



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

**FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH
TECHNOLOGIÍ**

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV BIOMEDICÍNSKÉHO INŽENÝRSTVÍ

DEPARTMENT OF BIOMEDICAL ENGINEERING

**PRÍSTUPY UČENIA NEURONOVÝCH SIETÍ S
NEVYROVNANÝM DATASETOM**

IMBALANCED DATA TRAINING APPROACHES IN NEURAL NETWORK

BAKALÁRSKA PRÁCA

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

Veronika Vicianová

VEDÚCI PRÁCE

ADVISOR

Ing. Roman Jakubíček, Ph.D.

BRNO 2023

Bakalářská práce

bakalářský studijní program **Biomedicínská technika a bioinformatika**

Ústav biomedicínského inženýrství

Studentka: Veronika Vicianová

ID: 231170

Ročník: 3

Akademický rok: 2022/23

NÁZEV TÉMATU:

Přístupy učení neuronových sítí s nevyrovnaným datasetem

POKyny PRO VYPRACOVÁNÍ:

1) Prostudujte oblast strojového učení se zaměřením na neuronové sítě, možnosti učení a obecné problémy. 2) Proveďte průzkum publikací a vypracujte rešerši o možnostech eliminace vlivu nevyrovnaného datasetu během trénování na úspěšnost modelu. 3) Navrhněte praktické experimenty s vybranými metodami strojového učení a zvolenými přístupy eliminace tohoto vlivu. 4) Z veřejně dostupných datasetů vytvořte trénovací a validační množinu a připravte data pro trénování. Navrhněte a implementujte vhodný model strojového učení. 5) Realizujte navržené experimenty, dle potřeby modifikujte. Navrhněte vhodnou metodiku hodnocení jednotlivých experimentů. 6) Proveďte srovnání implementovaných přístupů, případně rozšířte na jiné datasety. Vyhodnoťte a diskutujte výsledky dosažené v experimentech a vyvoďte závěry s provedených experimentů.

DOPORUČENÁ LITERATURA:

[1] Johnson, J.M., Khoshgoftaar, T.M. Survey on deep learning with class imbalance. J Big Data 6, 27 (2019). Dostupné z: <https://doi.org/10.1186/s40537-019-0192-5>

[2] Kumar P, Bhatnagar R, Gaur K, Bhatnagar A. Classification of Imbalanced Data: Review of Methods and Applications. {IOP} Conf Ser Mater Sci Eng [Internet]. 2021;1099(1):12077. Dostupné z: <https://doi.org/10.1088/1757-899x/1099/1/012077>

Termín zadání: 6.2.2023

Termín odevzdání: 29.5.2023

Vedoucí práce: Ing. Roman Jakubiček, Ph.D.

doc. Ing. Jana Kolářová, Ph.D.
předseda rady studijního programu

UPOZORNĚNÍ:

Autor bakalářské práce nesmí při vytváření bakalářské 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

Práca sa zaoberá rešeršou a implementáciou metód, ktoré eliminujú vplyv nevyrovnaného datasetu na učenie neurónových sietí. Jednotlivé metódy sú porovnané medzi sebou pre rôzne úrovne nevyrovnanosti. Experimenty realizované v práci sú taktiež porovnané s dostupnou literatúrou a kontrolným experimentom, ktorý bol realizovaný bez metódy eliminácie vplyvu nevyrovnaného datasetu na učenie neurónových sietí. Experimenty sú následne rozšírené na ďalší dataset, kde sú porovnané pre originálnu nevyrovnanosť nachádzajúcu sa v datasete. V teoretickej časti je priblížená téma neurónových sietí a problémy, ktoré sa môžu vyskytnúť pri učení. Následne sú priblížené konvolučné siete a ich optimalizačné algoritmy. Práca ďalej obsahuje bližšie spracovanú problematiku nevyrovnaného datasetu, vrátane používaných metrík a prístupov, ktoré umožňujú elimináciu tohto problému.

KĽÚČOVÉ SLOVÁ

Neurónové siete, Nevyrovnaný dataset, Vážená vzájomná entropia, Dátová augmentácia, SMOTE, Náhodné prevzorkovanie.

ABSTRACT

This thesis deals with the research and implementation of methods that eliminate the influence of an imbalanced dataset on the learning of neural networks. Individual methods are compared with each other for different levels of imbalance. The experiments carried out in the work are also compared with the available literature and a control experiment, which was carried out without the method of eliminating the influence of an imbalanced dataset. The experiments are extended to another dataset containing the original imbalance and compared. In the theoretical section, the topic of neural networks and the problems that may occur during learning are brought up. Subsequently, convolutional networks and their optimization algorithms are presented. The thesis also contains a more detailed presentation of the issue of an imbalanced dataset, including the metrics used in experiments and approaches used to eliminate this problem.

KEYWORDS

Neural networks, Imbalanced dataset, Weighted cross entropy loss, Data augmentation, SMOTE, Random weighted oversampling.

VICIANOVÁ, Veronika. *Prístupy učenia neuronových sietí s nevyrovnaným datasetom*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav biomedicínského inženýrství, 2023, 61 s. Bakalárska práca. Vedúci práce: Ing. Roman Jakubíček, Ph.D.

Vyhlásenie autora o pôvodnosti diela

Meno a priezvisko autora: Veronika Vicianová
VUT ID autora: 231170
Typ práce: Bakalárska práca
Akademický rok: 2022/23
Téma záverečnej práce: Prístupy učenia neuronových sietí s nevyrovnaným datasetom

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

Rada by som poďakovala vedúcemu bakalárskej práce pánovi Ing. Romanovi Jakubíčkovi, Ph.D. za odborné vedenie, konzultácie, trpezlivosť a podnetné návrhy k práci.

Obsah

Úvod	10
1 Teoretický úvod	11
1.1 Teória neurónových sietí	11
1.2 Matematická definícia neurónu	11
1.3 Aktivačné funkcie	11
1.4 Viacvrstvé neurónové siete	13
1.5 Spôsobu učenia neurónovej siete	14
2 Konvolučné neurónové siete	19
2.1 Štruktúra konvolyčných neurónových sietí	19
2.2 Metódy regularizácie	21
3 Problematika nevyrovnaného datasetu	24
3.1 Formulácia problému	24
3.2 Metódy hodnotenia predikcie modelu	24
3.3 Riešenia problému nevyrovnaného datasetu	26
4 Použité datasety	30
4.1 Microsoft ASIRRA	30
4.2 Dataset skoliózy, spondylolistézy a zdravej chrbtice	31
5 Návrh experimentov	33
5.1 Predspracovanie dát	33
5.2 Použitá neurónová sieť	34
5.3 Implementácia jednotlivých experimentov	36
5.4 Štatistické vyhodnotenie výsledkov	40
6 Výsledky a diskusia	41
6.1 Učenie s datasetom Microsoft ASIRRA	41
6.2 Učenie s datasetom skoliózy a zdravej chrbtice	50
6.3 Diskusia	53
Záver	55
Zoznam symbolov a skratiek	61

Zoznam obrázkov

1.1	Neurónové siete	13
1.2	Preučenie	18
2.1	Vrstvy konvolučnej neurónové siete	20
2.2	Funkcia MaxPooling	20
2.3	Architektúra enkodér-dekódér	22
3.1	Matica zámien	25
4.1	Ukážka datasetu Microsoft ASIRRA	31
4.2	Ukážka datasetu skoliózy, spondylolistézy a zdravej chrbtice	32
5.1	Vývojový diagram architektúry neurónovej siete	35
5.2	Vývojový diagram váhovanej vzájomnej entropie	37
5.3	Vývojový diagram dátovej augmentácie	37
5.4	Vývojový diagram SMOTE	38
5.5	Ukážka datasetu po realizácii algoritmu SMOTE, dataset č.1	39
5.6	Vývojový diagram náhodného prevzorkovania	39
6.1	Výsledky vzájomnej entropie	42
6.2	Výsledky trénovania s váženou vzájomnou entropiou	44
6.3	Výsledky dátovej augmentácie	45
6.4	Výsledky metódy SMOTE	47
6.5	Výsledky náhodného prevzorkovania	48
6.6	Porovnanie jednotlivých metód medzi sebou	50
6.7	Grafické výsledky experimentov s datasetom skoliózy a zdravej chrbtice	51
6.8	Grafické výsledky experimentov s datasetom skoliózy a zdravej chrbtice	53

Zoznam tabuliek

5.1	Rozdelenie vzoriek v absolútnom počte pre dataset Microsoft ASIRRA	34
5.2	Rozdelenie vzoriek v absolútnom počte pre dataset skoliózy a zdravej chrbtice	34
6.1	Výsledky testovania modelu natrénovaného bez pridania metódy eliminácie.	42
6.2	Výsledky tréovania s váženou vzájomnou entropiou	44
6.3	Výsledky tréovania s dátovou augmentáciou	46
6.4	Výsledky tréovania s metódou SMOTE	47
6.5	Výsledky tréovania s náhodným prevzorkovaním	49
6.6	Výsledky otestovaného modelu pre dataset skoliózy a zdravej chrbtice	52

Úvod

Neurónové siete sú dnes veľmi populárne najmä pri spracovaní veľkého množstva dát, ktoré je obtiažne, prípadne nemožné spracovať bez použitia výpočtovej techniky. Dáta, ktoré sú poskytnuté pre učenie neurónových sietí by mali spĺňať určité požiadavky. Pokiaľ dataset obsahuje viac ako jednu skupinu, ktorú sa neurónová sieť učí, tieto skupiny by mali byť v datasete zastúpené v približne rovnakom počte. Ak táto podmienka nie je splnená, dataset je nevyrovnaný, čo negatívne vplyva na učenie siete.

V praxi sa často vyskytuje situácia, kedy nie je možné získať potrebné množstvo dát menšinovej skupiny. Ako príklad je možné uviesť situáciu v zdravotníctve, kedy je potrebné klasifikovať pacientov na základe príznakov vzácnej choroby alebo novo vyskytnutej. Je samozrejmé, že v tomto prípade bude k dispozícii väčšie množstvo dát nazbieraných od zdravých pacientov a hrozí nesprávna diagnóza chorého pacienta. Naopak správna klasifikácia príznakov chorého pacienta môže viesť k včasnej diagnóze a liečbe alebo nasadeniu potrebných opatrení.

Problém nevyrovnaného datasetu sa vyskytuje aj v iných vedeckých oblastiach ako napríklad v meteorológii, kedy je nutné správne klasifikovať príznaky vzácnych a niekedy aj život ohrozujúcich meteorologických javov ako napríklad tornád, hurikánov, atď.

Použitie nevyrovnaného datasetu pre učenie neurónovej siete má za následok skreslené výsledky. Keďže neurónová sieť nemá k dispozícii dostatočný počet dát menšinovej skupiny, výstupom bude, že sa nenaučí klasifikovať menšinovú skupinu, teda bude ju klasifikovať náhodne alebo označí všetky dáta ako patriace väčšinovej skupine a menšinovú skupinu sa nenaučí vôbec. Avšak, cieľom je klasifikovať obe skupiny s rovnakou presnosťou, pričom menšinová skupina je často bodom záujmu.

Trénovanie neurónovej siete s elimináciou vplyvu nevyrovnaného datasetu by mohlo výrazne prispieť k úspešnosti klasifikácie aj pre deje, ktoré sú v zozbieraných dátach málo zastúpené.

Táto práca obsahuje teoretický rozbor týkajúci neurónových sietí, problémov učenia a predovšetkým sa zameriava na problém nevyrovnaného datasetu. Na základe spracovanej literatúry sú v bakalárskej práci implementované 4 prístupy s cieľom eliminovať vplyv nevyrovnaného datasetu, pričom úspešnosť je otestovaná na dvoch datasetoch. Výsledky týchto prístupov sú následne vyhodnotené, porovnané medzi sebou a s experimentom bez implementácie metódy eliminujúcej vplyv nevyrovnaného datasetu.

1 Teoretický úvod

1.1 Teória neurónových sietí

Neurónové siete sú jednou z oblastí strojového učenia. Prvé vytvorené neurónové siete slúžili najmä na spracovanie štatistických dát, zvukových nahrávok, či obrazovú analýzu. Dnes je ich využitie oveľa širšie a nájdeme ich vo všetkom od aplikácií, s ktorými pracuje bežný používateľ, až po ich využitie na vedeckých pracoviskách. Úlohy, pre ktoré bola neurónová sieť vytvorená, sa môžu líšiť. Vo všeobecnosti je ich úlohou rozoznávať rôzne vzory („patterns“), ktoré sú obsiahnuté vo vstupných údajoch a správne ich klasifikovať. [1]

1.2 Matematická definícia neurónu

Najzaužívanejším modelom umelého neurónu je McCulloch-Pittsov neurón, taktiež nazývaný aj formálny neurón alebo perceptrón. Matematický model tohto neurónu je možné opísať ako systém, na ktorého vstup je privádzaný signál v podobe n -rozmerného vektoru $\bar{x} = (x_1, x_2, \dots, x_n)$. Vstupný signál je ďalej vážený vektorom synaptických váh $\bar{w} = (w_1, w_1, \dots, w_n)$. Od vážená alebo presnejšie, od skalárneho súčinu vektorov \bar{x} a \bar{w} je odčítaná θ , ktorá vyjadruje prah excitácie neurónu („bias“, „threshold“). Neurón je aktívny, pokiaľ je vážená suma vstupov väčšia než prah. Vážená suma vektorov je vynásobená aktivačnou funkciou f . Úlohou aktivačnej funkcie je pretransformovať váženú sumu vektorov na požadované výstupné hodnoty. Výstupom neurónu je skalárny výstupný signál y . Opísaný vzťah je možné vyjadriť vzorcom 1.1: [2]

$$y = f(\bar{w} \cdot \bar{x}) = f\left(\sum_{j=1}^{n+1} w_j x_j\right) = f\left(\sum_{j=1}^n w_j x_j - \theta\right) \quad (1.1)$$

1.3 Aktivačné funkcie

Pre jednoduchšie spracovanie výstupu neurónu, je potrebné voliť aktivačnú funkciu takú, ktorá je diferencovateľná a hodnoty váženej sumy vektorov pretransformuje na reálne číslo medzi 0 a 1. Vo výnimočných prípadoch je možné voliť aj nediferencovateľnú funkciu. Tieto funkcie sú taktiež väčšinou nelineárne, rastúce, spojité. Konkrétna voľba aktivačnej funkcie závisí od problému, s ktorým sa neurónová sieť vysporadúva. Ich voľbou je ovplyvnený čas, za ktorý je natrénovaná neurónová sieť a je ovplyvnená taktiež aj výpočtová náročnosť siete. [1, 2]

1.3.1 Lineárna funkcia

Jednou z najjednoduchších aktivačných funkcií je lineárna funkcia. Používa sa napríklad pri filtrácii signálu. Je vyjadrená vzorcom 1.2:

$$f(x) = \lambda(x) \quad (1.2)$$

Kde λ je konštanta, určuje strmú funkciu a x sú vstupné dáta neurónu.

1.3.2 Skoková funkcia

Ďalším dobrým príkladom je skoková funkcia. Táto funkcia je nediferencovateľná a preto jej použitie nie je možné aplikovať v algoritmoch učenia založených na gradientných metódach, kde je využívané spätné šírenie chyby („backpropagation“). Je vhodná pre binárnu klasifikáciu datasetu, pokiaľ je potrebné získať presne zadanú hranicu medzi triedami. Je vyjadrená rovnicou 1.3: [2]

$$f(x) = \begin{cases} -1, & \text{ak } x < 0 \\ 1, & \text{ak } x \geq 0 \end{cases} \quad (1.3)$$

1.3.3 Sigmoida

Veľmi populárna je sigmoida, definovaná vzorcom 1.4. Používa sa na modelovanie nelineárnych závislostí. Ako konkrétny príklad je taktiež možné uviesť klasifikáciu binárneho datasetu. Výhodou tejto aktivačnej funkcie je, že je jednoducho diferencovateľná [3].

$$f(x) = \frac{1}{1 + e^{-\lambda(x)}} \quad (1.4)$$

1.3.4 ReLU

„Rectified linear unit“ (ReLU), v preklade vylepšená lineárna jednotka. V súčasnosti je najpoužívanejšia aktivačná funkcia, ktorá v mnohých ohľadoch prekonala sigmoidu. Dosahovala lepšie výsledky pri optimalizovaných gradientných metódach a taktisto predchádzala problému miznúceho gradientu podrobnejšie rozobratého v kapitole 1.5.6. Je vyjadrená rovnicou 1.5: [3]

$$f(x) = \max\{0, x\} \quad (1.5)$$

1.3.5 „Softmax“

„Softmax“ je skratka pre jemnejší („softer“) argument maxima. Táto funkcia vracia vektor reálnych hodnôt, ktorých suma sa rovná 1. Dôležitou vlastnosťou teda je, že

transformuje výstupné dáta neurónu x_i na hodnoty 0 až 1 a môžu byť interpretované ako pravdepodobnosti. Výpočet tejto aktivačnej funkcie je uskutočňovaný pomocou vzťahu 1.6. [4]

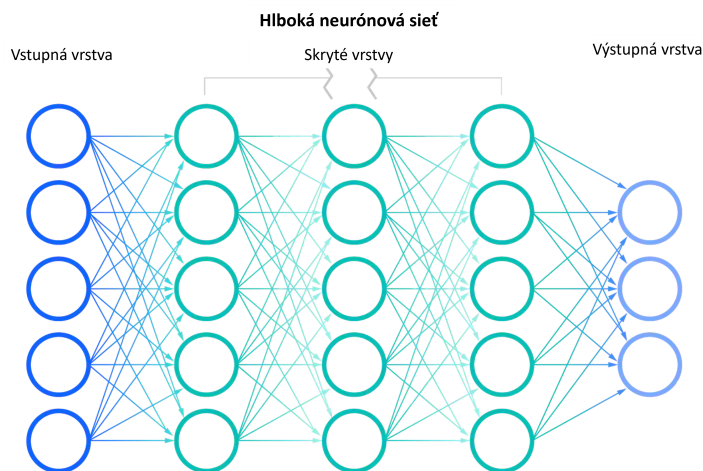
$$f(x_i) = \frac{e^{x_i}}{\sum_{j=1}^n e^{x_j}} \quad (1.6)$$

kde x_i je i -tý prvok vektoru x_n , pričom n je dĺžka vektoru.

