

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta elektrotechniky  
a komunikačních technologií

DIPLOMOVÁ PRÁCE

Brno, 2021

Bc. Karolína Šťáková



# 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

## IDENTIFIKACE ABNORMÁLNÍCH EKG SEGMENTŮ POMOCÍ METODY MULTIPLE-INSTANCE LEARNING

IDENTIFICATION OF ABNORMAL ECG SEGMENTS USING MULTIPLE-INSTANCE LEARNING

### DIPLOMOVÁ PRÁCE

MASTER'S THESIS

### AUTOR PRÁCE

AUTHOR

Bc. Karolína Šťávová

### VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Jakub Hejč

BRNO 2021



# Diplomová práce

magisterský navazující studijní program **Biomedicínské inženýrství a bioinformatika**

Ústav biomedicínského inženýrství

**Studentka:** Bc. Karolína Šťávovalá

**ID:** 174534

**Ročník:** 2

**Akademický rok:** 2020/21

## NÁZEV TÉMATU:

### Identifikace abnormálních EKG segmentů pomocí metody Multiple-Instance Learning

#### POKYNY PRO VYPRACOVÁNÍ:

1) Seznamte se s metodami hlubokého učení pro zpracování signálů. 2) Vytvořte literární rešerši se zaměřením na konvoluční neuronové sítě a principy multi-instanačního učení. 3) Prostudujte dostupná data. Navrhněte a implementujte pipe-line pro jejich předzpracování. 4) Vyberte a implementujte vhodný model pro segmentaci abnormálních úseků v elektrokardiografických záznamech. 5) Proveďte optimalizaci architektury a hyperparametrů modelu. 6) Proveďte vyhodnocení a diskuzi dosažených výsledků.

#### DOPORUČENÁ LITERATURA:

[1] Zhi-Hua Zhou, A brief introduction to weakly supervised learning, National Science Review, Volume 5, Issue 1, January 2018, Pages 44–53, <https://doi.org/10.1093/nsr/nwx106>

[2] Kanavati, F., Toyokawa, G., Momosaki, S. et al. Weakly-supervised learning for lung carcinoma classification using deep learning. Sci Rep 10, 9297 (2020), <https://doi.org/10.1038/s41598-020-66333-x>.

[3] L. Sun, Y. Lu, K. Yang and S. Li, "ECG Analysis Using Multiple Instance Learning for Myocardial Infarction Detection," in IEEE Transactions on Biomedical Engineering, vol. 59, no. 12, pp. 3348-3356, Dec. 2012, doi: 10.1109/TBME.2012.2213597.

**Termín zadání:** 8.2.2021

**Termín odevzdání:** 21.5.2021

**Vedoucí práce:** Ing. Jakub Hejč

**prof. Ing. Ivo Provazník, Ph.D.**  
předseda rady studijního programu

#### UPOZORNĚNÍ:

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

## **Abstrakt**

Srdeční arytmie jsou velmi častým srdečním onemocněním, jejichž výskyt v populaci se zvyšuje. Tato práce se zabývá automatickou detekcí komorových extrasystol ze záznamů 12svodového EKG, k čemuž využívá hlubokého učení. Prostřednictvím techniky založené na přístupu Multiple-Instance Learning je dále provedena lokalizace těchto arytmií (klíčových instancí) v záznamu. V teoretické části práce je popsána základní elektrofyziologie srdce a princip fungování hlubokého učení zejména se zaměřením na konvoluční neuronové sítě. V rámci programového řešení v jazyce Python byl využit model postavený na architektuře InceptionTime, pomocí něhož byla provedena klasifikace signálů do vybraných tříd. Pro zjištění lokalit klíčových instancí v EKG byla implementována metoda Grad-CAM. Hodnocení kvality detekce arytmií proběhlo prostřednictvím výpočtu F1 skóre a výsledky byly na konci práce diskutovány.

