c3d7f4b574240a31b/citation/download.
- [7] HASTIE, T., TIBSHIRANI, R. a FRIEDMAN, J. H. *The elements of statistical learning: data mining, inference, and prediction*. 2nd ed. New York: Springer, c2009. ISBN 978-0-387-84857-0.
- [8] INC., S. P. *Tactilus Temperature Body Mapping System*. Citované: 12.5.2023. Dostupné z: https://www.sensorprod.com/dynamic/temperature_mapping.php.
- [9] KOVACEVIC, J. a CHEBIRA, A. An Introduction to Frames. *Foundations and trends in signal processing*. 2008, zv. 2, č. 1, s. 1–94. DOI: 10.1561/20000000006. Dostupné z: https://www.researchgate.net/publication/220066972_An_Introduction_to_Frames.
- [10] LIU, J. J. e. a. Sleep posture analysis using a dense pressure sensitive bedsheet. *Pervasive and Mobile Computing*. 2014, zv. 10, s. 34–50. DOI: <https://doi.org/10.1016/j.pmcj.2013.10.008>. ISSN 1574-1192.
- [11] MALLAT, S. *A wavelet tour of signal processing : the sparse way*. 3rd ed. Boston : Amsterdam: Academic Press ; Elsevier, 2009. ISBN 978-0-12-374370-1.

- [12] MARSLAND, S. *Machine learning: an algorithmic perspective*. Boca Raton: CRC Press, 2009. ISBN 978-1-4200-6718-7.
- [13] MIHÁLIK, O., SÝKORA, T., HUSÁK, M. a FIEDLER, P. In-Bed Posture Classification Based on Sparse Representation in Redundant Dictionaries. *IFAC-PapersOnLine*. 2022, zv. 55, č. 4, s. 374–379. DOI: <https://doi.org/10.1016/j.ifacol.2022.06.062>. ISSN 2405-8963. Dostupné z: <https://www.sciencedirect.com/science/article/pii/S2405896322003779>.
- [14] MURPHY, K. P. *Machine learning: a probabilistic perspective*. Cambridge: MIT Press, 2012. ISBN 978-0-262-01802-9.
- [15] RAITOHARJU, J. Convolutional Neural Network. In: *Deep Learning for Robot Perception and Cognition*. Academic Press, 2022, s. 35–69. ISBN 978-0-323-85787-1.
- [16] RILEY, K. F. a HOBSON, M. P. *Essential Mathematical Methods for the Physical Sciences*. Cambridge University Press, 2011. ISBN 978-0-521-76114-7.
- [17] ROKACH, L. a MAIMON, O. Classification Trees. In: *Data Mining and Knowledge Discovery Handbook*. Boston: Springer, 2010, s. 149–174. ISBN 978-0-387-09823-4. Dostupné z: https://doi.org/10.1007/978-0-387-09823-4_9.
- [18] ROWLANDS, A. *Physics of Digital Photography*. IOP Publishing, 2017. ISBN 978-0-7503-1242-4. Dostupné z: <https://dx.doi.org/10.1088/978-0-7503-1242-4>.
- [19] RUSS, J. C. *The Image Processing Handbook*. 6th. ed. Boca Raton: CRC Press, 2011. ISBN 978-1-4398-4063-4.
- [20] SICOTTE, X. B. *Ridge and Lasso: visualizing the optimal solutions* [https://xavierbourethsicotte.github.io/ridge_lasso_visual.html#Ridge-regression]. 2018. Citované: 18.3.2023.
- [21] STEHMAN, S. V. Selecting and interpreting measures of thematic classification accuracy. *Remote Sensing of Environment*. 1997, zv. 62, č. 1, s. 77–89. DOI: [https://doi.org/10.1016/S0034-4257\(97\)00083-7](https://doi.org/10.1016/S0034-4257(97)00083-7). ISSN 0034-4257.
- [22] TEKSCAN. *Body Pressure Measurement System (BPMS)*. Citované: 12.5.2023. Dostupné z: <https://www.tekscan.com/products-solutions/systems/body-pressure-measurement-system-bpms-research>.
- [23] TIBSHIRANI, R. Regression Shrinkage and Selection via the Lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*. [Royal Statistical

- Society, Wiley]. 1996, zv. 58, č. 1, s. 267–288. ISSN 00359246. Dostupné z: <http://www.jstor.org/stable/2346178>.
- [24] TING, K. M. Confusion Matrix. In: SAMMUT, C. a WEBB, G. I., ed. *Encyclopedia of Machine Learning*. Boston, MA: Springer US, 2010, s. 209. DOI: 10.1007/978-0-387-30164-8_157. ISBN 978-0-387-30164-8.
- [25] WRIGHT, J., YANG, A. Y., GANESH, A., SASTRY, S. S. a MA, Y. Robust Face Recognition via Sparse Representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2009, zv. 31, č. 2, s. 210–227. DOI: 10.1109/T-PAMI.2008.79.

Zoznam symbolov a skratiek

β Vektor koeficientov regresie

C_k Klasifikačná trieda

CART Klasifikačné a rozhodovacie stromy (Classification and regression trees)

CNN Konvolučná neurónová sieť (Convolutional neural network)

H Hilbertov priestor

KNN K najbližších susedov (K nearest neighbour)

λ Regularizačný parameter

LASSO Operátor najmenšieho absolútneho zmenšenia a výberu (Least absolute shrinkage and selection operator)

MC Maximálny koeficient

MCR Minimálne reziduum triedy

MNŠ Metóda najmenších štvorcov

MSCC Maximálny súčet koeficientov triedy

Q Reziduum

SRC Klasifikácia riedkej reprezentácie (Sparse representation classification)

TPR Miera úspešnosti klasifikácie (True positive rate)

X Všeobecná veličina – skalár, vektor, matica

x Vektor

X Matica

\hat{y} Odhadovaná veličina

8 Obsah elektronickej prílohy

Príloha okrem textu záverečnej práce obsahuje aj sprievodné skripty vytvorené v prostredí MATLAB 2022b. Skript `createData.m` slúži na vytvorenie vhodného formátu datasetu `bedData.mat`, ktorý obsahuje samotné merania a ich anotáciu podľa subjektov a polohy. V `sparseClassifiers.m` sa nachádza implementácia klasifikátorov spolu s možnosťou výberu typu klasifikátora zvolením premennej `classifier` z ponúkaných možností, pre ktorý bude kód spustený. Posledným nastaviteľným parametrom sú konfigurácie okludovaných a čistých dát pre tréning a validáciu – `occlusions_in_dictionary`, `occlusions_in_dada`, zvolením ich hodnôt na `true` alebo `false`.

Obsah CD prílohy:

1. Text práce
2. MATLAB:
 - `sparseClassification.m`
 - `createData.m`
 - `bedData.mat`