1.4 Viacvrstvéové neurónové siete

Jeden neurón nie je schopný riešiť nelineárne problémy, čo viedlo k vzniku neurónových sietí. Viacvrstvéové dopredné neurónové siete sú zložené z niekoľkých vrstiev perceptrónov. Vrstvy sa zvyčajne rozdeľujú na vrstvu vstupných neurónov („input neurons“), skrytých neurónov („hidden neurons“) a výstupných neurónov („output neurons“). Pre predstavu sú zobrazené na obrázku 1.1. Počet vrstiev skrytých neurónov určuje hĺbku siete - odtiaľ pochádza aj termín „deep learning“, v preklade hlboké učenie. [1, 5, 3]

Dopredná sieť znamená, že výstup z každého neurónu sa dostane len na neuróny nasledujúcej vrstvy, pričom susedné neuróny v rámci jednej vrstvy nie sú prepojené. Medzi vrstvami neexistuje spätná väzba. Avšak dopredný smer nezaručuje učenie neurónovej siete. Aby bol výstup čo najbližšie požadovanej hodnote, je potrebná spätná väzba, ktorá tento výstup upraví. Pre optimalizáciu výstupu existujú rôzne algoritmy. Jedným z nich je algoritmus spätného šírenia chyby. [3, 2]



Obr. 1.1: Štruktúra neurónových sietí [6].

1.5 Spôsohy učenia neurónovej siete

Učenie neurónovej siete je proces, kedy sa hľadajú vhodné váhy pre vstupný vektor také, aby bola dosiahnutá čo najvyššia presnosť výstupu. Medzi základné spôsoby učenia patrí učenie bez učiteľa („unsupervised learning“) a učenie s učiteľom („supervised learning“). Táto práca sa zaoberá procesom učenia s učiteľom, čo znamená, že pri tréňovaní sú dáta dopredu označené správnym výstupom. Neskôr pri testovaní by teda neurónová sieť mala byť schopná sama klasifikovať dáta, ktoré neboli vopred označené. [1]

1.5.1 Úprava váh

Učenie s učiteľom je uskutočňované zmenou váh na základe spätnej väzby učiteľa. J -tá váha neurónu (w_j) sa mení podľa rovnice 1.7:

$$w_j(t+1) = w_j(t) + \Delta w_j(t) = w_j(t) + \alpha s(t)x_i(t) \quad (1.7)$$

kde α je rýchlosť učenia alebo aj krok učenia („learning rate“), w sú váhy, s odpovedá aritmetickému rozdielu medzi požadovanou hodnotou a reálnou, x_i je i -tý výstup neurónu a t slúži pre iteráciu. [1]

Učenie prebieha v epochách, pričom počas jednej epochy sú predstavené všetky dvojice vstupných vektorov a ich očakávaných hodnôt neurónovej siete. Po každej epoche by mal byť výstup neurónu bližšie k požadovanej hodnote, keďže počet epoch je možné označiť aj ako počet iterácií počas ktorých dochádza k uloženiu modifikovaných váh. Posúvanie k požadovanej hodnote je napríklad pri binárnom perceptróne zabezpečené δ (delta) pravidlom, ktoré upravuje váhy. Pričom δ je rovná aritmetickému rozdielu medzi požadovanou hodnotou výstupu d a reálnym výstupom neurónu y . Pre j -tú váhu platí: [7, 1]

$$\Delta w_j = \alpha(d - y)x_i \quad (1.8)$$

1.5.2 Chybová funkcia, výpočet odchýlky neurónu

Stredná kvadratická odchýlka

Odchýlka výstupu neurónu od očakávaného výstupu sa počíta pomocou chybovej funkcie. Konkrétny výber chybovej funkcie je opäť prispôbený problému, ktorý neurónová sieť rieši. Základným príkladom chybovej funkcie je kvadratická odchýlka vyjadrená vzorcom 1.9:

$$O = \frac{1}{2} \sum_{j=1}^n (d_j - o_j)^2 \quad (1.9)$$

kde j je počet výstupných neurónov, d_j je očakávaná hodnota z neurónovej siete, o_j je výstupná hodnota siete a O je výsledná kvadratická odchýlka. V tomto prípade je potrebné ešte vypočítať kumulovanú chybu, ktorá sa kumuluje počas jednej epochy. [1]

Po opakovaní jednotlivých epoch je možné vykresliť trend chybovej funkcie, ktorý je ukazovateľom nastavenia správnych parametrov a zároveň týmto spôsobom je umožnené sledovať ako prebieha tréning. Pokiaľ by krivka oscilovala, môže sa jednať o príliš malý tréningový súbor alebo príliš veľký krok učenia. Naopak, pokiaľ by bola plochá, môže to ukazovať príliš veľký testovací súbor alebo príliš malý krok učenia. Ideálne by teda krivka mala byť hladká funkcia a kvadratická odchýlka by mala postupne dosiahnuť svoje minimum. [2]

Vzájomná entropia

Vzájomná entropia, v angličtine známa ako „cross entropy loss“, je chybová funkcia, ktorá slúži na výpočet chyby za pomoci pravdepodobnostnej distribúcie. Vzájomná entropia exponenciálne rastie, pokiaľ výstup neurónu diverguje od požadovaného výstupu. To znamená, že váhy sú upravované v spätnom šírení chyby s oveľa väčšími krokmi, pokiaľ sa model mýli. Naopak, pokiaľ výstup konverguje k správnejmu riešeniu, entropia a teda aj chyba logaritmicke klesá a váhy sú upravované menším krokom. Pre binárny dataset je vzájomná entropia (VE) definovaná vzorcom 1.10: [8]

$$VE = - \sum_{c=1}^M (y_{o,c} \log(p) + (1 - y_{o,c}) \log(1 - p)) \quad (1.10)$$

kde y je binárny indikátor (nula alebo jedna) rovný jednej, ak c je správna klasifikácia pre daný objekt o , a p je výstup neurónu ako hodnota pravdepodobnosti, ktorá mohla vzniknúť napríklad aktivačnou funkciou „Softmax“. Pre viac tried je vyjadrená vzorcom 1.11:

$$VE = - \sum_{c=1}^M y_{o,c} \log(p_{o,c}) \quad (1.11)$$

kde M je počet tried. [8]

1.5.3 Gradientné metódy

Metóda gradientného zostupu

Hľadanie minima chybovej funkcie je zabezpečené metódou gradientného zostupu. Gradient totiž udáva smer najprudšieho nárastu funkcie. Vzťah na výpočet gradientu je uvedený ako:

$$\text{grad}(E) = \frac{\partial E}{\partial w} \quad (1.12)$$

kde ∂E je parciálna derivácia chybovej funkcie E podľa váh w . [7]

Adam

Problém s metódou gradientného zostupu je, že je možné ľahko preskočiť minimum funkcie, prípadne k nemu nikdy nedoraziť. V súčasnosti sa preto používa vylepšená metóda nazývaná ADAM, čo je skratkou pre „adaptive moment estimation“, v preklade odhad adaptívneho momentu. Princíp tejto metódy spočíva v adaptívnom upravovaní veľkosti gradientu a teda aj váh na základe učenia siete. Dôležitou vlastnosťou je, že Adam pri výpočte gradientu berie do úvahy aj predchádzajúce výpočty gradientov. Moment zabezpečuje väčšiu zmenu váh v prípade, že sieť sa učí správne, čím urýchľuje učenie a dosiahnutie minima chybovej funkcie. [9]

1.5.4 Trénovanie spätným šírením chýb

Algoritmus spätného šírenia chyby zahŕňa kombináciu vyššie uvedených metód tréningovania neurónovej siete. Jednotlivé kroky algoritmu je možné prehľadne zhrnúť v nasledujúcich vetách. Najskôr sú nastavené podmienky pre ukončenie učenia, väčšinou maximálny počet epoch alebo minimálna kvadratická odchýlka O . Ďalej sú inicializované váhy všetkých neurónov, ich hodnoty môžu byť nastavené aj náhodne. Na vstup neurónovej siete je privedený nový tréningový vektor $\bar{x} = (x_1, x_2, \dots, x_n)$ a na výstup je privedená očakávaná hodnota d . Nasleduje učenie, ktoré v tomto prípade pozostáva z troch krokov. Výpočet aktuálnych výstupov y (dopredné šírenie). Výpočet odchýlky od požadovaných hodnôt d (spätné šírenie). Modifikácia váh a prahu podľa zistených odchýliek pre dosiahnutie čo najmenšej chyby vybranou gradientnou metódou (dopredné šírenie). Postup od privedenia nového tréningového vektoru až po modifikáciu váh je opakované pre všetky prvky z tréningovej množiny až do konca epochy. Algoritmus sa opakuje až do splnenia jednej z ukončovacích podmienok. [7]

1.5.5 Testovanie naučeného modelu

Aby bolo možné naučený model testovať, množinu dát, ktoré sú k dispozícii je potrebné rozdeliť na tri podmnožiny. Prvá je tréningová, ktorá po privedení na vstup neurónovej siete slúži na výpočet chyby a modifikáciu váh na základe tohto výpočtu. Ďalšia je validačná, ktorá slúži na vyhodnocovanie presnosti klasifikácie modelu počas tréningovania, po privedení validačnej množiny na vstup nie je zaistená spätná väzba

a model sa ju teda neučí. Porovnávaním týchto dvoch podmnožín je možné rozpoznať mnoho chýb, ktoré môžu vzniknúť pri učení. Dáta bývajú zvyčajne rozdelené v pomere 80 % pre tréningovú množinu a 20 % pre testovaciu. Treťou podmnožinou je testovacia. Do neurónovej siete vstupuje len raz, keď je už model natrénovaný a uložený, neurónová sieť ju nikdy predtým nevidela. Vďaka tejto množine je vyhodnocované, ako neurónová sieť pracuje s novými dátami. Vzhľadom na to, že model sa validačnú množinu neučil, je možné použiť validačnú množinu aj ako testovaciu. [1]

1.5.6 Problémy pri učení

Preučenie

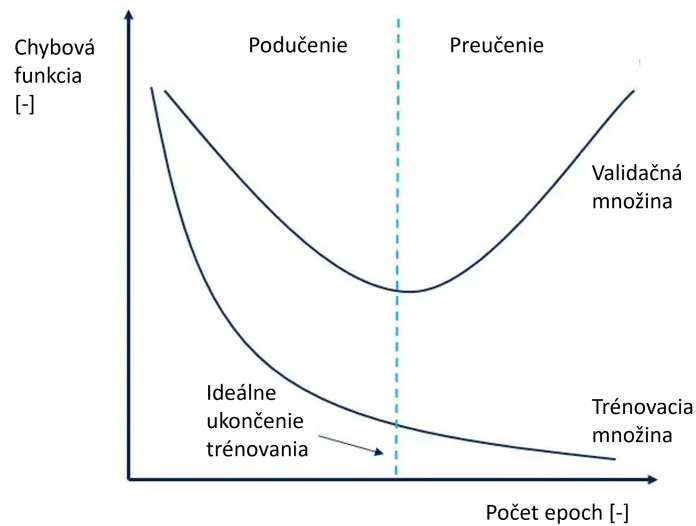
Pri sledovaní trendu chybovej funkcie je nutné dbať na to, aby nedochádzalo k preučeniu („overfitting“). Tento jav sa vyznačuje tým, že v priebehu tréningovania kumulovaná chyba tréningovej množiny (E_{train}) stále klesá, zatiaľ čo kumulovaná chyba validačnej množiny (E_{val}) začne pomaly stúpať. Ilustračné zobrazenie tohto javu sa nachádza na obrázku 1.2. Preučenie znamená, že neurónová sieť sa príliš dobre naučila tréningové dáta ale nedokáže už rozoznávať nové dáta. Môže to byť spôsobené napríklad tým, že model začne upravovať svoje váhy aj na základe šumu alebo redundantných informácií obsiahnutých v tréningovej množine, čím sa stráca generalizačná schopnosť modelu. Problému preučenia je možné zabrániť správnym nastavením počtu tréningových epoch, prípadne ukončením tréningovania vo chvíli, keď rozdiel kumulovanej chyby medzi E_{train} a E_{val} dosiahne prijateľnú vopred nastavenú hodnotu. [1]

Miznúci gradient

Miznúci gradient („vanishing gradient problem“) je problém najmä v hlbokých neurónových sieťach, ktoré sú založené na gradientných metódach a spätnom šírení chyby. Problém nastáva vtedy, keď gradient vypočítaný podľa rovnice 1.12 dosahuje príliš malých čísel. Ak tieto čísla nadobudnú hodnoty až tak malé, že prestanú ovplyvňovať úpravu váh, sieť sa prestane tréningovať. Jedným z riešení tohto problému je používanie aktivačnej funkcie ReLU, ktorej vlastnosti zabraňujú výskytu tohto problému alebo ďalším riešením je implementácia batch normalizácie, ktorá je bližšie vysvetlená v kapitole 2.2.2. [10]

Nevyrovnaný dataset

Aby bola neurónová sieť správne natrénovaná, je potrebné mať adekvátny dataset. Medzi vyžadované vlastnosti patrí, aby bol predovšetkým reprezentatívny a mal by



Obr. 1.2: Zobrazenie trendu chybovej funkcie s vyznačením preučenia.

obsahovať rovnomerné zastúpenie všetkých skupín. Napríklad v binárnej klasifikácii je ideálne, aby dataset obsahoval 50 % dát jednej triedy a 50 % dát druhej triedy. V opačnom prípade pri tréovaní vzniká riziko, že sieť natrénuje klasifikovanie jednej skupiny lepšie ako druhej. Práca sa zaoberá riešeniami tohto problému a sú zhrnuté v samostatnej kapitole 3.

2 Konvolučné neurónové siete

Konvolučné neurónové siete sú špecializovaný druh neurónových sietí používané na dáta s mriežkovou topológiu. Napríklad na obrazové dáta, ktoré obsahujú 2-D mriežku pixelov. Najčastejšie sa používajú na detekciu vzorov („patterns“), ktorými môžu byť v obraze napríklad hrany, textúry, tvary, atď. Ako už naznačuje ich názov, tieto neurónové siete využívajú špeciálny druh lineárnej operácie, teda konvolúciu. Konvolúcia je v tomto prípade využívaná namiesto všeobecného násobenia matíc, ktoré sa využíva v jednoduchých neurónových sieťach. Jej vzorec je vyjadrený ako: [3]

$$x(t) = x_1(t) * x_2(t) = \int_{-\infty}^{\infty} x_1(\tau) \cdot x_2(t - \tau) d\tau \quad (2.1)$$

kde $x_2(\tau)$ sa nazýva konvolučné jadro. Výsledkom konvolúcie je funkcia $x(t)$, niekedy označovaná aj ako mapa vlastností alebo príznaková mapa („feature map“), ktorá vzniká ako $x_1(t)$ modifikovaná vlastnosťami konvolučného jadra $x_2(t)$. [11]

2.1 Štruktúra konvolučných neurónových sietí

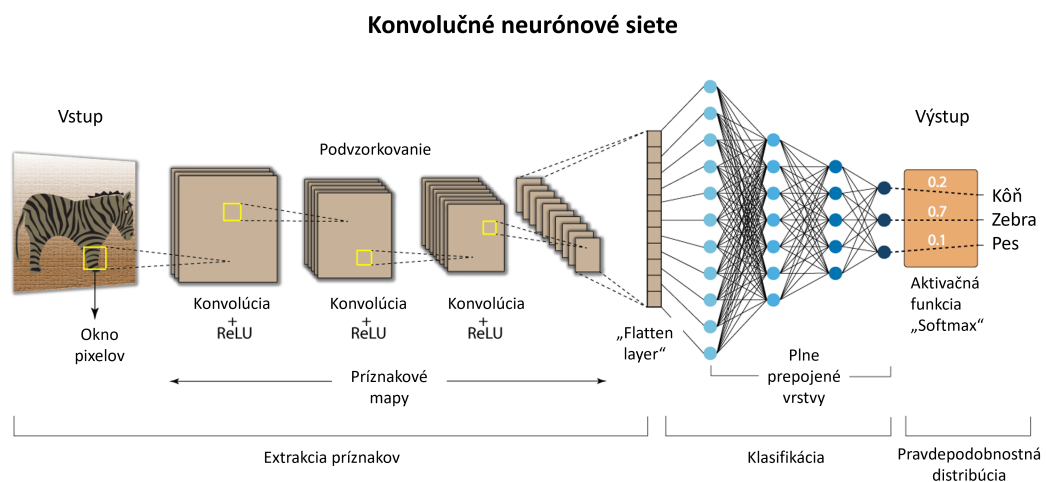
2.1.1 Základná štruktúra konvolučnej siete

Konvolučná časť neurónovej siete je formovaná niekoľkými typmi neurónových vrstiev. Prvá je konvolučná vrstva, ktorá slúži ako filter, ktorého veľkosť je buď rovnako veľká ako rozmery obrázku (počet pixelov) alebo menšia. Druhá je nelineárna alebo aj korekčná vrstva, ktorá je väčšinou spojená s konvolučnou. Vzniká teda nelineárny dvojdimenzionálny filter, čo je zapríčinené nelineárnym výstupom neurónu, konkrétne jeho aktivačnou funkciou. Pôvodne bola ako aktivačná funkcia používaná sigmoida avšak ReLU, dosahovala lepšie výsledky. Tretím typom vrstvy je podvzorkovacia („pooling“) vrstva, ktorá podvzorkovaním zmenší dimenzionalitu príznakových máp. Dochádza teda k priestorovej kompresii dát. Väčšinou je na podvzorkovanie používaná funkcia, ktorá hľadá maximum z vybraného okna pixelov, táto funkcia sa nazýva „MaxPooling“ a jej príklad je ilustrovaný na obrázku 2.2. Maximá z vybraných okien sa spoja do novej množiny a vzniká tak nová menšia príznaková mapa. Po aplikovaní podvzorkovacej vrstvy na všetky pôvodné príznakové mapy, zmenšené príznakové mapy vstupujú znovu do konvolučnej vrstvy. Tento proces sa opakuje dovtedy, kým nie sú dáta zmenšené na potrebnú veľkosť. [12, 3]

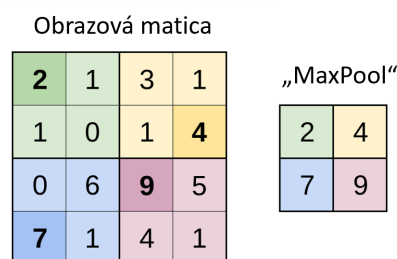
Poslednými vrstvami môžu byť plne prepojené vrstvy a výstupná vrstva. Plne prepojená vrstva je usporiadaním neurónov podobná klasickým neurónovým sieťam. Jednotlivé neuróny sú prepojené s neurónmi z predchádzajúcej aj nasledujúcej

vrstvy, ale nie sú prepojené medzi sebou v rámci jednej vrstvy. Prvá z plne prepojených vrstiev spojí všetky príznakové mapy do jedného vektora. Táto vrstva sa nazýva aj „flatten layer“, pretože pretransformuje všetky priestorové dimenzie do jednej dimenzie. Na ďalších plne prepojených vrstvách nasleduje extrakcia globálnych príznakov a klasifikácia na základe týchto príznakov. Pravdepodobnosti, s akými boli vstupné dáta roztriedené do jednotlivých tried sú určené napríklad funkciou „Softmax“. [3]

Na týchto princípoch uvedených vyššie, s upraveným usporiadaním jednotlivých vrstiev, fungujú známe architektúry neurónových sietí ako napríklad AlexNet a VGG16.