## **Klíčová slova**

Hluboké učení, konvoluční neuronové sítě, Multiple-Instance Learning, Grad-CAM, arytmie, elektrokardiografie (EKG), Python

## **Abstract**

Heart arrhythmias are a very common heart disease whose incidence is rising. This thesis is focused on the detection of premature ventricular contractions from 12-lead ECG records by means of deep learning. The location of these arrhythmias (key instances) in the record was found using a technique based on Multiple-Instance Learning. In the theoretical part of the thesis, basic electrophysiology of the heart and deep learning with a focus on the convolutional neural networks are described. Afterward, a program was created using the Python programming language, which contains a model based on the InceptionTime architecture, using which classification of the signals into the selected classes was performed. Grad-CAM was implemented to find locations of the key instances in the ECGs. The evaluation of the arrhythmia detection quality was done using the F1 score and the results were discussed at the end of the thesis.

## **Keywords**

Deep learning, convolutional neural networks, Multiple-Instance Learning, Grad-CAM, arrhythmias, electrocardiography (ECG), Python

## **Bibliografická citace**

ŠŤÁVOVÁ, Karolína. *Identifikace abnormálních EKG segmentů pomocí metody Multiple-Instance Learning* [online]. Brno, 2021 [cit. 2021-05-21]. Dostupné z: <https://www.vutbr.cz/studenti/zav-prace/detail/134973>. Diplomová práce. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav biomedicínského inženýrství. Vedoucí práce Jakub Hejč.

## Prohlášení autora o původnosti díla

<b>Jméno a příjmení studenta:</b>	Karolína Šťávová
<b>VUT ID studenta:</b>	174534
<b>Typ práce:</b>	Diplomová práce
<b>Akademický rok:</b>	2020/21
<b>Téma závěrečné práce:</b>	Identifikace abnormálních EKG segmentů pomocí metody Multiple-Instance Learning

Prohlašuji, že svou závěrečnou práci jsem vypracovala samostatně pod vedením vedoucího závěrečné práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené závěrečné práce dále prohlašuji, že v souvislosti s vytvořením této závěrečné práce jsem neporušila autorská práva třetích osob, zejména jsem nezasáhla nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědoma 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.

V Brně dne: 21. května 2021

-----  
podpis autora

## **Poděkování**

Děkuji vedoucímu diplomové práce Ing. Jakubu Hejčovi za účinnou metodickou, pedagogickou a odbornou pomoc a další cenné rady při zpracování mé diplomové práce.

V Brně dne: 21. května 2021

-----  
podpis autora

# Obsah

<b>SEZNAM ZKRATEK .....</b>	<b>9</b>
<b>SEZNAM OBRÁZKŮ .....</b>	<b>10</b>
<b>SEZNAM TABULEK.....</b>	<b>11</b>
<b>ÚVOD.....</b>	<b>12</b>
<b>1. ELEKTROFYZIOLOGIE SRDCE .....</b>	<b>13</b>
1.1    Převodní systém srdeční .....	13
1.2    Srdeční arytmie .....	14
1.3    EKG .....	17
<b>2. NEURONOVÉ SÍTĚ .....</b>	<b>19</b>
2.1    Úvod ke strojovému učení .....	19
2.2    Úvod k neuronovým sítím .....	20
2.3    Hluboké učení .....	22
2.4    Konvoluční neuronové síť .....	22
2.5    Ztrátová funkce .....	27
2.6    Učební algoritmy .....	28
2.7    Problémy konvolučních neuronových sítí .....	29
2.8    Hyperparametry sítě.....	30
2.9    Neuronová síť Inception .....	32
2.10   Multiple-Instance Learning.....	33
2.11   Class Activation Mapping .....	36
2.12   Metriky pro zhodnocení kvality modelu.....	37
<b>3. NAVRŽENÝ PROGRAM.....</b>	<b>39</b>
3.1    Data .....	39
3.2    Předzpracování dat.....	40
3.3    Řešení nevyvážených dat a jejich augmentace .....	46
3.4    Učení modelu .....	47
3.5    Výběr architektury .....	48
3.6    Hyperparametry a jejich optimalizace .....	52
3.7    Lokalizace PVC .....	53
<b>4. VÝSLEDKY .....</b>	<b>54</b>
4.1    Bayesovská optimalizace hyperparametrů.....	54
4.2    Klasifikace .....	56
4.3    Grad-CAM a lokalizace PVC .....	63



<b>5. DISKUZE .....</b>	<b>70</b>
5.1 Bayesovská optimalizace hyperparametrů.....	70
5.2 Klasifikace .....	71
5.3 Lokalizace PVC .....	74
<b>6. ZÁVĚR .....</b>	<b>75</b>
<b>LITERATURA.....</b>	<b>76</b>

## SEZNAM ZKRATEK

AV	Atrioventrikulární
AVNRT	AV nodální reentry tachykardie
AVRT	AV reentry tachykardie
CAM	Class Activation Mapping
CNN	Konvoluční neuronová síť (Convolutional Neural Network)
EKG	Elektrokardiografie
FCN	Plně propojená vrstva (Fully Connected Layer)
FEKT	Fakulta elektrotechniky a komunikačních technologií
FN	Falešně negatívni (False Negative)
FP	Falešně pozitivní (False Positive)
GAP	Globální průměrovací pooling (Global Average Pooling)
GPU	Grafický procesor (Graphics Processing Unit)
Grad-CAM	Gradientem vážený Class Activation Mapping (Gradient-weighted Class Activation Mapping)
LBBB	Blokáda levého Tawarova raménka (Left Bundle Branch Block)
LSTM	Long Short-Term Memory
MIL	Multiple-Instance Learning
PAC	Síňová extrasystola (Premature Atrial Contraction)
PVC	Komorová extrasystola (Premature Ventricular Contraction)
RBBB	Blokáda pravého Tawarova raménka (Right Bundle Branch Block)
ReLU	Rektifikovaná lineární jednotka (Rectified Linear Unit)
SMOTE	Syntetická technika minoritního vzorkování (Synthetic Minority Oversampling TEchnique)
TN	Pravdivě negativní (True Negative)
TP	Pravdivě pozitivní (True Positive)
VUT	Vysoké učení technické v Brně
WPW	Wolff-Parkinson-Whiteův syndrom

# SEZNAM OBRÁZKŮ

Obr. 1.1: Převodní systém srdeční [3].....	13
Obr. 1.2: EKG – komorová extrasystola [8] .....	16
Obr. 1.3: Křivka EKG [10] .....	17
Obr. 1.4: Umístění hrudních svodů [11] .....	18
Obr. 2.1: Schéma perceptronu, převzato z [20].....	21
Obr. 2.2: Vícevrstvá neuronová síť, převzato z [20].....	21
Obr. 2.3: Schéma konvoluční vrstvy, převzato z [17].....	24
Obr. 2.4: Aktivační funkce ReLU .....	25
Obr. 2.5: Příklad použití pooling vrstev .....	25
Obr. 2.6: Schéma plně propojené vrstvy .....	26
Obr. 2.7: Vlevo: Dropout s $p = 0$ , vpravo Dropout s $p = 0,5$ , převzato z [21] .....	27
Obr. 2.8: Sigmoidální funkce a její derivace.....	29
Obr. 2.9: Porovnání velkého/malého učebního kroku, převzato z [22] .....	31
Obr. 2.10: Inception modul s redukcí dimenzí, převzato z [32].....	33
Obr. 2.11: Schéma přístupu založeném na instancích [33] .....	34
Obr. 2.12: Schéma přístupu založeném na vnoření [33] .....	35
Obr. 2.13: Schéma architektury umožňující CAM, převzato z [38] .....	36
Obr. 2.14: Teplotní mapa, pneumotorax [39] .....	36
Obr. 2.15: Matice záměn.....	37
Obr. 3.1: Signál (jeden svod) před decimací.....	41
Obr. 3.2: Signál (jeden svod) po decimaci.....	41
Obr. 3.3: Frekvenční charakteristika Butterworthova filtru (horní propust) .....	42
Obr. 3.4: Frekvenční charakteristika Butterworthova filtru (pásmová zádrž) .....	43
Obr. 3.5: Signál (jeden svod) po filtraci.....	43
Obr. 3.6: Signály (12 svodů) před filtrací .....	44
Obr. 3.7: Signály (12 svodů) po filtraci .....	44
Obr. 3.8: Vývoj hodnoty učebního kroku za 1 cyklus .....	48
Obr. 3.9: Schéma architektury InceptionTime, převzato z [55].....	51
Obr. 3.10: Schéma Inception modulu .....	51
Obr. 4.1: Grad-CAM + signál EKG, správná klasifikace (0,98).....	63
Obr. 4.2: Grad-CAM + signál EKG, správná klasifikace (1,00).....	64
Obr. 4.3: Grad-CAM + signál EKG, nesprávná klasifikace (0,18).....	64
Obr. 4.4: Graf závislosti jednotlivých metrik na zvoleném prahu .....	65
Obr. 4.5: Záznam EKG (svod III) s lokalizací PVC .....	66
Obr. 4.6: Záznam EKG (svod III) se správnou lokalizací PVC .....	67
Obr. 4.7: Záznam EKG (svod III) s částečně správnou lokalizací PVC .....	68
Obr. 4.8: Záznam EKG (svod III) zobrazující PAC.....	68
Obr. 4.9: Záznam EKG (svod III) – šum, artefakty .....	69