Obr. 2.1: Jednotlivé vrstvy konvolučnej neurónovej siete [13].



Obr. 2.2: Funkcia slúžiaca na podvzorkovanie - MaxPooling [14].

ResNet-50

Jedna z najpoužívanejších architektúr je ResNet-50 („Residual Network“). Číslo 50 značí počet vrstiev, pričom je možné nájsť aj ResNet-18 alebo ResNet-38. Táto

architektúra neurónových sietí sa vyznačuje tým, že je hlbšia, teda obsahuje viac vrstiev, než predchádzajúce neurónové siete. Autori tejto architektúry dokázali predísť problémom, ktoré vznikali pri navýšení počtu skrytých vrstiev, ako napríklad problém miznúceho gradientu. To, v čom sa ResNet-50 líši od klasických architektúr, sú špeciálne prepojeniami medzi vrstvami („skip connections“), ktoré vznikajú implementáciou reziduálnej funkcie. Výsledkom je, že neuróny sa neučia len z výstupu predchádzajúcej vrstvy, ale aj z vektoru vstupných hodnôt, ktoré vstupovali do siete o niekoľko vrstiev predtým. [15]

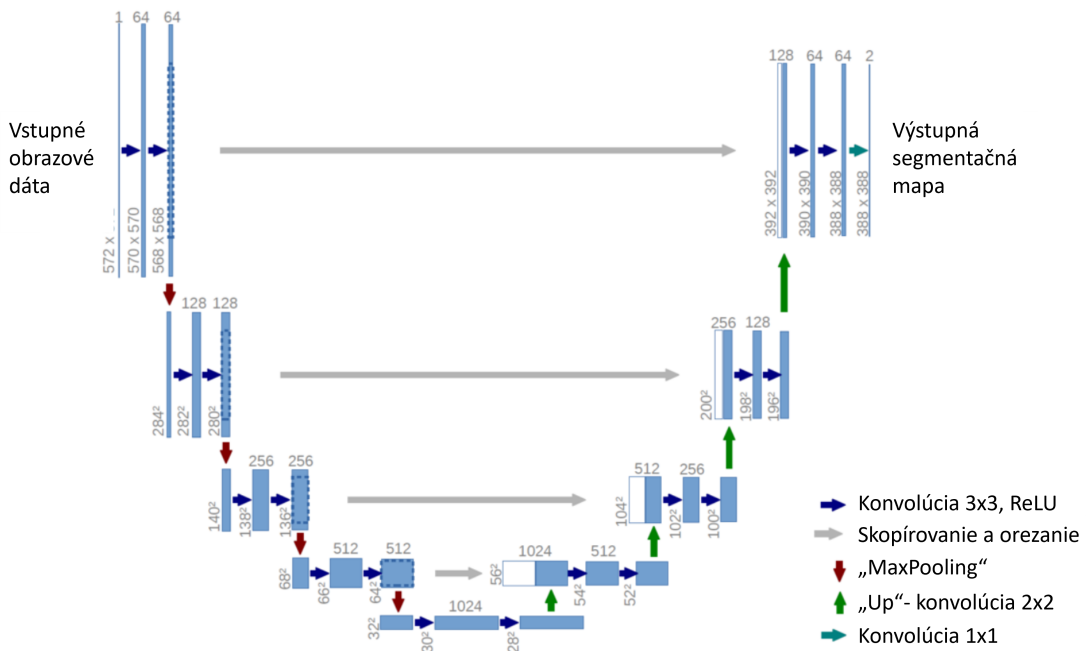
2.1.2 Sieť typu enkodér-dekodér

Pre správnu redukciu a selekciu príznakov vznikol typ neurónovej siete nazývaný enkodér-dekodér. Okrem uvedenej vlastnosti sú znížené aj výpočtová a pamäťová náročnosť siete. Princíp spočíva v zmenšovaní dátových blokov a ich podvzorkovaní. Následne sú dáta skrz dekodér opätovne rozšírené prevzorkovaním až do pôvodnej veľkosti. Výstupom je matica s vektormi príznakov, ktoré popisujú pôvodný vstup s menšou transversálnou dimenziou, než vstupný obraz. Tento koncept je používaný v U- a V- typoch sietí, pričom názvy sú zvolené podľa písmena, ktoré náčrt ich architektúry pripomína. Príklad je možné vidieť na obrázku 2.3. Známu konkrétnou architektúrou je napríklad U-net. Architektúra enkodér-dekodér je používaná najčastejšie pri neurónových sieťach určených na segmentáciu. Pri neurónových sieťach určených na klasifikáciu je postačujúca architektúra enkodéru. [12, 3]

2.2 Metódy regularizácie

2.2.1 Vrstva „Drop out“

„Drop out“ je regularizačná vrstva, ktorá sa používa najmä pri neurónových sieťach, ktoré majú mnoho skrytých vrstiev. Viac skrytých vrstiev neurónovým sieťam dovoľuje naučiť sa zložité vzťahy obsiahnuté v tréningových dátach. Táto vlastnosť môže byť výhodou pri klasifikácii zložitých vzoriek, avšak medzi zložité vzťahy je možné zaradiť aj napríklad pozadie obrázkov alebo šum, ktoré nemusia byť obsiahnuté vo validačnej množine. Môže tak dochádzať k preučeniu siete, ktoré bolo opísané v kapitole 1.5.6. „Drop out“ slúži na náhodné, dočasné vypnutie niektorých neurónov spolu s ich spojeniami. Zabraňuje tým preučeniu a zároveň pomáha znížiť aj výpočtovú náročnosť. [17]



Obr. 2.3: Princíp enkodéru a dekodéru, ilustrovaný s jednotlivými vrstvami. Obrázok je preložený z originálnej štúdie architektúry U-net [16].

2.2.2 Vrstva „Batch normalization“

„Batch normalization“ je metóda, ktorá sa taktiež používa na veľmi hlboké neurónové siete. Regularizácia pomocou batch normalizácie spočíva v normalizácii dát, ktoré prichádzajú na vstup vrstvy. Dáta tak prichádzajú na vstup stále v rovnakom rozmedzí, čím je zabránené vstupu príliš malých alebo príliš veľkých hodnôt. Týmto spôsobom je urýchlený tréning, pretože vďaka normalizovaným dátam je možné vyhnúť sa miznúcemu gradientu a váhy sú teda upravované rýchlejšie a efektívnejšie. [18]

2.2.3 L1 a L2 regularizácia

Metóda opäť slúži problému preučenia. Úlohou L1 a L2 regularizácie je upravovať výpočet chybovej funkcie, ktorá spätným šírením upravuje váhy, pretože problém často spočíva v príliš vysokých, alebo príliš presných váhach, ktoré nadmerne kopírujú vlastnosti tréningového datasetu.

V prípade L1 regularizácie je k chybovej funkcii pripočítavaný súčet absolútnej hodnoty váh, čím sú niektoré váhy spätným šírením chyby znížené až na nulu. Táto úprava má za výsledok redšiu neurónovú sieť a v ideálnom prípade by týmto spôsobom mali byť odstránené váhy pre prebytočné vlastnosti. Pokiaľ by sme ako príklad

chybovej funkcie zobrali kvadratickú odchýlku O , tak je možné vyjadriť úpravu chybovej funkcie vzťahom 2.2.

$$O = \frac{1}{2} \sum_{j=1}^n (d_j - o_j)^2 + \lambda \sum_{j=1}^n |w_j| \quad (2.2)$$

kde w_j reprezentuje jednotlivé váhy a λ je parameter, ktorý určuje ako veľmi budú váhy znížené. Ak je nastavený na vysokú hodnotu, tak váhy zostanú nízke, model bude podučený a teda aj neschopný správnej klasifikácie. Ak naopak, bude nastavený na príliš nízku hodnotu, tak váhy môžu zostať vysoké pre prebytočné vlastnosti, a teda môže dôjsť k preučeniu.

Algoritmus L2 regularizácie spočíva v pričítaní súčtu hodnôt všetkých váh umocnených na druhú, čím váhy s vysokou hodnotou nadobúdajú ešte vyššiu hodnotu, čo zároveň zvyšuje efekt regularizácie vysokých váh spätným šírením. Váhy sú týmto spôsobom znížené ale nedosahujú nulu, čiže L2 regularizácia nemá za výsledok preriedenie spojov v neurónovej sieti. V ideálnom prípade by mali byť dostatočne znížené váhy neurónov, ktoré slúžili pre klasifikáciu prebytočných vlastností. Ak by bola ako chybová funkcia opäť zvolená kvadratická odchýlka, L2 regularizáciu je možné vyjadriť vo vzťahu 2.3. [19, 20]

$$O = \frac{1}{2} \sum_{j=1}^n (d_j - o_j)^2 + \lambda \sum_{j=1}^n (w_j)^2 \quad (2.3)$$

3 Problematika nevyrovnaného datasetu

3.1 Formulácia problému

Dataset, v ktorom sa nachádzajú dáta kategorizované do tried, sa označuje za vyrovnaný alebo za nevyrovnaný v závislosti od pomeru týchto tried. V prípade, že je binárny dataset nevyrovnaný, jeho triedy sa označujú ako menšina a väčšina. Ako už bolo spomenuté v predchádzajúcich kapitolách, problém s nevyrovnaným datasetom spočíva v číselnej prevahe väčšiny nad menšinou. Následkom tohto problému je, že váhy neurónovej siete sú upravované najmä väčšinou skupinou, ktorá prichádza na vstup modelu častejšie. Okrem nízkej frekvencie výskytu, menšina môže byť zastúpená aj malým množstvom dôležitých dát a príznakov. Neurónová sieť teda dosahuje nízku presnosť v klasifikácii menšinovej skupiny a jej výstup je nedôveryhodný. Podľa literatúry problém nastáva najmä vtedy, keď percento menšiny klesne pod 5 % [21].

Situácia, kedy nie je možné získať dostatok dát menšinovej skupiny, je v praxi pomerne častá. Napríklad z medicínskeho hľadiska sa často vyskytujú pacienti so vzácnou chorobou, prípadne novo vzniknutou, o ktorých nie je možné nazbierať viac dát. Je potrebné poznamenať, že je to práve minorita alebo vzácne vzorky, ktoré sú bodom záujmu a je potrebné ich skúmať. [22]

Pri hľadaní riešenia na problém nevyrovnaného datasetu je potrebné brať do úvahy aj ďalšie dôležité charakteristiky datasetu, ktoré rozhodujú o správnej klasifikácii vzoriek. Dôležitejšia než rovnováha dát je ich dobrá reprezentácia. Teda aby boli obe skupiny dobre prezentované vhodnými vzorkami s charakteristikami, ktoré vystihujú danú triedu. Taktiež je potrebné, aby jednotlivé triedy boli voči sebe exkluzívne. Rovnako bolo zistené, že nekomplexné a lineárne separovateľné problémy nie sú ovplyvnené nevyrovnanosťou, naopak pri zvýšenej komplexite narastá citlivosť modelu na nevyrovnanosť datasetu. [22, 23]

Ďalej je potrebné poznamenať, že oveľa lepším ukazovateľom než percentá alebo pomer, je v niektorých prípadoch absolútny počet menšiny. To je možné uviesť na príklade, kde by bol k dispozícii dataset, v ktorom sa celkovo nachádza 100 tis. súborov a menšina by bola prezentovaná 1 %. Menšina teda obsahuje 1 tis. vzoriek, čo je dostatočný počet pre dosiahnutie prijateľnej úspešnosti, pokiaľ sú vzorky danej triedy reprezentatívne. [22, 24]

3.2 Metódy hodnotenia predikcie modelu

Predikcia modelu môže vykazovať zavádzajúci výsledok, ak je hodnotená nevhodnými metrikami. Ako príklad je možné uviesť porovnávanie predikovaného výstupu

siete so správnym výstupom. Pokiaľ by teda väčšinová skupina bola zastúpená v 99 % vzoriek, model dosiahne presnosť 99 %, ak označí všetky vzorky ako patriace väčšine. Menšinovú skupinu, ktorá sa vyskytuje v 1 % bude taktiež nesprávne klasifikovať ako patriacu do väčšinovej. Vo výsledku takto natrénovaný model nerozozná menšinovú skupinu, a teda nie je dosiahnutý cieľ správnej klasifikácie, napriek tomu presnosť klasifikácie dosiahla 99 %. [22]

Pre správne zhodnotenie úspešnosti klasifikácie je potrebné odvodiť presnosť predikcie modelu pomocou matice zámien. Matica zámien vyjadruje vzťah medzi výstupom modelu a skutočnou hodnotou. Jej ukážku je možné vidieť na obrázku 3.1. Pre prehľadnosť je vhodné označiť väčšinovú triedu ako negatívnu a menšinovú ako pozitívnu.

		Skutočná hodnota	
		Pozitívna (1)	Negatívna (0)
Predikovaná hodnota	Pozitívna (1)	TP	FP
	Negatívna (0)	FN	TN

Obr. 3.1: Matica zámien

Kde TP („True pozitive“) značí pravdivo pozitívny výstup. To znamená, že daná vzorka vstupujúca do modelu patrila do pozitívnej skupiny a bola modelom správne klasifikovaná ako patriaca do pozitívnej skupiny. Ďalej TN („True negative“) značí pravdivo negatívny výstup, teda negatívna vzorka bola modelom správne klasifikovaná ako negatívna. Nesprávne zaradené vzorky popisujú vzťahy FN („False Negative“) a FP („False Positive“), pričom FN je pozitívna vzorka, nesprávne klasifikovaná ako negatívna, teda falošne negatívna. Naopak FP je negatívna vzorka, nesprávne klasifikovaná ako pozitívna, teda falošne pozitívna. [25]

Z matice zámien je možné odvodiť nasledujúce vzťahy:

$$Presnosť = \frac{TP + TN}{TP + FN + FP + TN} \quad (3.1)$$

Presnosť definovaná vzorcom 3.1, s akou presnosťou model celkovo klasifikuje vstupné dáta. Avšak, môže byť značne skreslená, keďže určuje celkovú klasifikáciu

výstupu pre obe triedy spolu. Celková presnosť je napriek tomu dobrým ukazovateľom priebehu učenia modelu. Aby bolo možné určiť správnosť klasifikácie výstupu do jednotlivých tried, je potrebné sa orientovať aj vzťahmi vysvetlenými nižšie.

Pozitívna prediktívna hodnota, z anglického „positive predictive value“ (*PPV*) je vyjadrená rovnicou 3.2. Udáva pomer správnych pozitívnych výstupov k celkovo pozitívnym výstupom označených modelom. Inými slovami, pozitívna prediktívna hodnota udáva pravdepodobnosť, že výstup bude správne pozitívny.

$$PPV = \frac{TP}{TP + FP} \quad (3.2)$$

Vo vzorci 3.3 je senzitivita (*SE*) mierou pravdivej pozitivity („true positive rate“, skratkou *TPR*). Udáva pomer klasifikovaných pozitívnych ku všetkým pozitívnym (spolu s falošne negatívnymi), čiže presnosť, s akou je klasifikovaná pozitívna skupina.

$$SE = TPR = \frac{TP}{TP + FN} \quad (3.3)$$

Údaj o tom, aký je pomer medzi naozaj negatívnymi vzorkami a negatívnymi výstupmi celkovo spolu s falošne pozitívnymi, udáva špecificita (*SP*). Je vyjadrená vzťahom 3.4. Označuje sa aj ako miera pravej negativity, „true negative rate“, skratkou *TNR*.

$$SP = TNR = \frac{TN}{TN + FP} \quad (3.4)$$

V praxi je často používané *F*-skóre, ktoré kladie vyšší dôraz na správne pozitívny výstup. Je definované vzťahom 3.5. [25, 22]

$$F - \text{skóre} = \frac{2TP}{(2TP) + FN + FP} \quad (3.5)$$

3.3 Riešenia problému nevyrovnaného datasetu

V literatúre je možné nájsť dve hlavné kategórie prístupov k problému nevyrovnaného datasetu. Prvou sú implementácie metód na úrovni datasetu, ktoré upravujú dataset tak, aby bol vhodný pre tréning neurónovej siete. Ako príklad je možné uviesť metódy generujúce syntetické vzorky menšinovej skupiny alebo metódy, ktoré privádzajú na vstup neurónovej siete menšinovú skupinu častejšie s určitou pravdepodobnosťou. Druhou kategóriou sú algoritmy, ktoré upravujú priebeh učenia tak, aby bol prispôsobený nevyrovnanému datasetu. Sem sa radia napríklad metódy upravujúce váhu vzoriek v chybovej funkcii. Taktiež je možné zdefinovať hybridné metódy, ktoré zahŕňajú kombináciu vyššie uvedených metód. Napríklad kombináciu

dátovej augmentácie a váženej vzájomnej entropie. Obe metódy sú bližšie opísané v nasledujúcich kapitolách. [23, 22]

3.3.1 Metódy na úrovni datasetu

Náhodné podvzorkovanie a prevzorkovanie

Náhodné podvzorkovanie spočíva v zmenšení počtu vzoriek väčšinovej skupiny použitých pre trénovanie. Problémom je, že podvzorkovanie môže odstrániť aj veľa dôležitých dát a reprezentatívnych vzoriek väčšinovej skupiny, ktoré môžu byť užitočné pre správne naučenie modelu. Náhodné prevzorkovanie spočíva v častejšom privádzaní vzoriek menšinovej skupiny na vstup neurónovej siete. To znamená, že s určitou pravdepodobnosťou, ktorá závisí od pomeru nevyrovnanosti, sa vzorky menšinovej skupiny na vstupe neurónovej siete opakujú tak, aby prichádzali na vstup rovnako často ako vzorky väčšinovej skupiny. Bez implementácie tejto metódy by prišla počas jednej epochy na vstup neurónovej siete každá vzorka len raz. Implementácia náhodného prevzorkovania môže mať aj negatívne následky, pretože dôslednejšie vzorkovanie trvá dlhšie a pri trénovaní môže dôjsť k preučeniu menšinovej skupiny. [22, 21]

Nedostatky predchádzajúcich metód viedli k vzniku algoritmov ako dynamické vzorkovanie, ktoré dosahovali o niečo lepšie výsledky. Princíp tohto algoritmu má analógiu v ľudskom učení, kedy sa model doučí to, čo v predchádzajúcej epoche klasifikoval nesprávne. Počas spätného šírenia chyby je na vstup modelu opäť posielaná vzorka, ktorú model predtým zaradil nesprávne. V prípade nevyrovnaného datasetu sa teda väčšinou jedná o menšinovú skupinu. [26]

Generovanie syntetických vzoriek

Generovanie syntetických vzoriek, v angličtine „Synthetic Minority Over-Sampling Technique“ (SMOTE), je metóda, ktorá vytvára nové vzorky pre menšinu interpoláciou medzi existujúcimi vzorkami a ich najbližšími susedmi. Inak povedané, algoritmus náhodne vyberie bod v rozsahu dát menšiny a vypočíta k-najbližších susedov pre tento bod. Následne na spojnici tohto bodu a jedného zo susedov vyberie ďalší bod, ktorý bude reprezentovať syntetickú vzorku. Tento spôsob generovania vzoriek je opakovaný, až kým triedy nie sú voči sebe vybalancované. Algoritmus bol taktiež viacerými spôsobmi modifikovaný pre lepšie výsledky a existuje niekoľko jeho variánt, vrátane hybridných metód. [22, 27]

Dátová augmentácia

Dátová augmentácia je primárne metóda zabráňujúca preučeniu modelu. Spočíva v náhodnej úprave snímok niekoľkými voliteľnými procesmi. Medzi tieto procesy patrí napríklad náhodné orezanie snímky, pootočenie, zrkadlové otočenie, zmena farieb, pridanie šumu alebo úprava jasových hodnôt. Dataset sa vďaka týmto úpravám stáva robustnejším a model sa nepreučuje znaky nevýznamné pre klasifikáciu, ako napríklad pozadie snímok. Vzhľadom na to, že dátová augmentácia týmto spôsobom generuje snímky, ktoré nie sú úplne identické s pôvodnými, je možné túto metódu využiť aj ako metódu eliminácie vplyvu nevyrovnaného datasetu. [28]

„Hard data mining“

„Hard data mining“ je možné preložiť ako ťaženie obtiažnych dát. Princíp tejto metódy spočíva v tom, aby sa model naučil správne klasifikovať vzorky alebo dáta, ktorých klasifikácia je obtiažna, inak povedané „hard examples“, teda ťažké vzorky. To, či je klasifikácia obtiažna, je vyhodnocované na základe vysokej chybovej funkcie. Presnejšie, po privedení sady („batch“) na vstup neurónu je vypočítaná chyba a gradient. Následne sú vzorky zoradené zostupne podľa veľkosti chybovej funkcie. Zo zoradených výsledných chybových funkcií je pre spätné šírenie vybraných prvých k -vzoriek s vysokou chybou. Tieto vzorky majú väčší vplyv na úpravu váh, preto sú spätne šírené. Týmto spôsobom, je prispievané k efektívnejšiemu učeniu a presnejšiemu výsledku neurónovej siete.