## SEZNAM TABULEK

Tab. 3.1: Třídy vyskytující se v datech a jejich počty.....	40
Tab. 3.2: Nastavení filtrace .....	42
Tab. 3.3: Výsledky testování poměrů rozdělení dat.....	45
Tab. 3.4: Nastavení hyperparametrů pro výběr architektury .....	49
Tab. 3.5: Výsledky učení za použití různých architektur.....	50
Tab. 3.6: Hyperparametry a jejich rozsahy, Bayesovská optimalizace.....	52
Tab. 4.1: Výsledek Bayesovské optimalizace, klasifikace do 2 tříd .....	54
Tab. 4.2: Výsledek Bayesovské optimalizace, klasifikace do 3 tříd .....	55
Tab. 4.3: Výsledek Bayesovské optimalizace, klasifikace do 8 tříd .....	55
Tab. 4.4: Hyperparametry použité pro klasifikaci do 2 tříd .....	56
Tab. 4.5: Matice záměn, klasifikace do 2 tříd, validační množina.....	57
Tab. 4.6: F1 skóre, klasifikace do 2 tříd, validační množina .....	57
Tab. 4.7: Počet záznamů ve třídách, klasifikace do 2 tříd, testovací množina .....	57
Tab. 4.8: Matice záměn, klasifikace do 2 tříd, testovací množina .....	57
Tab. 4.9: Metriky, klasifikace do 2 tříd, testovací množina .....	57
Tab. 4.10: Hyperparametry použité pro klasifikaci do 3 tříd .....	58
Tab. 4.11: Matice záměn, klasifikace do 3 tříd, validační množina.....	58
Tab. 4.12: F1 skóre, klasifikace do 3 tříd, validační množina .....	58
Tab. 4.13: Počet záznamů ve třídách, klasifikace do 3 tříd, testovací množina .....	59
Tab. 4.14: Matice záměn, klasifikace do 3 tříd, testovací množina .....	59
Tab. 4.15: Metriky, klasifikace do 3 tříd, testovací množina.....	59
Tab. 4.16: Hyperparametry použité pro klasifikaci do 8 tříd .....	60
Tab. 4.17: Matice záměn, klasifikace do 8 tříd, validační množina.....	60
Tab. 4.18: F1 skóre, klasifikace do 8 tříd, validační množina .....	61
Tab. 4.19: Počet záznamů ve třídách, klasifikace do 8 tříd, testovací množina.....	61
Tab. 4.20: Matice záměn, klasifikace do 8 tříd, testovací množina .....	61
Tab. 4.21: Metriky, klasifikace do 8 tříd, testovací množina.....	62
Tab. 4.22: Srovnání hlavních metrik klasifikace do 2,3 a 8 tříd .....	62
Tab. 4.23: Délky učení.....	63
Tab. 4.24: Evaluace výsledků lokalizace PVC .....	67
Tab. 5.1: Doporučované úpravy rozsahů pro Bayesovskou optimalizaci hyperparametrů.....	71
Tab. 5.2: Srovnání s literaturou.....	73