Táto metóda nachádza uplatnenie nielen v nevyrovnaných datasetoch, kde sú obtiažnymi vzorkami dáta menšinovej triedy, ale aj v datasetoch, ktoré jednoducho obsahovali ťažké príklady, teda jednotlivé vzorky neboli pre triedu reprezentatívnymi. „Hard data mining“ je preto všeobecné označenie a v dostupných štúdiách sa jednotlivé implementácie pod týmto názvom môžu líšiť. Medzi dôležité štúdie súvisiace s nevyrovnaným datasetom patrí napríklad „Class Rectification Hard Mining for Imbalanced Deep Learning“ [29]. [29, 30, 31]

3.3.2 Metódy na úrovni algoritmu

Vážená vzájomná entropia

Vzájomná entropia s rovnomernými váhami, teda klasická vzájomná entropia, nezohľadňuje frekvenciu výskytu skupín v datasete, čo viedlo k vzniku váženej vzájomnej entropie (VVE). Vážená vzájomná entropia vznikla pridaním váhových faktorov ku klasickému vzájomnej entropii. Tieto faktory je možné vidieť vo vzorci 3.6. V originálnej štúdii [32] autor uvádza, že prvý modifikujúci faktor (váhy) $\omega_{o,c}y_{o,c}$ závisí od

spolahlivosti pozitívneho označenia výstupu a druhý faktor závisí od spolahlivosti negatívneho označenia.

$$VVE = - \sum_{c=1}^M \omega_{o,c} y_{o,c} \log(p) + \bar{\omega}_{o,c} y(1 - y_{o,c}) \log(1 - p) \quad (3.6)$$

kde ω sú váhy pozitívnej triedy a $\bar{\omega}$ sú váhy negatívnej triedy.

Zjednodušene je možné povedať, že váhy implementované v chybovej funkcii môžu upravovať, s akou mierou sa budú meniť váhy modelu. Napríklad, pokiaľ budú zvolené nižšie váhy v chybovej funkcii pre danú triedu, váhy modelu budú upravované menej. Naopak, väčšie budú váhy pre danú triedu, tým bude trieda dôležitejšia. Váhy môžu byť zvolené používateľom, alebo vybrané náhodne a upravené počas procesu tréovania. [32, 33]

„Focal loss function“

„Focal loss function“ (FL) je metóda, ktorá tiež upravuje vzájomnú entropiu. Chybová funkcia je pozmenená tak, aby ľahko klasifikované vzorky mali nižší dopad na zmenu váh. Táto úprava je dosiahnutá vynásobením chybovej funkcie modifikujúcim faktorom $(1 - p_t)^\gamma$, taktiež uvedeným v kompletnej rovnici 3.7. To, či sú vzorky jednoducho klasifikovateľné, je určované premennou p^t . Ako jednoducho klasifikované vzorky je možné označiť tie, pre ktoré platí, že $p_t \rightarrow 1$. [22, 34]

$$FL(p_t) = -\alpha_t(1 - p_t)^\gamma \log(p_t) \quad (3.7)$$

Hyper parameter $\gamma \geq 0$ upravuje ako veľmi je znížená váha jednoducho klasifikovateľných vzoriek. Výnimočný prípad je $\gamma = 0$, pretože modifikujúci faktor sa v tomto prípade rovná jednej, a teda ide o klasickú vzájomnú entropiu. Najlepšie výsledky sú dosiahnuté pri $\gamma = 2$ [34]. Hyper parameter $\alpha_t \geq 0$ slúži, naopak, na vyváženie nevyrovnanosťami medzi jednotlivými triedami. Predvolená hodnota hyperparametru α je nastavená ako 0,25. [22, 34]

4 Použité datasety

4.1 Microsoft ASIRRA

Ako prvý bol použitý dataset s názvom ASIRRA („Animal Species Image Recognition for Restricting Access“), ktorý vytvorila spoločnosť Microsoft v roku 2007 uvedený v zdroji [35]. Je voľne dostupný online na stránke Kaggle [36]. Dataset je vopred rozdelený na testovaciu a trénovaciu (validačnú) množinu v osobitných priechkoch. Trénovacia množina obsahuje 25 tis. fotografií, určených pre jednoduchú binárnu klasifikáciu zvierat na mačky a psy. Pôvodný dataset je vyrovnaný, teda polovica fotografií patrí do skupiny mačiek a zvyšná polovica patrí do skupiny psov. Každá položka datasetu vo svojom názve obsahuje označenie príslušnosti pre jednu z dvoch skupín. Položky sú pomenované v anglickom jazyku označením „cat“, čiže mačka alebo označením „dog“, čiže pes.

V testovacej množine sa nachádza 12,5 tis fotografií, ktoré majú podobné vlastnosti ako fotografie trénovacej množiny. Líši sa iba tým, že neobsahuje označenia „cat“ a „dog“. V názve položiek sa nachádzajú len indexy jednotlivých fotografií. V oboch množinách sa nachádza vyrovnaný dataset, teda jednotlivé skupiny sú v pomere 1:1.

Rozmery každej fotografie sa líšia, pričom fotografia s najväčšími rozmermi má šírku 1023 pixelov a výšku 768 pixelov. Najmenšia má šírku 60 pixelov a výšku 39 pixelov. Všetky položky sa nachádzajú vo formáte „jpeg“.

V datasete sa nachádzajú fotografie mačiek a psov s vysokou rôznorodosťou. Fotografie boli fotené pri rôznych svetelných podmienkach, a teda obsahujú rôzne jasové úrovne. Zvieratá na fotografiách majú rôzne farby a taktiež ich môžeme zaradiť do rôznych plemien. Sú fotené pri rôznych uhloch s rôznou veľkosťou záberu fotografie. Na niektorých fotografiách sa nachádza len hlava zvierata a na niektorých je zobrazené celé telo, prípadne sú viditeľné len niektoré časti tela zvierata. Pozadie fotografií sa taktiež líši či už farebnosťou, textúrou alebo jasom. V niektorých prípadoch sú zvieratá aj prekryté objektmi (napríklad sa nachádzajú za plotom alebo je pred nimi umiestnený iný objekt). Na niektorých fotografiách sa nachádzajú dve a viac zvierat rovnakého druhu, takže dataset nie je obmedzený na jeden objekt na fotografii. V datasete sa nenachádzajú len dospelé mačky a psy, ale zvieratá sú rôzne staré. V niektorých prípadoch sa na fotografiách spolu so zvieratom nachádza aj človek. V datasete je taktiež možné nájsť okrem fotografií kreslené obrázky týchto dvoch skupín zvierat. Ukážku datasetu je možné vidieť na obrázku 4.1:



Obr. 4.1: Ukážka datasetu Microsoft ASIRRA použitého pre tréovanie neurónovej siete v rôznych nevyrovnanostiach

4.2 Dataset skoliózy, spondylolistézy a zdravej chrbtice

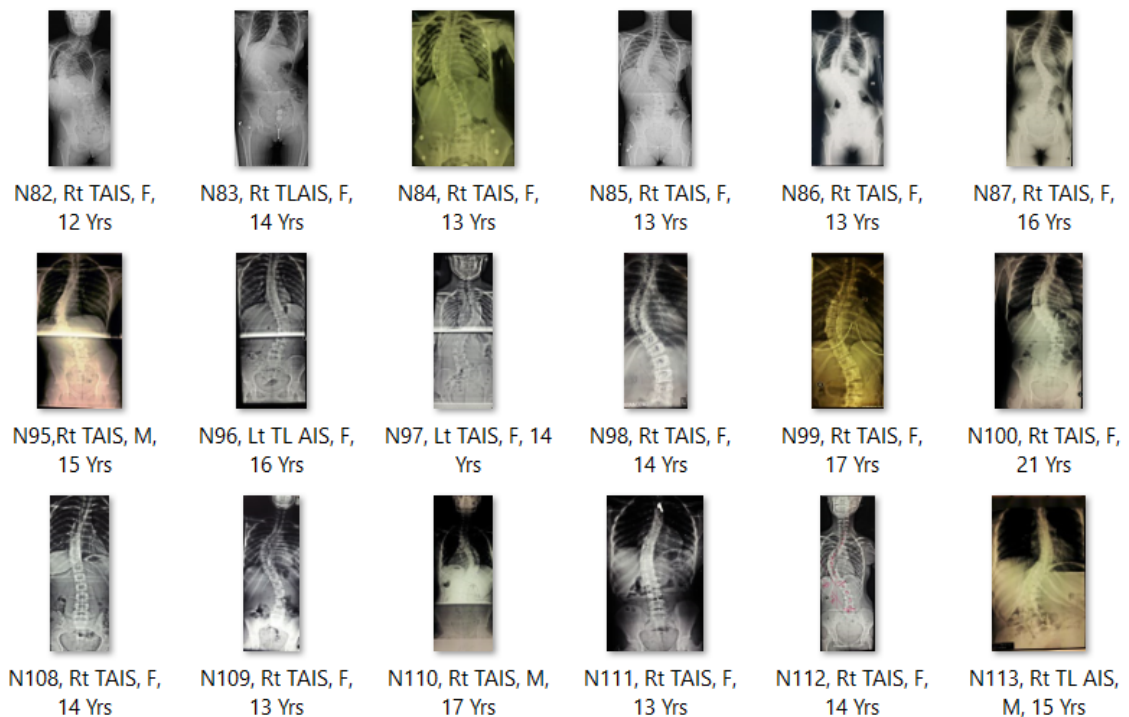
Dataset bol poskytnutý v roku 2017 ako voľne dostupný na stránke Mendeley Data [37] univerzitou „Jordan University of Science and Technology“. Dáta boli nasnímané v nemocnici „King Abdullah University Hospital“. Dataset je zameraný na röntgenové snímky patológií chrbtice, pričom obsahuje 3 skupiny pacientov. Prvou skupinou sú zdraví pacienti v počte 71, druhou sú pacienti so spondylolistézou v počte 79 a tretou je 188 pacientov so skoliózou.

Spondylolistéza je ochorenie, pri ktorom má pacient sagitálne posunutý stavce voči susednému stavcu. Príčinou môže byť vrodené ochorenie, úraz, alebo nestabilita chrbtice v pokročilom veku. Skolióza je ochorenie typické rotačnou deformitou chrbtice vo frontálnej rovine. Príčinou je zvyčajne porucha vývoja s motorickým deficitom, úraz alebo poruchou formácie chrbtice, ktorá sa môže vyskytnúť už v embryonálnom štádiu. [38, 39]

Jednotlivé patológie a zdravá chrbtica boli rozdelené do samostatných priečinkov. Jednotlivé snímky v názve obsahujú anotácie k pacientom. Uvádzajú číslo snímky, opis patológie, pohlavie a vek pacienta. Snímky pochádzajú z rengenového vyšetrenia, avšak často majú nízku kvalitu, sú rôzne zafarbené, prípadne vznikli ako spojenie dvoch rengenových snímok, viď. obrázok 4.2. Snímky sú taktiež rôzne orezané. Rozmery a rozlíšenie jednotlivých snímok sa líšia. Snímka s najväčším rozlíšením

obsiahnutá v datasete má výšku 3733 pixelov a šírku 1710 pixelov, naopak snímka s najmenším rozlíšením má výšku 700 pixelov a šírku 149 pixelov. Všetky snímky sú vo formáte „jpeg“.

Táto práca je zameraná na binárnu klasifikáciu, preto na trénovanie modelu boli použité len dve skupiny. Prvou boli zdraví pacienti, ktorí tvorili 28 % datasetu a pacienti so skoliózou, tvoriaci 72 % datasetu. Z tohto dôvodu bude v práci tento dataset referovaný len ako „dataset skoliózy a zdravej chrbtice“. Dataset bol rozdelený na validačný (20 %) a trénovací (80 %).



Obr. 4.2: Ukážka datasetu skoliózy, spondylolistézy a zdravej chrbtice. Na obrázku je čiastočný obsah prierečného rengenového snímku pacientov so skoliózou.

5 Návrh experimentov

Na základe spracovanej teórie bolo navrhnutých 5 experimentov, ktoré budú následne vymenované. Všetky experimenty boli z programového hľadiska spracované v programovacom jazyku Python 3.9.13, pomocou vývojového prostredia Spyder 5.4.2.

Ako prvý experiment bola vyhodnotená klasifikácia neurónovej siete bez pridania akejkoľvek metódy eliminácie vplyvu nevyrovnaného datasetu na tréning neurónovej siete. Ďalej bola implementovaná jedna metóda na úrovni algoritmu, teda vážená vzájomná entropia a tri metódy na úrovni datasetu, menovite dátová augmentácia, náhodné prevzorkovanie a SMOTE.

Účinnosť jednotlivých metód bola vyhodnotená pre dva rôzne datasety, pričom prvý (Microsoft ASIRRA, bližšie opísaný v 4.1) bol upravený na rôzne stupne nevyrovnanosti. To znamená, že ako prvé bolo vyhodnotené tréning vyrovnaného datasetu, v ktorom každá skupina tvorí 50 %, čo odpovedá pomeru skupín 1:1. Následne bola menšinová skupina obsiahnutá v 30 % datasetu, čo odpovedá pomeru 7:3. A posledný experiment mal obsah menšinovej skupiny 10 %, čiže v pomere 9:1. Druhý dataset bol vyhodnotený s nevyrovnanosťou, ktorá sa v datasete pôvodne nachádzala, to znamená 72 % väčšinovej skupiny a 28 % menšinovej skupiny.

Pri vyhodnocovaní každého experimentu budú ich najdôležitejšími ukazovateľmi špecificita a senzitivita, pozitívna prediktívna hodnota a F-skóre.

5.1 Predspracovanie dát

Oba datasety boli rozdelené na tréningovú a validačnú množinu, pričom tréningová obsahovala 80 % snímok a validačná 20 % snímok. Rozdelenie bolo realizované funkciou „train_test_split“ z knižnice „Scikit-learn“ [40]. Táto funkcia náhodne rozdelí dáta na validačnú a tréningovú množinu. Snímky následne boli uložené do samostatných priečinkov pre jednotlivé nevyrovnanosti, aby tréningová aj validačná množina obsahovala konštantné dáta pre všetky metódy implementované v experimentoch. Absolútne počty vzoriek v jednotlivých skupinách sú uvedené v tabuľke 5.1 a tabuľke 5.2.

Predspracovanie ďalej spočívalo v prevedení všetkých fotografií nachádzajúcich sa v datasete na rovnakú veľkosť. To znamená, že po transformácii majú obrázky výšku aj šírku 224 pixelov, čím je znížená výpočetná náročnosť modelu. Pixely boli prevedené na vektor knižnice „Pytorch“. Jednotlivým obrázkom boli taktiež priradené označenia 0 alebo 1 podľa toho, do ktorej skupiny patria. Pre dataset Microsoft ASIRRA boli mačky (väčšinová skupina) označené ako 0 a psy, (menšinová skupina) označené ako 1. V prípade datasetu skoliózy a zdravej chrbtice by štandardne boli

volené označenia 0 pre zdravú chrbticu a 1 pre skoliózu. Avšak vzhľadom na to, že zdraví pacienti sú v menšine, je nutné túto skupinu označiť ako 1 a väčšinou (pacienti so skoliózou) ako 0. Týmto spôsobom je možné použiť rovnaké metriky pre oba datasety.

Po správnom označení skupín bol použitý dataloader z knižnice „Pytorch“. Úlohou dataloaderu je rozdeliť dáta na jednotlivé sady („batch“) a privádzať ich na vstup neurónovej siete.

Tab. 5.1: Rozdelenie vzoriek v absolútnom počte pre dataset Microsoft ASIRRA.

vyrovnaný dataset 1:1	trénovací [-]	validačný [-]
psy (1)	10000	2500
mačky (0)	10000	2500
pomer nevyrovnanosti 7:3	trénovací [-]	validačný [-]
psy (1)	4270	1087
mačky (0)	10015	2485
pomer nevyrovnanosti 9:1	trénovací [-]	validačný [-]
psy (1)	1100	288
mačky (0)	10010	2490

Tab. 5.2: Rozdelenie vzoriek v absolútnom počte pre dataset skoliózy a zdravej chrbtice.

absolútne počty v datasete	trénovací [-]	validačný [-]
normal (1)	58	13
skolioza (0)	149	39

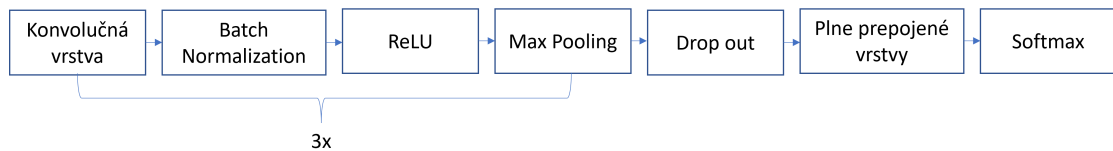
5.2 Použitá neurónová sieť

5.2.1 Architektúra neurónovej siete

Pre navrhnutý experiment bola použitá jednoduchá voľne dostupná neurónová sieť typu enkodér, prevzatá zo stránky [41]. Vývojový diagram architektúry je možné vidieť na obrázku 5.1.