# ÚVOD

Srdeční arytmie jsou poruchy vzniku nebo převodu elektrických impulzů v srdci. Lidé trpící arytmiemi mohou být asymptomaticí, ale také mohou mít sníženou kvalitu života tím, že pociťují bolesti na hrudi a nepříjemně rychlý tlukot srdce, mají sklony k omdlávání nebo mají dechové obtíže. V nejhorších případech mohou arytmie vést k poškození srdce a systémovým emboliím. Lidská populace po celém světě stárne a prevalence arytmií se zvyšuje. Proto je důležitá diagnostika těchto zdravotních obtíží, aby se předešlo větším komplikacím asociovaným se srdečními arytmiemi. [1]

Hlavním diagnostickým prostředkem pro tento typ onemocnění je elektrokardiografie (EKG). Délka záznamů EKG ovšem narůstá a zvyšuje se počet pacientů, proto se objevují snahy o automatickou detekci arytmií ke snížení personální i časové náročnosti diagnostiky. [1]

Tato diplomová práce se zabývá identifikací abnormálních segmentů EKG pomocí hlubokého učení se zaměřením na detekci komorových extrasystol. Nejdříve je představen převodní systém srdeční a nejčastěji diagnostikované arytmie, další části práce již pojednávají o učení i stavbě konvolučních neuronových sítí. A protože je pro lékaře důležitá informace i o tom, kde přesně se určitá arytmie vyskytuje v EKG, je popsán přístup Multiple-Instance Learning a gradientem vážený Class Activation Mapping (Grad-CAM).

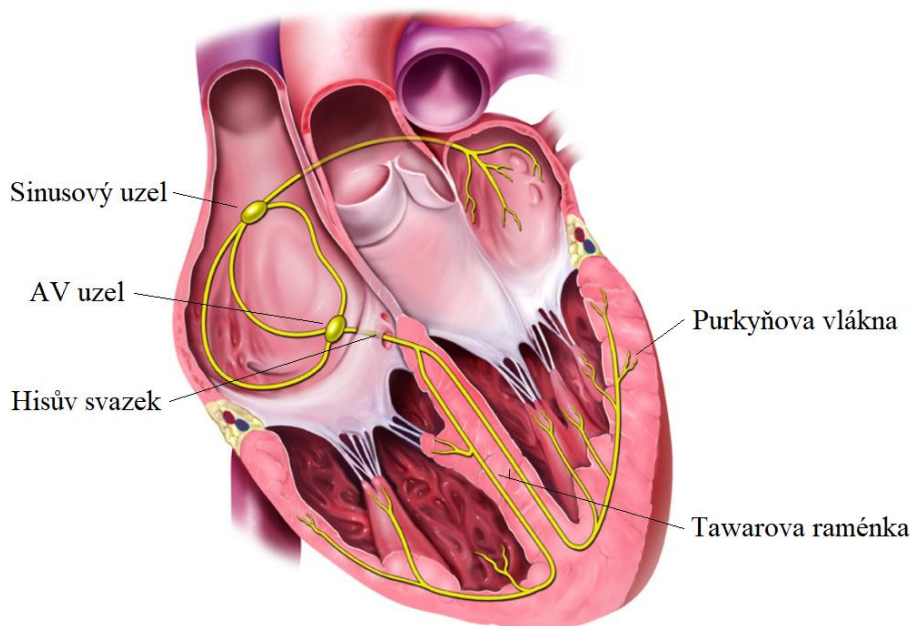
V praktické části je představen program pro klasifikaci komorových extrasystol a zobrazení lokalit výskytu těchto arytmií v EKG vytvořený prostřednictvím programovacího jazyka Python s využitím architektury InceptionTime. Je popsáno předzpracování dat, které je nedílnou součástí toho, aby měl klasifikační algoritmus co nejlepší výsledky. Dále jsou rozebrány detaily trénování modelu spolu s popisem provedené Bayesovské optimalizace hyperparametrů. Klasifikace záznamů probíhala do 2 tříd (komorové extrasystoly a ostatní), do 3 tříd (komorové extrasystoly, sinusový rytmus a ostatní) a do 8 tříd (komorové extrasystoly, síňové extrasystoly, sinusový rytmus, fibrilace síní, tachykardie, bradykardie, infarkt myokardu a ostatní). V práci jsou tedy představeny jak výsledky těchto klasifikačních problémů, tak výstupy metody Grad-CAM, která vizualizuje umístění arytmií v signálech, a tyto výsledky jsou na konci práce diskutovány.

# 1. ELEKTROFYZIOLOGIE SRDCE

Myokard obsahuje dva typy buněk – buňky jednoho typu vytváří vzruchy a převádí je do dalších částí srdečního svalu a buňky druhého typu, které se jako odpověď na vzruch kontrahují (pracovní myokard). Oproti příčně pruhovanému svalu tedy elektrické impulzy vznikají přímo uvnitř srdečního svalu a tento jev se označuje jako autonomie srdce. Velkou roli v šíření vzruchu sehrávají gap junctions, tedy typ propojení buněk myokardu. Jakmile vznikne vzruch v určitém místě srdce, tyto spoje zaručí, že se vzruch rozšíří po celém srdci a vždy se tak kontrahují obě síně i obě komory. [2]

## 1.1 Převodní systém srdeční

Převodní systém srdeční se skládá ze sinusového uzlu, atrioventrikulárního uzlu, Hisova svazku, pravého a levého Tawarova raménka a Purkyňových vláken. Celý tento systém, včetně uložení jednotlivých jeho součástí v rámci srdce, je zobrazen na Obr. 1.1. [2]



Obr. 1.1: Převodní systém srdeční [3]

Excitační signál vzniká ve zdravém srdci v sinusovém uzlu, protože má nejvyšší frekvenci spontánní depolarizace. Všechny ostatní části převodního systému mají tuto frekvenci nižší, proto se sinusový uzel označuje jako primární pacemaker. Leží v horní stěně pravé síně a impulzy generované tímto uzlem zapříčiňují kontrakci obou síní, a tedy pohyb krve ze síní do komor. Rychlost vzniku těchto impulzů je ovlivňována autonomním nervovým systémem. [2]

Jakmile projde vzruch přes síně, pokračuje do atrioventrikulárního uzlu (AV uzlu) uloženého ve stěně pravé síně u trikuspidální chlopně. AV uzel zpomaluje šíření impulzu o asi 120 ms, aby se zajistilo, že proběhne kontrakce celé oblasti síní a vypudí se z nich tak všechna krev do komor. Pokud nepracuje správně sinusový uzel (primární pacemaker) a negeneruje tedy excitační signály, přebírá na sebe tuto roli AV uzel jako sekundární pacemaker.

Hisův svazek je pokračováním AV uzlu a je jediným místem, kudy vzruch může projít ze svaloviny síní do komor. Následně se Hisův svazek rozděluje na pravé a levé Tawarovo raménko přenášející vzruch do pravé a levé komory. Poslední částí převodního systému srdečního jsou Purkyňova vlákna rozvádějící impulz po obou komorách. [2,4]

## 1.2 Srdeční arytmie

Jako arytmie se označují poruchy vzniku nebo převodu podráždění – jde o jakékoliv abnormální místo vzniku impulzů, rychlou či naopak pomalou tepovou frekvenci, nepravidelnost rytmu apod. Může se jednat o jednotlivý stah nebo i trvalý problém. Diagnostika se provádí zejména pomocí elektrokardiografie (EKG).

Symptomy arytmií jsou různé – jde zejména o pocity silného bušení srdce a bolesti na hrudi, v případě zhoršení funkce srdce o závratě nebo mdloby a v nejhrošším případě může rovněž dojít k náhlému úmrtí pacienta. [5]

Příčinu vzniku arytmií může být poměrně těžké určit. Častým viníkem je ischemická choroba srdeční nebo různá postižení srdečního svalu. Také některé léky nebo stres a zvýšená fyzická zátěž mohou způsobovat arytmie. Silný vliv na vznik arytmií má i elektrolytová nerovnováha, např. nedostatek draslíku. [2,5]

V další části této kapitoly budou představeny příklady arytmií, které se běžně vyskytují v populaci.