Prvé tri vrstvy pozostávajú zo spojenia 2D konvolučnej vrstvy, 2D „Batch Normalizácie“ a nelineárnej vrstvy s aktivačnou funkciou ReLU a zakončením podvzorokovacou vrstvou („Max Pooling“).

Do konvolučnej vrstvy vstupuje obrázok rozdelený na tri farebné kanály, čiže prvý parameter bol volený 3. Ďalej je nutné zadať počet výstupných kanálov, zvolený



Obr. 5.1: Architektúra implementovanej neurónovej siete.

ako 16, veľkosť konvolučného filtra, ktorá bola zvolená ako 3. Doplnenie nulami („padding“) bolo nastavené na 0, čiže neprebiehalo a posun okna („stride,“) bol nastavený na 2. V ďalších dvoch vrstvách do konvolúcie vstupoval rovnaký počet kanálov ako bol na výstupe predchádzajúcej vrstvy a počet výstupných kanálov bol dvojnásobný.

Parameter nastavený pre „Batch Normalizáciu“ je vždy rovný počtu výstupných kanálov 2D konvolúcie. Aktivačná funkcia pre nelineárnu (korekčnú) vrstvu bola volená ReLU. Na záver prebehol „MaxPooling“, kde voleným parametrom bola veľkosť podvzorkovávajúceho filtra ako 2.

Výstup z týchto vrstiev ďalej prechádzal na lineárnu vrstvu, ktorá slúži ako plne prepojená vrstva. V architektúre je taktiež zaimplementovaný „drop out“, z dôvodov uvedených v kapitole 2.2.1. Hodnota „drop out“ bola nastavená na 0,5. Výstup druhej plne prepojenej vrstvy je zredukovaný na 2 výstupné kanály, ktoré sú zároveň vstupom do aktivačnej funkcie Softmax.

5.2.2 Hyperparametre neurónovej siete

Prvým nastavovaným hyperparametrom je krok učenia, ktorý je optimalizovaný algoritmom Adam, spomenutým v 1.5.3. Tento algoritmus vykazoval najlepšie výsledky podľa originálnej štúdie pre zvolenú hodnotu 0,001 [9]. Z tohto dôvodu je rovnaká hodnota zadefinovaná aj pre tento experiment. Ďalším hyperparametrom je veľkosť sady, pre ktorú bolo zvolených 100 vzoriek. Veľkosť sady ovplyvňuje aj mieru preučenia, teda je to spôsob regularizácie a môže eliminovať uviaznutie v lokálnom extréme. Posledným hyperparametrom je počet epoch, ktorý bol zvolený 120 - tento parameter bol určený sledovaním nastavenia rôznych hodnôt a bol hľadaný počet epoch, pre ktorý sieť dosahuje čo najmenšiu chybu a zároveň nedochádza k preučeniu.

Dataset, na ktorý bol experiment rozšírený, obsahoval odlišné hyperparametre z dôvodu odlišnej veľkosti datasetu. Vzhľadom na to, že dataset obsahoval len 206 snímok na tréning, veľkosť sady bola volená 20 a počet epoch na 50, pretože od 50. epochy sa model už preučoval. Krok učenia zostal nastavený na 0,001.

5.2.3 Chybová funkcia a optimalizačné algoritmy

V práci bola implementovaná klasická vzájomná entropia definovaná vzorcom 1.10 v kapitole 1.5.2. Na jej implementáciu bola použitá knižnica „Pytorch“, v ktorej je táto chybová funkcia zabudovaná. Implementácia prebehla podľa dokumentácie [42].

V rámci optimalizačných algoritmov bol použitý Adam optimizer z knižnice „Pytorch“, ktorý okrem optimalizácie kroku obsahuje aj voliteľný parameter L2 regularizácie, ktorý regularizuje zmenu váh, a tým zabraňuje preučeniu. Tento algoritmus je bližšie opísaný v kapitole 2.2.3. Odporúčané nastavenie hodnoty tohto parametru je 0,0001. Avšak, pri použití tejto hodnoty sa model stále preučoval, experimentálne bola teda stanovená hodnota 0,01. Táto hodnota bola použitá pre regularizácii oboch datasetov.

5.3 Implementácia jednotlivých experimentov

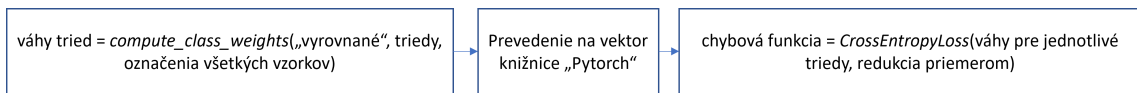
5.3.1 Vážená vzájomná entropia

Na realizáciu experimentu s váženou vzájomnou entropiou bola použitá rovnaká funkcia ako pre vzájomnú entropiu. Avšak, zatiaľ čo pri implementácii vzájomnej entropie do funkcie nevstupujú žiadne parametre, tak pre implementáciu vázenej vzájomnej entropie platí, že prvým parametrom funkcie sú zadané váhy pre každú triedu. Pre jednotlivé nevyrovnanosti boli nastavené váhy pomocou funkcie z knižnice „Scikit-learn“, kde sú jednotlivé váhy počítané podľa vzorca 5.1.

$$v_i = \frac{n}{m \cdot u_i} \quad (5.1)$$

kde v je výsledná váha pre i -tú triedu, n je celkový počet vzoriek, m je počet tried a u je počet unikátnych vzoriek pre danú triedu.

Do funkcie vstupuje niekoľko parametrov, pričom prvý parameter funkcie zabezpečuje, aby bol využitý vzorec na výpočet váh, pokiaľ by parameter chýbal, váhy by mali rovnakú hodnotu pre všetky triedy datasetu. Ďalej do funkcie vstupujú unikátne triedy (0,1) a posledným parametrom sú označenia všetkých vzorkov (0,0,0,... 1). Aby bolo možné váhy použiť v modeli, je potrebné ich previesť na vektor knižnice „Pytorch“. Váhy v správnom tvare neskôr vstupujú ako parameter do chybovej funkcie, spolu s ďalším parametrom chybovej funkcie, ktorý udáva ako má byť počítaná chyba. V tomto prípade je nastavený parameter „mean“, čiže chyba je priemerovaná skrz jednotlivé sady.

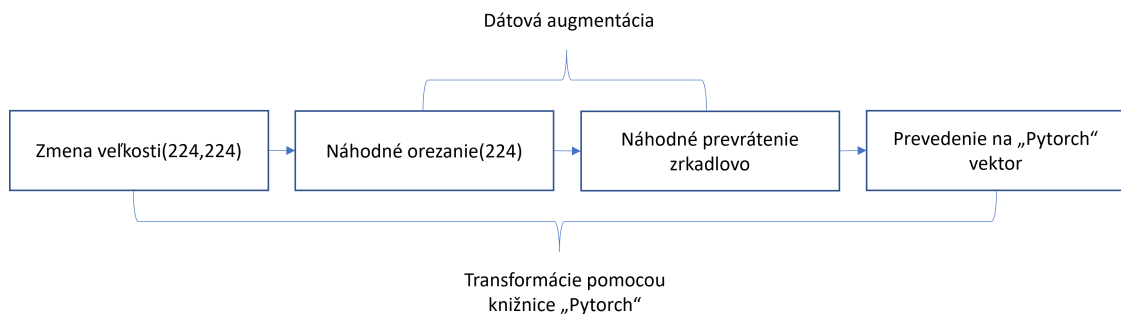


Obr. 5.2: Vývojový diagram implementácie váhovanej vzájomnej entropie.

5.3.2 Dátová augmentácia

Dátová augmentácia implementovaná v tejto práci pozostáva z niekoľkých procesov, ktoré slúžia k zväčšeniu počtu vzoriek, na ktorých model trénuje. Prvým je náhodné orezanie fotografií, pričom veľkosť fotografií 224×224 je zachovaná. Ďalej sú náhodne vybrané fotografie zrkadlovo otočené a prevedené na vektor knižnice „Pytorch“. Všetky opísané procesy sú uskutočnené pomocou funkcie slúžiacej na transformáciu snímok z knižnice „Pytorch“.

Za účelom dosiahnutia čo najlepších výsledkov pre vyrovnaný aj nevyrovnaný dataset pri implementácii dátovej augmentácie nebola použitá L2 regularizácia vzhľadom na to, že dátová augmentácia predchádzala problému preučenia. Pokiaľ bola L2 regularizácia implementovaná, dochádzalo k horším výsledkom a model sa v tomto prípade poduchoval.



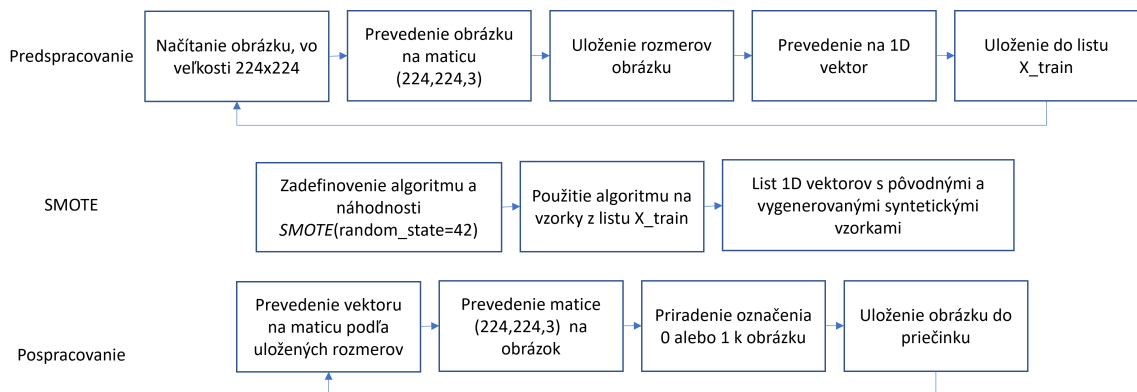
Obr. 5.3: Vývojový diagram implementácie dátovej augmentácie.

5.3.3 SMOTE

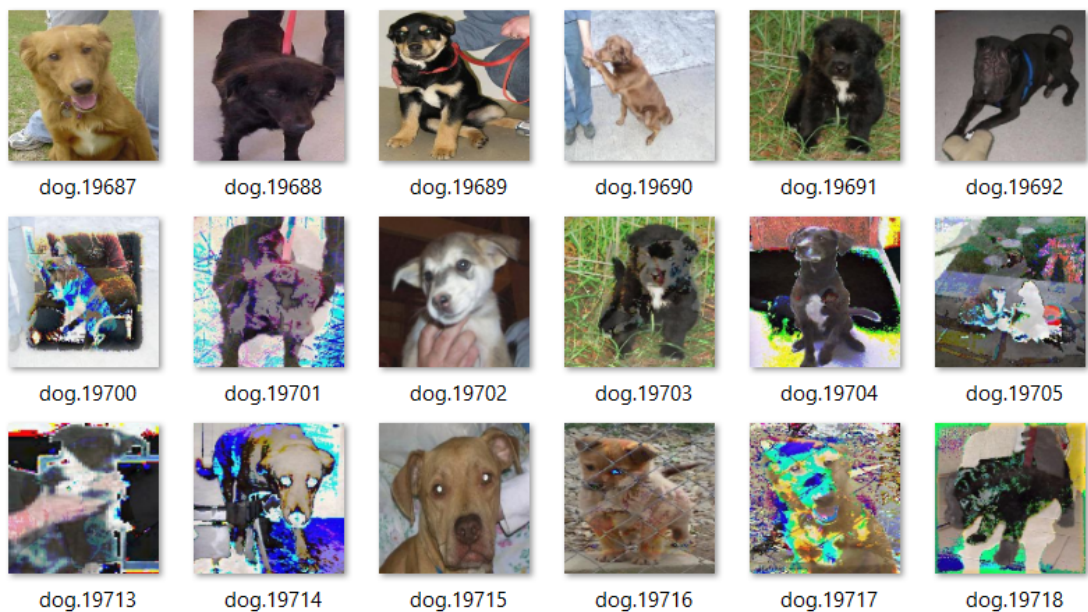
SMOTE bol uskutočnený „off-line“. To znamená, že k jednotlivým skupinám boli vytvorené syntetické vzorky v samostatnom skripte, následne boli uložené a vznikol tak nový dataset, ktorý bol neskôr použitý pre trénovanie siete. Tento proces zahŕňal prevedenie datasetu na list ciest, odkiaľ boli získané názvy jednotlivých fotografií. Názvy boli prevedené na 0 a 1 v závislosti od prislúchajúcej skupiny a uložené do listu. Neskôr boli použité v metóde SMOTE, do ktorej vstupujú vstupné dáta

(snímky) a označenia. Pri použití metódy sa taktiež volí parameter náhodnosti algoritmu, ktorý môže byť ľubovoľne zvolený. V tomto prípade bol zvolený na 42.

Táto metóda funguje najmä pre 1D dáta, preto bolo nutné jednotlivé snímky uložiť do 1D vektorovej podoby. Výstupmi z algoritmu boli taktiež 1D vektory, ktoré bolo potrebné previesť späť do 3 matíc, ktoré RGB obraz obsahuje. Následne boli snímkam priradené označenia v závislosti od triedy, do ktorej patria a na záver boli uložené v samostatnom priečinku. Vznikol tak dataset obsahujúci pôvodné vzorky, avšak jednotlivé skupiny boli v pomere 1:1 pridaním synteticky vygenerovaných vzoriek. Ukážku datasetu je možné vidieť na obrázku 5.5. Jednotlivé úpravy snímok, zobrazené taktiež na blokovej schéme 5.4 boli uskutočnené pomocou funkcií z knižníc „Keras“, „Pillow“ a „Numpy“.



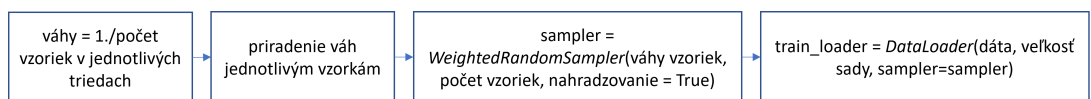
Obr. 5.4: Vývojový diagram implementácie SMOTE.



Obr. 5.5: Ukážka datasetu Microsoft ASIRRA po realizácii algoritmu SMOTE.

5.3.4 Náhodné prevzorkovanie

Na realizáciu experimentu bola použitá funkcia „WeightedRandomSampler“ ktorá bola implementovaná podľa oficiálnej dokumentácie knižnice „Pytorch“ [42]. Do tejto funkcie vstupujú váhy vypočítané pre každú vzorku, počet vzoriek a parameter, pomocou ktorého je možné zvoliť, či sa vzorky budú pri vstupe do neurónovej siete opakovať. Opakovanie vzoriek je pri náhodnom prevzorkovaní žiadúce a preto bol nastavený na hodnotu „True“. Pre implementáciu bolo nutné najskôr získať váhy pre každú skupinu a následne ich priradiť jednotlivým vzorkám. Tieto váhy reprezentujú, s akou pravdepodobnosťou by mala byť vzorka privádzaná na vstup. Postup výpočtu je možné vidieť na obrázku zjednodušenej blokovej schémy 5.6. Funkcia „WeightedRandomSampler“ je s vymenovanými vstupmi zadaná do premennej a následne vstupuje ako parameter do dataloader-u pre trénovacie dáta.



Obr. 5.6: Vývojový diagram náhodného prevzorkovania.

5.4 Štatistické vyhodnotenie výsledkov

Na štatistické vyhodnotenie výsledkov boli použité vzorce uvedené v kapitole 3.2. Podľa týchto vzťahov bola zhotovená funkcia, ktorá vracia na svojom výstupe výsledné hodnoty senzitivity, špecificity, presnosti pozitívnej prediktívnej hodnoty a F-skóre.

V priebehu experimentu bolo vyhodnocované tréovanie modelu a taktiež aj jeho validácia. Pri tréovaní modelu bola vyhodnocovaná celková presnosť, kde bol porovnávaný vstup s výstupom všeobecne pre obe skupiny. Táto hodnota bola často skreslená väčšinou skupinou v prípade nevyrovnaného datasetu. Preto slúžila skôr ako ukazovateľ toho, že sa sieť učí. Celková presnosť bola pre tréovanie aj validáciu v každej epoche vykreslená do grafu. Vďaka tomu bolo možné sledovať aj možnosť, kedy model označil všetky snímky ako patriace väčšinovej skupine.

Pre tréovanie aj validáciu bola taktiež vykresľovaná chybová funkcia, ktorá taktiež slúži ako ukazovateľ správneho učenia. Týmto spôsobom je možné spozorovať, či chybová funkcia klesá a model teda konverguje k správnejmu výsledku, prípadne určiť preučovanie modelu.

Z validačných hodnôt bola graficky vyhodnocovaná aj špecificita a senzitivita pre každú epochu. Je tak možné sledovať, či špecificita alebo senzitivita počas tréovania rástla alebo klesala, prípadne zistiť približné hodnoty, v akých sa pohybovala. Senzitivita v tomto prípade určovala schopnosť modelu naučiť sa rozoznávať menšinou skupinu, označenú ako 1 a špecificita, naopak, schopnosť modelu rozoznávať väčšinou skupinu označovanú ako 0.

Modely boli počas jednotlivých experimentov ukladané, aby mohli byť neskôr načítané a vyhodnotené ich výsledky pri testovaní. Testovanie modelu prebiehalo taktiež pomocou dataloader-u, ktorý bol iterovaný skrz jednotlivé sady s veľkosťou 100 vzoriek pre dataset Microsoft ASIRRA a 20 vzoriek pre dataset zdravej chrbtice a skoliózy. Pre každú sadu bola vyhodnocovaná celková presnosť, F-skóre, pozitívna prediktívna hodnota, špecificita a senzitivita. Výsledné hodnoty jednotlivých sád boli spriemerované a zapísané ako konečné výsledky otestovaného modelu.

6 Výsledky a diskusia

6.1 Učenie s datasetom Microsoft ASIRRA

6.1.1 Učenie bez metódy eliminácie

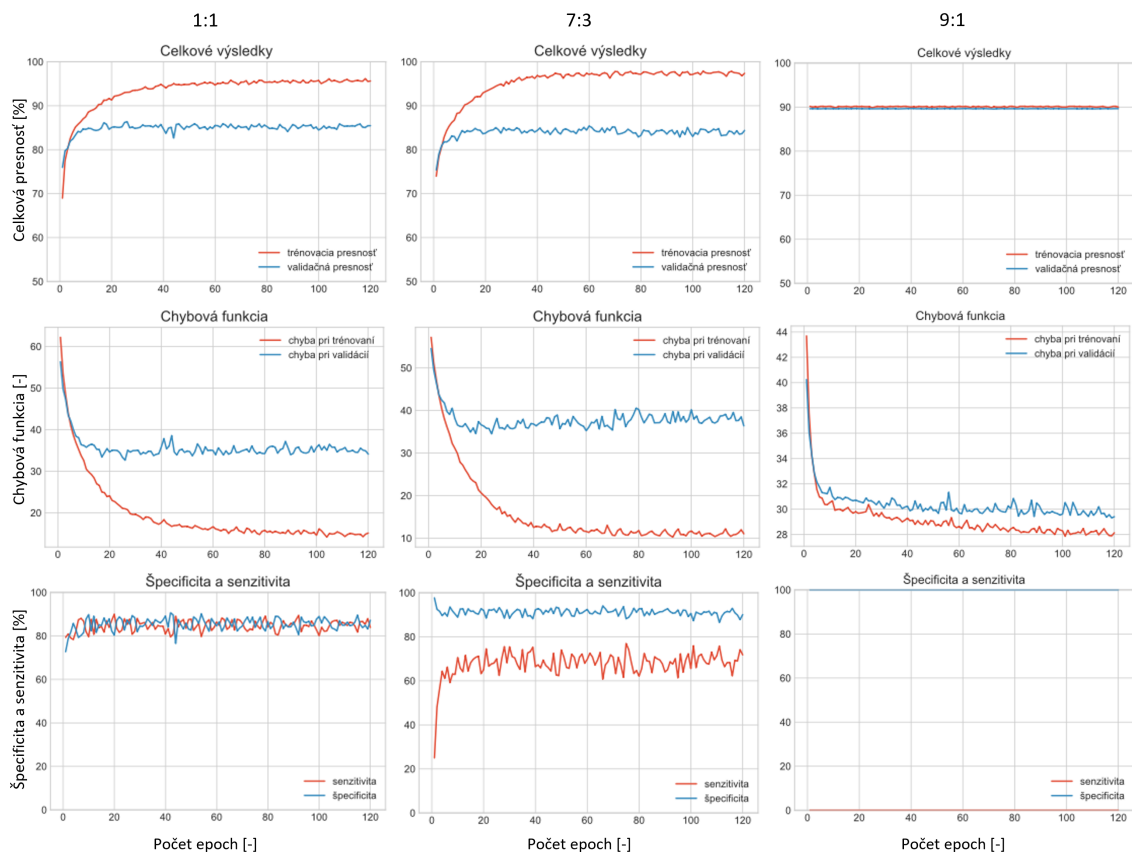
Prvý stĺpec grafických výsledkov zobrazuje výsledky pre vyrovnaný dataset, pričom najskôr je možné vidieť celkové výsledky vyhodnotené vzorcom 3.1, kedy je výstup modelu porovnávaný so správnym označením. Podľa celkovej presnosti, ktorá stúpala, je teda možné vidieť, že model sa učil a podľa chybovej funkcie je možné vidieť, že sa nepreučoval.

Pri experimente s vyrovnaným datasetom sa špecificita a senzitivita pohybovali v približne rovnakých hodnotách, z čoho je možné usúdiť, že model sa naučil obe skupiny s približne rovnakou presnosťou.

V nasledujúcom experimente, kde bola menšinová skupina (psy) obsiahnutá v 30 % datasetu, je možné vidieť, že špecificita dosahuje vyššie hodnoty (okolo 95 %) než senzitivita (okolo 70 %), čiže model sa naučil lepšie väčšinovú skupinu (mačky), než menšinovú (psy). Navyše, chybová funkcia ku koncu mierne rastie, čiže je možné, že model sa začal preučovať.

V poslednom experimente bol pomer skupín väčšinovej k menšinovej 9:1. Z kriviek celkovej presnosti je možné pozorovať, že model pravdepodobne označil všetky vzorky ako väčšinovú skupinu, pretože percento presnosti sa rovná pomeru nevyrovnosti datasetu (v experimente sa nachádza 90 % mačiek). To isté je možné vidieť aj v grafe s krivkami špecificity a senzitivity. Menšinovú skupinu sa model vôbec nenaučil rozoznávať.

Konečné výsledky učenia je možné vidieť v tabuľke 6.1.1. Z výsledkov je zjavné, že skreslená presnosť rástla, zatiaľ čo F-skóre a pozitívna prediktívna hodnota klesala so snižujúcim sa percentom menšinovej skupiny. Taktiež je možné vidieť, ako senzitivita klesala a teda presnosť klasifikácie menšinovej skupiny klesala, zatiaľ čo špecificita rástla so zvyšovaním pomeru väčšinovej skupiny v datasete.



Obr. 6.1: Grafické zobrazenie porovnania výsledkov pri tréovaní so vzájomnou entropiou pre rôzne nevyrovnanosti. Výsledky sú uvedené v pomere väčšiny skupiny k menšiny.

Tab. 6.1: Výsledky testovania modelu natréovaného bez pridania metódy eliminácie.

pomer väčšina:menšina	1:1	7:3	9:1
presnosť [%]	85,6	84,4	89,6
F-skóre [%]	85,3	73,2	0,0
pozitívna prediktívna hodnota [%]	89,0	76,0	0,0
senzitivita [%]	83,9	71,2	0,0
špecifita [%]	87,5	90,4	99,9

6.1.2 Učenie s váženou vzájomnou entropiou

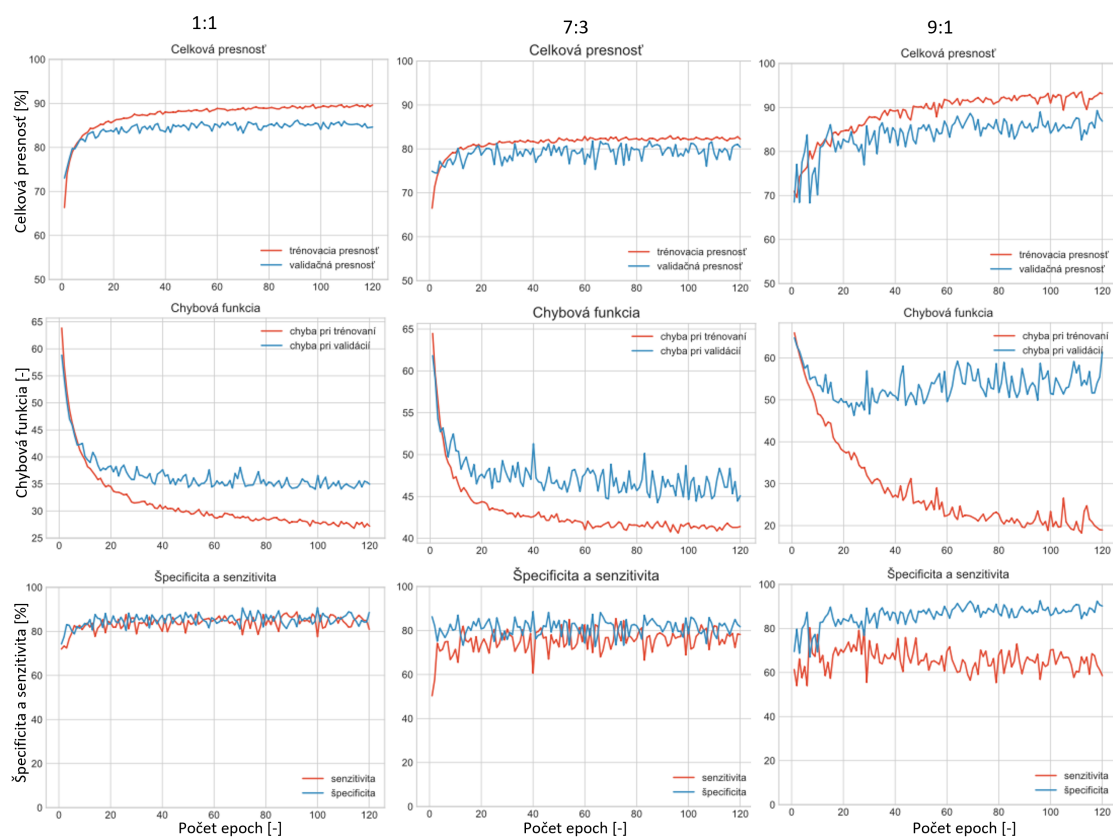
V prvom stĺpci obrázku 6.2, v prípade vyrovnaného datasetu, celková presnosť dosahuje okolo 85 %, to isté platí aj pre špecifitu a senzitivitu, čo znamená že presnosť

klasifikácie mačiek aj psov dosahuje približne rovnaké výsledky. Model sa v tomto prípade naučil rozoznávať obe skupiny s presnosťou 85 %. V tabuľke 6.1.2 je možné vidieť, že pri testovaní modelu bola senzitivita o niečo nižšia aj pri vyrovnanom datasete.

V druhom stĺpci, pre dataset v pomere 7:3, presnosť dosahuje rovnako okolo 80 %. Medzi krivkami špecificity a senzitivity je vo výsledku len malý rozdiel. Obe krivky dosahujú hodnôt okolo 80 %, z čoho je možné usúdiť, že model sa opäť naučil obe skupiny.

V poslednom, treťom stĺpci, kde je menšinová skupina zastúpená už len 10 %, je na krivke chybovej funkcie možné sledovať, že model sa po 20. epoche začal preučovať, pretože validačná chyba stúpa. Tomuto javu by bolo možné zabrániť nastavením vyššej hodnoty regularizácie. Napriek tomu, z kriviek špecificity a senzitivity je možné zhodnotiť, že model sa aj pri tomto pomere nevyrovnanosti učil menšinovú skupinu a pred preučeníím dosiahol presnosť klasifikácie menšinovej skupiny takmer 80 %. Celková presnosť sa príliš nemení, avšak v tomto prípade nie je možné určiť, nakoľko je ovplyvnená väčšinou skupinou, keďže model sa vo všetkých prípadoch aspoň z časti klasifikuje menšinovú skupinu.

Konečné výsledky otestovaného modelu je možné vidieť v 6.1.2. F-skóre, senzitivita aj pozitívna prediktívna hodnota klesajú s pomerom menšinovej skupiny v datasete, z čoho je možné usúdiť, že nevyrovnaný dataset ovplyvnil tréning modelu. Napriek tomu, model s implementáciou tejto metódy dosahuje výrazne lepšie výsledky než v predchádzajúcom experimente. Oproti vyrovnanému datasetu senzitivita klesla len o 1,5 % pri pomere 7:3. V poslednom experimente, kedy je obsah menšinovej skupiny len 10 %, senzitivita dosiahla 64 %. Táto hodnota nie je veľmi vysoká, a preto je možné usúdiť, že model mohol klasifikovať tieto vzorky aj náhodne. F-skóre aj pozitívna prediktívna hodnota sú v tomto prípade tiež nízke. Výsledné hodnoty v poslednom experimente sú len orientačné, vzhľadom na to, že model sa preučil.



Obr. 6.2: Grafické zobrazenie porovnania výsledkov pri tréningu s váženou vzájomnou entropiou. Výsledky sú uvedené v pomere väčšinovej skupiny k menšinovej.

Tab. 6.2: Výsledky otestovaného modelu natrénovaného s implementáciou vázenej vzájomnej entropie.

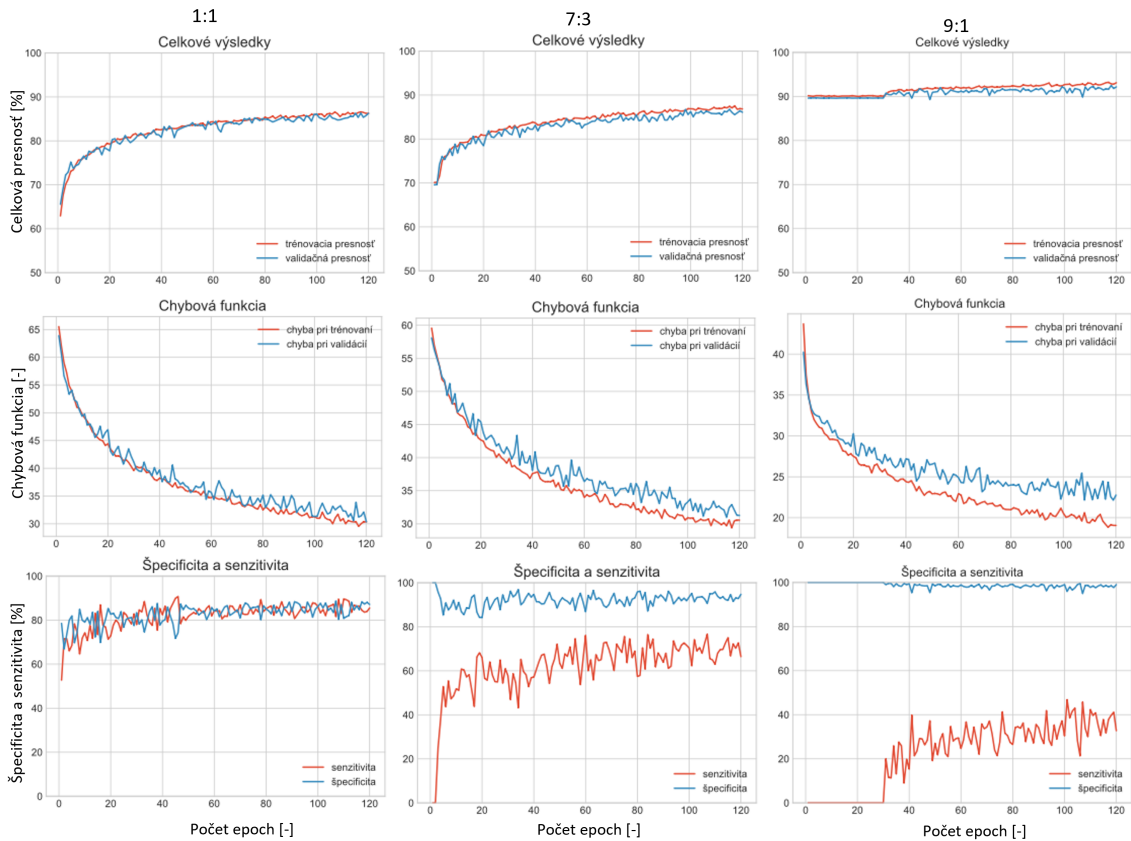
pomer väčšina:menšina	1:1	7:3	9:1
presnosť [%]	86,4	85,4	88,8
F-skóre [%]	85,9	77,1	53,8
pozitívna prediktívna hodnota [%]	88,9	73,7	47,6
senzitivita [%]	83,5	82,0	64,3
špecifita [%]	89,7	87,4	91,8

6.1.3 Učenie s dátovou augmentáciou

Pri pomere 1:1 je možné vidieť, že model sa nepreučoval a naučil sa klasifikovať obe skupiny s presnosťou 90,61 % pre mačky a 86,81 % pre psy.

V ďalšom experimente pri pomere 7:3 je možné vidieť rozdiel v presnosti klasifikácie jednotlivých skupín. Model dosahuje väčšiu presnosť klasifikácie väčšinovej skupiny (95 %), než menšinovej skupiny (67,84 %). Senzitivita stúpala s pridávaním augmentovaných fotografií počas epoch. Pokiaľ by experiment nebol obmedzený na 120 epoch, je možné, že klasifikácia menšinovej skupiny by mohla dosiahnuť rovnakú úspešnosť ako klasifikácia väčšinovej skupiny.

V poslednom experimente s pomerom 9:1 väčšinovej skupiny k menšinovej, je možné vidieť v krivkách celkovej presnosti a krivkách špecificity a senzitivity, že model sa menšinovú skupinu najskôr neučil a všetky vzorky označil za patriace väčšinovej skupine. Neskôr, s dostatočnou augmentáciou začal niektoré vzorky označovať aj ako menšinovú skupinu, avšak vzhľadom na to, že senzitivita nedosiahla vyššie hodnoty, pravdepodobne sa jednalo o náhodnú klasifikáciu.



Obr. 6.3: Grafické zobrazenie porovnania výsledkov pri tréovaní s dátovou augmentáciou. Výsledky sú uvedené v pomere väčšinovej skupiny k menšinovej.

Tab. 6.3: Výsledky otestovaného modelu, natrénovaného s implementáciou dátovej augmentácie.

pomer väčšina:menšina	1:1	7:3	9:1
presnosť [%]	88,6	87,1	92,2
F-skóre [%]	88,3	75,8	44,2
pozitívna prediktívna hodnota [%]	90,2	87,1	77,2
senzitivita [%]	86,8	67,8	32,5
špecificita [%]	90,6	95,7	99,0

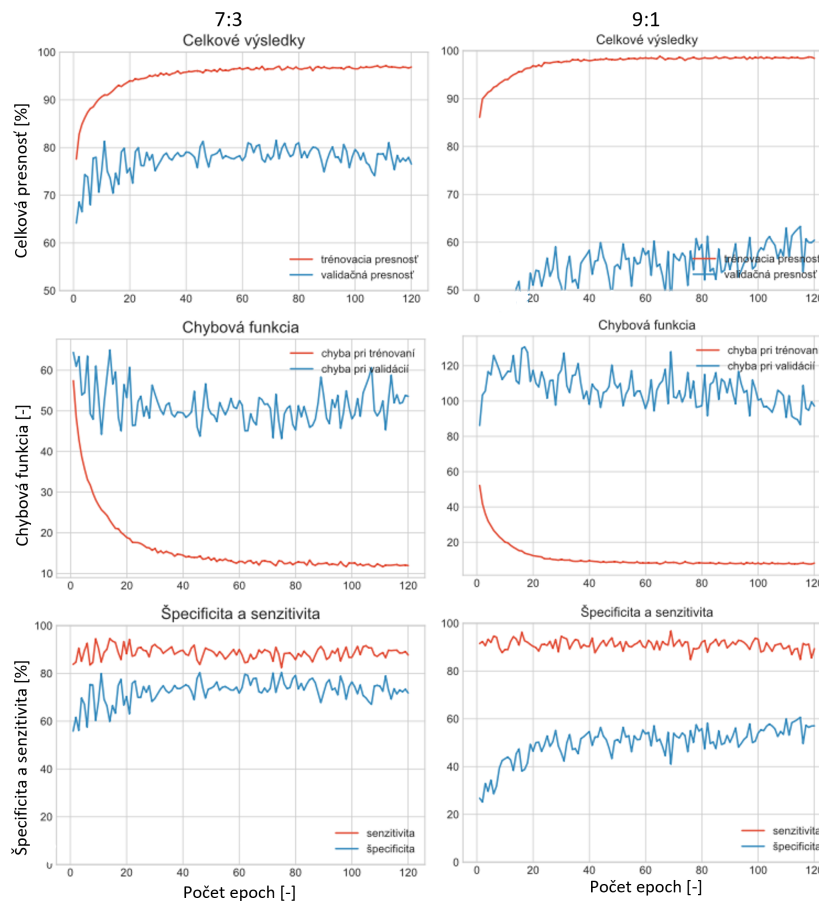
6.1.4 Učenie so SMOTE

Rozšírenie datasetu pomocou metódy SMOTE nebolo realizované pre vyrovnaný dataset, keďže v tomto prípade nie je potrebné generovať syntetické vzorky menšinovej skupiny.

Pri nevyrovnanosti 7:3 dosahuje celková presnosť okolo 80 %. Chybová funkcia neklesá tak, ako by po správnosti mala. Je možné vidieť, že validačná chyba osciluje a klesá len mierne v prvých epochách. V posledných epochách zas mierne stúpa, z čoho je možné usúdiť, že model sa ku koncu mohol začať preučovať. Senzitivita dosahuje okolo 90 % a špecificita okolo 75–80 %. Avšak vzhľadom na chybovú funkciu, je náročné posúdiť dôveryhodnosť týchto výsledkov. Trénovanie modelu s týmto algoritmom by vyžadovalo obsirnejšiu analýzu.

Pri nevyrovnanosti 9:1 sa celková presnosť pohybovala okolo 50 %, na otestovaných dátach dosiahla 59,85 %, z čoho je možné usúdiť, že model klasifikoval jednotlivé vzorky náhodne. To, že chybová funkcia stúpa už na začiatku značí, že model sa neučil správne. Rast chybovej funkcie môže byť spôsobený tým, že dataset, na ktorom bol model trénovaný, teda dataset vygenerovaný algoritmom SMOTE, bol príliš odlišný od datasetu, na ktorom bol model validovaný. SMOTE teda vygeneroval snímky príliš odlišné od skutočných príkladov psov a mačiek, čo je možné vidieť aj na ukážke vygenerovaného datasetu v predchádzajúcej kapitole 5.5.

Konečné výsledky uvedené v tabuľke 6.1.4 ukazujú, že model v oboch prípadoch dosiahol pomerne nízku presnosť v klasifikácii, čo ukazuje F-skóre a taktiež aj pozitívna prediktívna hodnota. Pri nevyrovnanosti 9:1 je model nepoužiteľný, vzhľadom na to, že hodnoty pozitívnej prediktívnej hodnoty, F-skóre, špecificity a aj celkovej validačnej presnosti sú príliš nízke.



Obr. 6.4: Grafické zobrazenie porovnania výsledkov pri tréningu s metódou SMOTE. Výsledky sú uvedené v pomere väčšinovej skupiny k menšinovej.

Tab. 6.4: Výsledky otestovaného modelu, natrénovaného s implementáciou algoritmu SMOTE.

pomer väčšina:menšina	1:1	7:3	9:1
presnosť [%]	–	76,0	59,9
F-skóre [%]	–	68,6	31,1
pozitívna prediktívna hodnota [%]	–	56,9	19,1
senzitivita [%]	–	87,4	88,5
špecifita [%]	–	71,2	56,6

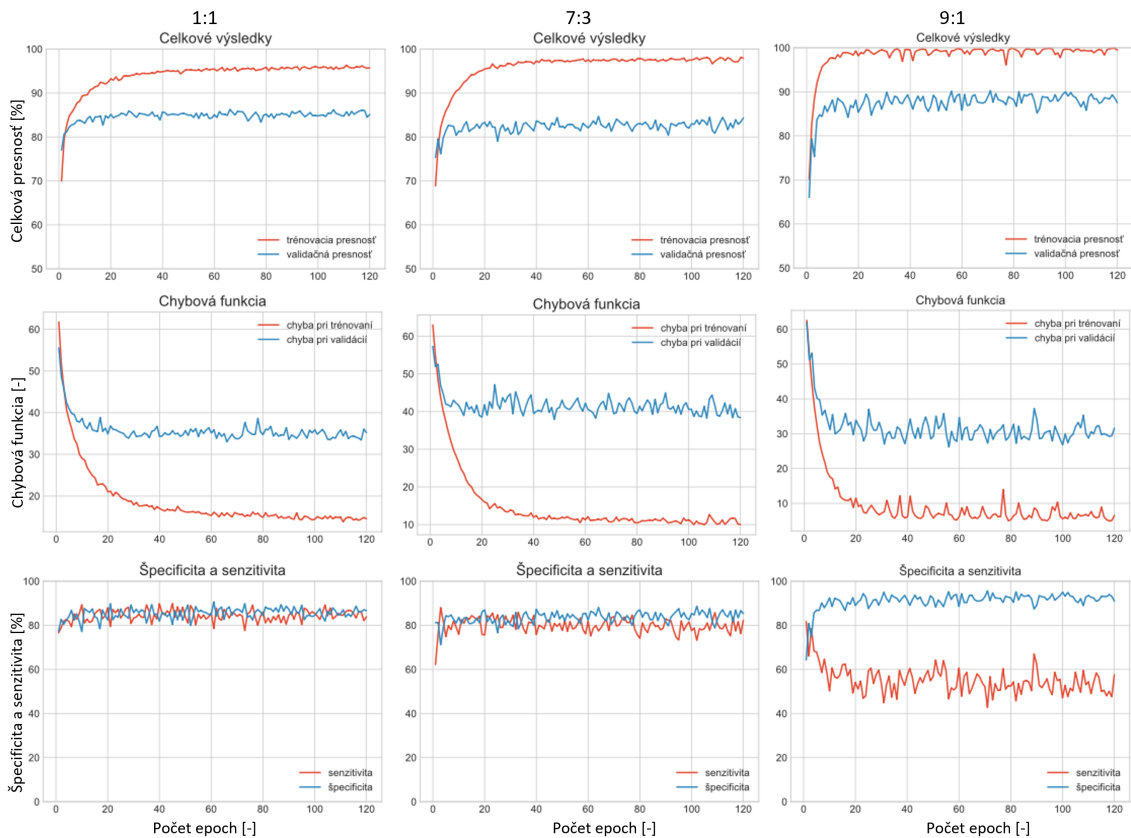
6.1.5 Učenie s váženým náhodným prevzorkovaním

Z grafických výsledkov na obr. 6.5 je možné vidieť, že pri vyrovnanom datasete sa model správne naučil obe skupiny, čo je možné pozorovať z kriviek špecifity

a senzitivity. Z chybových funkcií vo všetkých experimentoch je možné vidieť, že model sa nepreučoval ani v jednom prípade.

Pri nevyrovnanosti 7:3 je z kriviek špecificity a senzitivity možné pozorovať, že model dosiahol takmer rovnako dobré výsledky ako pri vyrovnanom datasete. Avšak pri datasete v pomere 9:1 väčšinovej skupiny k menšinovej už metóda nedosiahla tak dobré výsledky a je možné pozorovať zreteľný rozdiel v klasifikácii jednotlivých skupín. Senzitivita v tomto prípade dosahuje hodnoty len okolo 60 % a postupne klesá. Preto by bolo vhodné včasnejšie prerušenie tréningu modelu.

Konečné výsledky otestovaného modelu uvedené v tabuľke 6.1.5 ukazujú, že presnosť klasifikácie jednotlivých skupín pri nevyrovnanosti 7:3 klesla o 2–3 % oproti vyrovnanému datasetu. To potvrdzuje, že model sa úspešne naučil obe skupiny. Výraznejšie klesla pozitívna prediktívna hodnota a F-skóre. Pri nevyrovnanosti 9:1 už senzitivita dosiahla len 57 %, čo je možné označiť za náhodnú klasifikáciu. F-skóre a pozitívna prediktívna hodnota sú taktiež nízke.



Obr. 6.5: Grafické zobrazenie porovnania výsledkov pri tréningu s náhodným pre-
vzorkovaním. Výsledky sú uvedené v pomere väčšinovej skupiny k menšinovej.

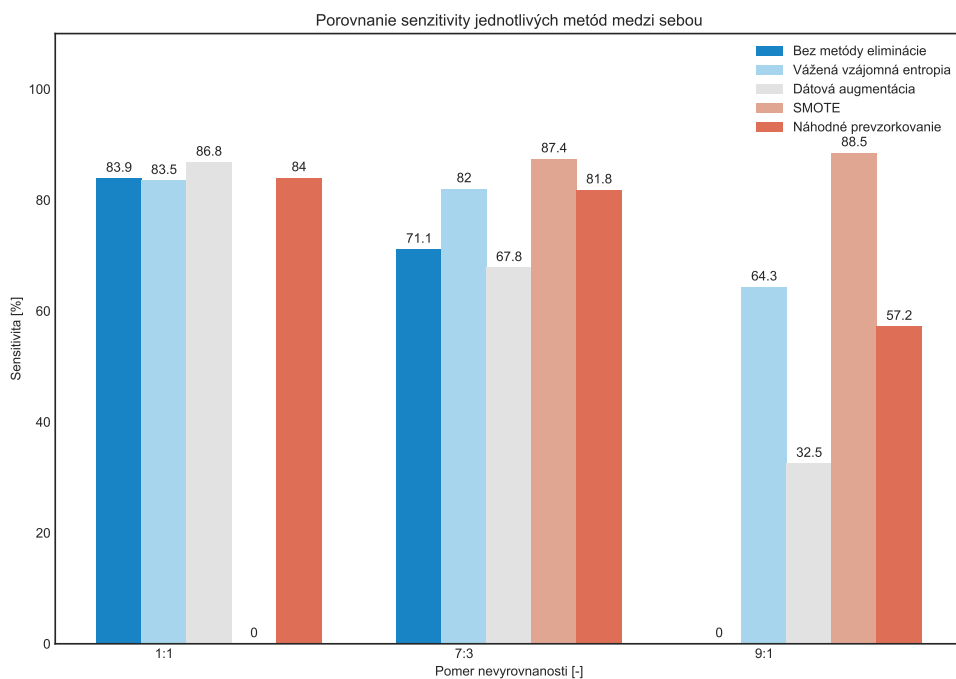
Tab. 6.5: Výsledky otestovaného modelu natrénovaného s metódou náhodného prevzorkovania.

pomer väčšina:menšina	1:1	7:3	9:1
presnosť [%]	85,3	83,8	87,6
F-skóre [%]	85,1	75,2	48,3
pozitívna prediktívna hodnota [%]	86,4	70,1	42,9
senzitivita [%]	84,0	81,8	57,2
špecificita [%]	86,9	84,8	91,2

6.1.6 Celkové porovnanie

Senzitivitu je možné označiť ako dobrý ukazovateľ úspešnosti metódy, preto boli prehľadne vykreslené výsledné hodnoty senzitivity pre jednotlivé experimenty v obrázku 6.6. Pre vyrovnaný dataset dosahovali všetky metódy približne rovnaké výsledky s výnimkou dátovej augmentácie, ktorá dosiahla o niečo lepšie výsledky senzitivity pri vyrovnanom datasete.

Pri nevyrovnanosti v pomere 7:3 sa už výsledky líšili. Najvyššiu senzitivitu dosiahol SMOTE, avšak dôveryhodnosť výsledkov tohto modelu je pochybná. Najlepšie výsledky teda dosiahla vážená vzájomná entropia a náhodné prevzorkovanie. Pri nevyrovnanosti 9:1 dosiahla najlepšie výsledky vážená vzájomná entropia. Avšak vzhľadom na to, že hodnota senzitivity je v tomto prípade 64 %, je možné usúdiť, že klasifikácia menšinovej skupiny bola takmer náhodná. V tejto nevyrovnanosti ale metódy celkovo nedosahujú vysokú úspešnosť.



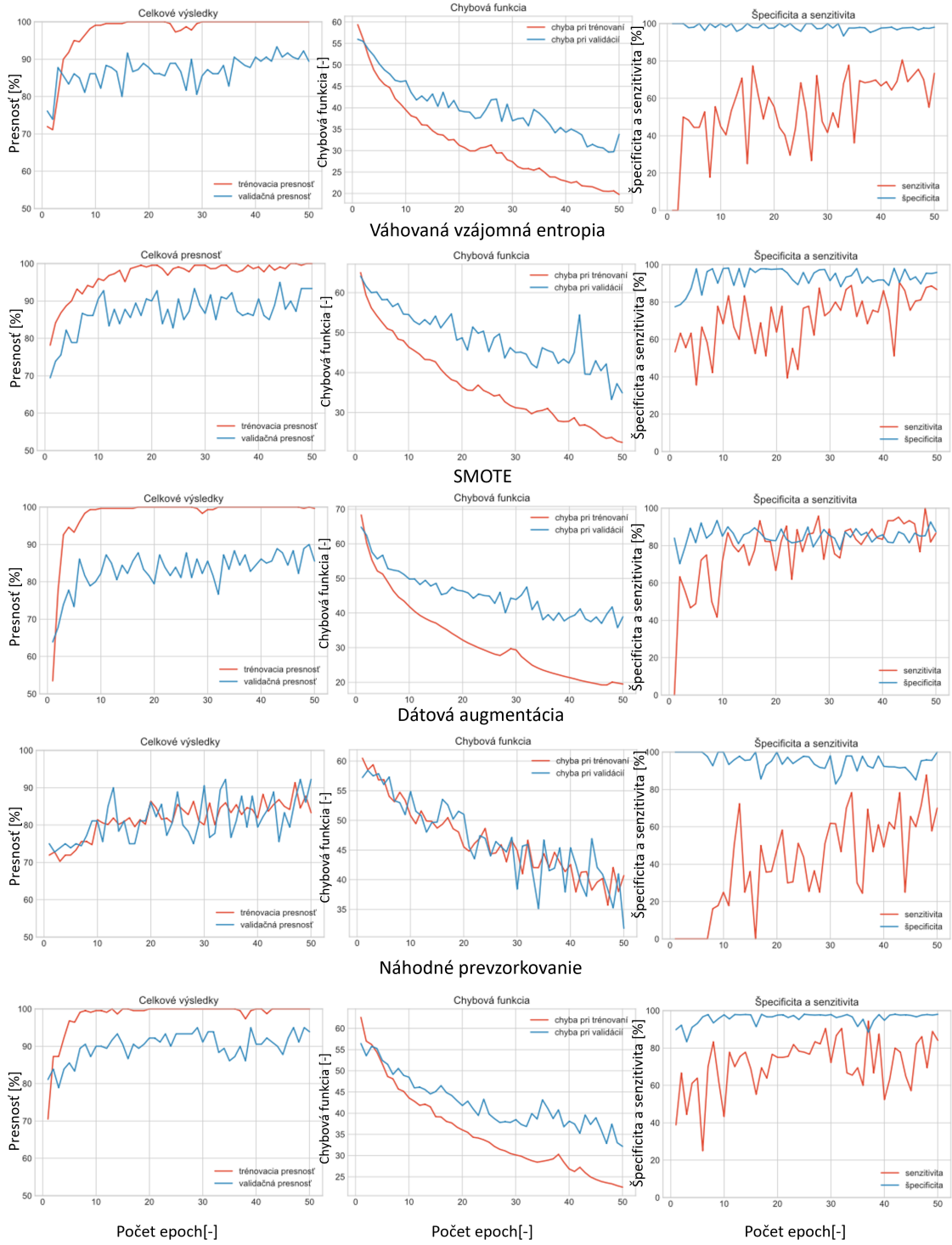
Obr. 6.6: Grafické porovnanie všetkých metód slúžiacich pre elimináciu vplyvu nevyrovnaného datasetu.

6.2 Učenie s datasetom skoliózy a zdravej chrbtice

Pre lepšie porovnanie priebehu a výsledkov jednotlivých metód boli výsledky vykreslené vedľa seba na obrázku 6.7. Z kriviek celkovej presnosti a chybovej funkcie je možné vidieť, že model sa vo všetkých prípadoch učil a bol regularizovaný, teda nepreučoval sa.

Pri zhodnotení kriviek špecificity a senzitivity je možné vidieť, že model sa vo všetkých prípadoch aspoň čiastočne naučil rozoznávať menšinovú skupinu. V prípade modelu, kde nie je implementovaná žiadna metóda, je možné vidieť rozdiel medzi krivkami senzitivity a špecificity. Skupinu pacientov so skoliózou sa model naučil rozoznávať takmer na 100 %, zatiaľ čo skupinu zdravých pacientov sa naučil na 64 %. Túto hodnotu je možné vidieť v konečných výsledkoch otestovaného modelu v tabuľke 6.2. Najúspešnejšiu klasifikáciu dosiahol pri implementácii váženej vzájomnej entropie s hodnotou senzitivity 87,8 %. Úspešné boli aj metódy SMOTE a náhodného prevzorkovania, model sa vďaka nim naučil rozoznávať menšinovú skupinu na 86 %. Pri použití dátovej augmentácie sieť niekoľko prvých epoch označovala všetky vzorky ako väčšinovú skupinu. Napriek tomu, v posledných epochách senzitivita vystúpila nad 80 % a v konečnom testovaní model dosiahol úspešnosť klasifikácie 81 %.

Bez metódy eliminácie nevyrovnaného datasetu



Obr. 6.7: Grafické porovnanie všetkých metód použitých na dataset skoliózy, spondylolistézy a zdravej chrbtice.

Tab. 6.6: Výsledky otestovaného modelu pre dataset skoliózy a zdravej chrbtice.

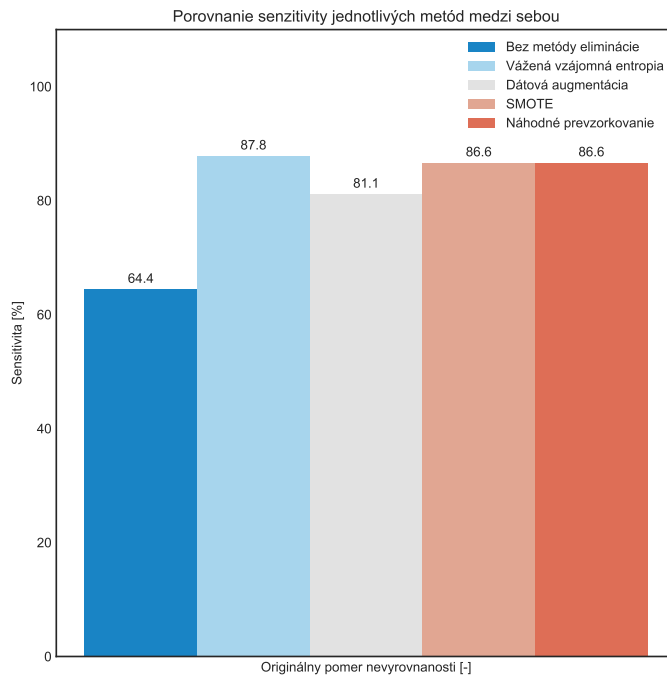
Dataset skoliózy a zdravej chrbtice	1.	2.	3.	4.	5.
presnosť [%]	92,2	95,0	95,0	84,4	94,9
F-skóre [%]	77,5	90,3	88,6	72,2	88,9
pozitívna prediktívna hodnota [%]	99,9	93,3	99,9	64,3	91,7
senzivitita [%]	64,4	87,8	81,1	86,7	86,7
špecificita [%]	99,9	97,8	99,9	84,4	97,8

Kde čísla stĺpcov predstavujú:

1. experiment bez metódy eliminácie,
2. experiment s váženou vzájomnou entropiou,
3. experiment s dátovou augmentáciou,
4. experiment s algoritmom SMOTE,
5. experiment s náhodným prevzorkovaním.

6.2.1 Porovnanie pre dataset skoliózy a zdravej chrbtice

Dataset obsahoval nevyrovnanosť 72 % väčšinovej skupiny a 28 % menšinovej skupiny. Model dosiahol pomerne dobré výsledky vo všetkých experimentoch, čo je možné odôvodniť tým, že obe skupiny boli zastúpené reprezentatívnymi vzorkami s odlišnými charakteristikami. Boli voči sebe exkluzívne a pravdepodobne teda ľahké na klasifikáciu. Prehľadné grafické porovnanie výsledkov senzitivity je možné vidieť na obrázku 6.8.



Obr. 6.8: Grafické porovnanie všetkých metód použitých na dataset skoliózy a zdravej chrbtice.

6.3 Diskusia

Na základe realizovaných experimentov je možné usúdiť, že náhodné prevzorkovanie spolu s váženou vzájomnou entropiou dosiahli najlepšie výsledky. Podobné tvrdenia je možné nájsť aj v literatúre, napríklad v článku [43], kde boli porovnávané metódy náhodného prevzorkovania, váženej vzájomnej entropie, SMOTE a ADASYN („Adaptive Synthetic“). Pričom ADASYN je metóda podobná algoritmu SMOTE, avšak generovanie syntetických vzoriek je viac zamerané na príklady, ktoré sú náročné pre učenie (odlahlé hodnoty). Autor v závere usúdil, že v experimentoch fungovalo najlepšie náhodné prevzorkovanie, za ním išli experimenty SMOTE a ADASYN.

V ďalších zdrojoch [21, 22] sú odporúčané metódy založené na prevzorkovaní a odporúča sa taktiež ich rozšírenie na hybridné metódy. Prípadne sa odporúča pristupovať k nevyrovnanému datasetu inými oblasťami strojového učenia ako napríklad náhodný les („Random Forrest“) alebo metóda podporných vektorov („Support Vector Machines“).

Ďalej je potrebné vyzdvihnúť faktory ovplyvňujúce pôsobenie nevyrovnaného datasetu na učenie siete. Dataset skoliózy a zdravej chrbtice obsahoval podobnú nevyrovnanosť ako dataset Microsoft ASIRRA v experimente s pomerom 7:3. Napriek tomu je možné pozorovať, že pri datasete skoliózy a zdravej chrbtice boli dosiahnuté

lepšie výsledky. Odlišné výsledky sú pravdepodobne spôsobené tým, že dataset Microsoft ASIRRA mohol byť náročnejší na klasifikáciu. Obsahoval veľa rôznych druhov mačiek aj psov, čo značí vysokú variabilitu. Model môže mať problém s nachádzaním dôležitých charakteristík. Experimenty potvrdili, že komplexita dát spolu s veľkosťou datasetu ovplyvňujú úspešnosť klasifikácie nevyrovnaného datasetu.

Taktiež je potrebné poznamenať, že vzhľadom na to, že dataset bol rozdelený jednorázovo na tréningovú a validačnú množinu a z výpočtového hľadiska neboli zavedené algoritmy ako „K-Fold Cross Validation“. Je možné, že výsledky jednotlivých experimentov sú ovplyvnené náhodnosťou, s ktorou boli rozdelené do jednotlivých datasetov.

Záver

V bakalárskej práci bola naštudovaná oblasť strojového učenia so zameraním na neurónové siete. Teória zahŕňala matematickú definíciu neurónu, prehľad používaných aktivačných funkcií a vysvetlenie učenia neurónovej siete pomocou úpravy váh a spätného šírenia chyby sieťou. Následne boli zhrnuté problémy vyskytujúce sa pri učení neurónovej siete. V práci je obsiahnutá aj teória konvolučných neurónových sietí a ich optimalizačné algoritmy, ktoré sú neskôr použité v praktickej časti.

Práca je zameraná na elimináciu nevyrovnaného datasetu, preto bol tento problém v práci bližšie definovaný a bola vypracovaná rešerš dostupných metód eliminácie tohto problému. Metódy je možné rozdeliť do dvoch skupín. Prvou sú metódy na úrovni algoritmu, ktoré spočívajú najmä v modifikácií chybovej funkcie. Druhou skupinou sú metódy na úrovni datasetu, teda modifikujú dataset potrebným spôsobom tak, aby bol eliminovaný vplyv problému nevyrovnaného datasetu.

Na základe naštudovaných metód, bolo navrhnutých 5 experimentov, slúžiacich na porovnanie úspešnosti jednotlivých metód v odstránení vplyvu nevyrovnaného datasetu. Prvý experiment neobsahoval žiadnu metódu eliminácie. V ďalších experimentoch bola implementovaná jedna metóda na úrovni algoritmu, teda vážená vzájomná entropia. A tri metódy na úrovni datasetu, kam patrí algoritmus SMOTE, dátová augmentácia a náhodné prevzorkovanie. Práca sa zaoberala úspešnosťou metód pri binárnej klasifikácii, preto boli použité datasety obsahujúce dve skupiny. Prvým datasetom bol Microsoft ASIRRA [35] obsahujúci fotografie psov a mačiek. Pre tento dataset boli experimenty vyhodnotené pre rôzne nevyrovnanosti. Učenie bolo realizované v pomeroch skupín 1:1, 7:3 a 9:1. Druhým datasetom je Dataset skoliózy, spondylolistézy a zdravej chrbtice [37], z ktorého bola použitá len skupina pacientov so skoliózou a skupina pacientov so zdravou chrbticou. Dataset bol pre experimenty použitý s originálnou nevyrovnanosťou.

Pre správne vyhodnotenie úspešnosti metód, boli použité datasety rozdelené na tréningovú a validačnú množinu, ktorá bola konštantná pre všetky experimenty. Výsledky boli hodnotené štandardnými metrikami vyvedenými z matice zámien, kam spadá špecificita, senzitivita, presnosť, F-skóre a pozitívna prediktívna hodnota. Pri použití nevyrovnaného datasetu pre učenie neurónovej siete má model problém sa naučiť menšinovú skupinu. Preto bola senzitivita zvolená ako hlavný ukazovateľ toho, úspešnosti metód a na základe nej boli porovnávané jednotlivé medzi sebou. Výsledky ukázali, že najväčšia úspešnosť bola dosiahnutá implementáciou váženej vzájomnej entropie a náhodného prevzorkovania.

Literatúra

1. BEŇUŠKOVÁ, Lubica. *Umelá inteligencia: Umelé neurónové siete*. 2. vyd. Bratislava: Slovenská technická univerzita, 2007. ISBN 978-80-227-2629-0.
2. TUČKOVÁ, Jana. *Vybrané aplikace umělých neuronových sítí při zpracování signálů*. Praha: České vysoké učení technické v Praze, 2009. ISBN 978-80-01-04229-8.
3. GOODFELLOW, Ian; BENGIO, Yoshua; COURVILLE, Aaron. *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016. <http://www.deeplearningbook.org>.
4. BRIDLE, John S. Training Stochastic Model Recognition Algorithms as Networks Can Lead to Maximum Mutual Information Estimation of Parameters. In: *Proceedings of the 2nd International Conference on Neural Information Processing Systems*. Cambridge, MA, USA: MIT Press, 1989, s. 211–217.
5. MASTERS, Timothy. *Signal and Image Processing with Neural Networks: C++ Sourcebook*. 1. vyd. New Jersey, USA: Wiley, 1994. ISBN 0-471-04963-8.
6. *What are neural networks?* IBM, 2021. Dostupné tiež z: <https://www.ibm.com/topics/neural-networks>.
7. KOZUMPLÍK, Jiří; PROVAZNÍK, Ivo. *Umelá inteligencia v medicíně*. Brno: FEKT VUT, 2007.
8. ZHANG, Zhilu; SABUNCU, Mert. Generalized Cross Entropy Loss for Training Deep Neural Networks with Noisy Labels. In: *Advances in Neural Information Processing Systems*. 2018. Dostupné tiež z: <https://dblp.org/rec/journals/corr/abs-1805-07836.bib>.
9. KINGMA, Diederik P.; BA, Jimmy. Adam: A Method for Stochastic Optimization. [B.r.]. Dostupné tiež z: <https://doi.org/10.48550/arXiv.1412.6980>.
10. BASODI, Sunitha; JI, Chunyan; ZHANG, Haiping; PAN, Yi. Gradient amplification: An efficient way to train deep neural networks. *Big Data Mining and Analytics*. 2020, roč. 3, č. 3, s. 196–207. Dostupné z DOI: 10.26599.
11. JIŘÍ HOLČÍK, Martin Komenda. *Matematická biologie: e-learningová učebnice*. 1. vyd. Brno: Masarykova univerzita, 2015. ISBN 978-80-210-8095-9. Dostupné tiež z: <https://portal.matematickabiologie.cz/>.
12. JAN, Jiří. *Medical image processing, reconstruction and analysis: Concepts and methods*. Second edition. London: CRC Press, 2020. ISBN 978-1-138-31028-5.

13. KOTHIYA, Aditi. *Understanding “convolution” operations in CNN* [online]. [cit. 2022-12-25]. Dostupné z : <https://medium.com/analytics-vidhya/convolution-operations-in-cnn-deep-learning-computer-vision-128906ece7d3>.
14. WALTERS, Austin G. *Convolutional Neural Networks (CNN) to Classify Sentences* [online]. [cit. 2022-12-25]. Dostupné z : <https://austingwalters.com/convolutional-neural-networks-cnn-to-classify-sentences/>.
15. HE, Kaiming; ZHANG, Xiangyu; REN, Shaoqing; SUN, Jian. Deep Residual Learning for Image Recognition. 2015. Dostupné z arXiv: 1512.03385.
16. RONNEBERGER, Olaf; FISCHER, Philipp; BROX, Thomas. *U-Net: Convolutional Networks for Biomedical Image Segmentation*. 2015. Dostupné z arXiv: 1505.04597 [cs.CV].
17. SRIVASTAVA, Nitish; HINTON, Geoffrey; KRIZHEVSKY, Alex; SUTSKEVER, Ilya; SALAKHUTDINOV, Ruslan. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research*. 2014, roč. 15, č. 56, s. 1929–1958. Dostupné tiež z: <http://jmlr.org/papers/v15/srivastava14a.html>.
18. IOFFE, Sergey; SZEGEDY, Christian. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. *CoRR*. 2015. Dostupné tiež z: <http://arxiv.org/abs/1502.03167>.
19. NAGPAL, Anuja. Towards Data Science: L1 and L2 Regularization Methods. In: [b.r.]. Dostupné tiež z: <https://towardsdatascience.com/l1-and-l2-regularization-methods-ce25e7fc831c>.
20. NG, Andrew Y. Feature selection, L 1 vs. L 2 regularization, and rotational invariance. In: *Twenty-first international conference on Machine learning - ICML '04*. New York, New York, USA: ACM Press, 2004, s. 78–. ISBN 1581138285. Dostupné z DOI: 10.1145/1015330.1015435.
21. KUMAR, Pradeep; BHATNAGAR, Roheet; GAUR, Kuntal; BHATNAGAR, Anurag. Classification of Imbalanced Data: Review of Methods and Applications. *IOP Conference Series: Materials Science and Engineering*. 2021. Dostupné z DOI: 10.1088/1757-899X/1099/1/012077.
22. JOHNSON, Justin M.; KHOSHGOFTAAR, Taghi M. Survey on deep learning with class imbalance. *Journal of Big Data*. 2019, roč. 6, č. 1. ISSN 2196-1115. Dostupné z DOI: 10.1186/s40537-019-0192-5.

23. KRAWCZYK, Bartosz. Learning from imbalanced data: Open challenges and future directions. *Progress in Artificial Intelligence*. 2016, roč. 5, č. 4, s. 221–232. ISSN 2192-6352. Dostupné z DOI: 10.1007/s13748-016-0094-0.
24. JAPKOWICZ, Nathalie. The Class Imbalance Problem: Significance and Strategies. In: 2000. Dostupné tiež z: <https://www.aaai.org/Papers/Workshops/2000/WS-00-05/WS00-05-003.pdf>.
25. SHARMA, Deepak Kumar; CHATTERJEE, Mayukh; KAUR, Gurmehak; VAVILALA, Suchitra. 3 - Deep learning applications for disease diagnosis. In: GUPTA, Deepak; KOSE, Utku; KHANNA, Ashish; BALAS, Valentina Emilia (ed.). *Deep Learning for Medical Applications with Unique Data*. Academic Press, 2022, s. 31–51. ISBN 978-0-12-824145-5. Dostupné z DOI: <https://doi.org/10.1016/B978-0-12-824145-5.00005-8>.
26. POUYANFAR, Samira; TAO, Yudong; MOHAN, Anup; TIAN, Haiman; KASEB, Ahmed; GAUEN, Kent; DAILEY, Ryan; AGHAJANZADEH, Sarah; LU, Yung-Hsiang; CHEN, Shu-Ching; SHYU, Mei-Ling. Dynamic Sampling in Convolutional Neural Networks for Imbalanced Data Classification. In: 2018. Dostupné z DOI: 10.1109/MIPR.2018.00027.
27. BOWYER, Kevin W.; CHAWLA, Nitesh V.; HALL, Lawrence O.; KEGELMEYER, W. Philip. SMOTE: Synthetic Minority Over-sampling Technique. 2011. Dostupné tiež z: <http://arxiv.org/abs/1106.1813>.
28. SUTIONO, Arie Pratama. *Towards Advanced Accommodation: Deep Learning for Photos Classification — Part 1* [online]. [cit. 2023-05-14]. Dostupné z : <https://medium.com/airy-science/towards-advanced-accommodation-photos-classification-part-1-a64b542b31ed>.
29. DONG, Qi; GONG, Shaogang; ZHU, Xiatian. Class Rectification Hard Mining for Imbalanced Deep Learning. 2017. Dostupné tiež z: <http://arxiv.org/abs/1712.03162>.
30. SRIVASTAVA, Yash; MURALI, Vaishnav; DUBEY, Shiv Ram. Hard-Mining Loss based Convolutional Neural Network for Face Recognition. 2019. Dostupné tiež z: <http://arxiv.org/abs/1908.09747>.
31. SHRIVASTAVA, Abhinav; GUPTA, Abhinav; GIRSHICK, Ross B. Training Region-based Object Detectors with Online Hard Example Mining. 2016. Dostupné tiež z: <http://arxiv.org/abs/1604.03540>.
32. IBRAHIM, Karim M; EPURE, Elena; PEETERS, Geoffroy; RICHARD, Gael. Confidence-based Weighted Loss for Multi-label Classification with Missing Labels. In: *The 2020 International Conference on Multimedia Retrieval (ICMR '20)*. Dublin, Ireland, 2020. Dostupné z DOI: 10.1145/3372278.3390728.

33. PHAN, Trong Huy; YAMAMOTO, Kazuma. Resolving Class Imbalance in Object Detection with Weighted Cross Entropy Losses. 2020. Dostupné tiež z: <https://arxiv.org/abs/2006.01413>.
34. LIN, Tsung-Yi; GOYAL, Priya; GIRSHICK, Ross; HE, Kaiming; DOLLÁR, Piotr. Focal Loss for Dense Object Detection. In: *IEEE International Conference on Computer Vision (ICCV)*. 2017. Dostupné z DOI: 10.1109/ICCV.2017.324.
35. ELSON, Jeremy; DOUCEUR, John (JD); HOWELL, Jon; SAUL, Jared. Asirra: A CAPTCHA that Exploits Interest-Aligned Manual Image Categorization. In: *Proceedings of 14th ACM Conference on Computer and Communications Security (CCS)*. Association for Computing Machinery, Inc., 2007. Dostupné tiež z: <https://www.microsoft.com/en-us/research/publication/asirra-a-captcha-that-exploits-interest-aligned-manual-image-categorization/>.
36. CUKIERSKI, Will. *Dogs vs. Cats Redux: Kernels Edition*. Kaggle, 2016. Dostupné tiež z: <https://kaggle.com/competitions/dogs-vs-cats-redux-kernels-edition>.
37. FRAIWAN, Mohammad; AUDAT, Ziad; MANASREH, Tareq. A dataset of scoliosis, spondylolisthesis, and normal vertebrae X-ray images [online]. [B.r.] [cit. 2023-05-13]. Dostupné z DOI: 10.17632/xkt857dsxk.1.
38. KRBEC, Martin. Neurologie pro praxi: Spondylolistéza – Chirurgické léčení [online]. 2002, s. 8–12 [cit. 2023-05-22]. Dostupné z: <https://www.neurologiepropraxi.cz/pdfs/neu/2002/01/03.pdf>.
39. SOSNA, Antonín. *Základy ortopedie*. Vyd. 1. Praha: Triton, 2001. ISBN 80-7254-202-8. Dostupné tiež z: <http://www.digitalniknihovna.cz/mzk/uuid/uuid:d5a73e20-d2f7-11e4-97af-005056827e51>.
40. PEDREGOSA, Fabian; VAROQUAUX, Gaël; GRAMFORT, Alexandre; MICHEL, Vincent; THIRION, Bertrand; GRISEL, Olivier; BLONDEL, Mathieu. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*. 2011, roč. 12, s. 2825–2830.
41. CHO, Luke. *Pytorch CNN tutorial with cats and dogs* [online]. [cit. 2023-05-17]. Dostupné z: <https://www.kaggle.com/code/reukki/pytorch-cnn-tutorial-with-cats-and-dogs/notebook>.
42. *Pytorch: Documentation* [online]. [cit. 2022-12-14]. Dostupné z: <https://pytorch.org/docs/stable/data.html>.

43. SINGHAL, Vani. *Comparative Analysis of Oversampling Techniques on Imbalanced Data* [online]. [cit. 2023-05-17]. Dostupné z : <https://towardsdatascience.com/comparative-analysis-of-oversampling-techniques-on-imbalanced-data-cd46f172d49d>.

Zoznam symbolov a skratiek

Adam	„Adaptive moment estimation“
ADASYN	„Adaptive synthetic“
FN	„False negative“
FP	„False positive“
PPV	„Positive predictive value“
ReLU	„Rectified linear unit“
ResNet	„Residual Network“
SE	„Sensitivity“
SMOTE	„Synthetic Minority Over-Sampling Technique“
SP	„Specificity“
TN	„True negative“
TNR	„True negative ratio“
TP	„True Positive“
TPR	„True positive ratio“
VE	Vzájomná entropia
VVE	Vážená vzájomná entropia