### 1.2.1 Převodní blokády

Převodní blokády označují všechny narušení postupu elektrických impulzů, kdy tyto blokády mohou vzniknout v kterémkoliv místě převodního systému srdečního. Podle místa postižení je možné jejich rozdělení na několik typů.

Atrioventrikulární (AV) blokády jsou blokády lokalizované mezi sinusovým uzlem a Purkyňovými vlákny. U AV blokády 1. stupně dochází ke zpoždění převodu impulzů v AV uzlu nebo v Hisově svazku, všechny vzruchy jsou převedeny ze síní do komor a dochází tedy jen ke zpoždění převodu. AV blokády 2. stupně vznikají ve stejných místech, do komor dorazí ale pouze některé impulzy ze síní, poměr vln P ke komplexům QRS se tedy zvedá. AV blokáda 3. stupně je tou nejzávažnější formou této arytmie, dochází totiž k tomu, že do komor už neproniká žádný impulz ze síní. Komory tedy zareagují náhradním rytmem (idioventrikulárním), který má nižší frekvenci než sinusový rytmus, což má za následek, že komory a síně bijí nezávisle na sobě svým vlastním

rytmem. Vlny P se tedy v EKG objevují častěji než komplexy QRS a jejich vzájemná poloha je na sobě nezávislá. [5]

Dalším typem převodních blokády jsou blokády ramének. Při blokáde pravého Tawarova raménka (RBBB = right bundle branch block) má zpoždění depolarizace pravé komory, která proběhne až po dokončení depolarizace komory levé. U blokády levého Tawarova raménka (LBBB = left bundle branch block) má naopak zpoždění depolarizace levé komory. Obě blokády mají velký vliv na tvar a šířku QRS komplexů. Blokáda pravého Tawarova raménka se projevuje rozšířením QRS komplexu a vznikem dvou kmitů R – QRS komplex má tedy konfiguraci označovanou jako RSR'. Blokáda levého Tawarova raménka také zapříčiňuje rozšíření QRS komplexu, který má nyní tvar podobný písmenu M s rozeklanými vrcholy. [6]

### **1.2.2 Fibrilace síní**

Jako fibrilace síní se označují nekontrolované stahy síní, kdy AV uzel je zahlcen více než 500 depolarizačními vlnami za minutu. Tyto stahy se šíří všemi různými směry a kmity tedy nejsou pravidelné. AV uzel propustí pouze některé z nich a komory tak kmitají frekvencí přibližně 120-180 stahů za minutu. V EKG se fibrilace síní pozná tak, že nejsou viditelné vlny P, ale základní linie EKG se vlní, což se označuje jako fibrilační vlny. Komplexy QRS mají normální tvar. [5,6]

### **1.2.3 Flutter síní**

Flutter síní je jev, kdy je frekvence stahu síní nad 250 za minutu, impulzy jsou pravidelné a izoelektrická linie mezi vlnami P již není patrná, jsou viditelné pouze tzv. flutterové vlny. [5]

Flutter síní nejčastěji vzniká, když vzruch krouží kolem trikuspidální chlopně. Do AV uzlu přichází impulzy s velmi vysokou frekvencí a tento uzel tedy některé vzruchy blokuje a na komory tak přechází impulzy s mnohem menší frekvencí. Častá je AV blokáda 2:1 (tedy ze dvou vzruchů, které dorazí do AV uzlu, se převede jeden na komory a v EKG je tak patrný jeden QRS komplex na dvě flutterové vlny), popřípadě méně často i 3:1 a 4:1. [5,6]

### **1.2.4 Extrasystoly**

Extrasystoly (předčasné stahy, předčasné kontrakce) označují elektrické impulzy, které depolarizují určitou část srdce dříve, než přijde pravidelný sinusový impulz. Dělí se na ty vznikající v síních, komorách a v junkční oblasti. [6]

U síňových extrasystol (PAC = premature atrial contraction) má vlna P odlišný tvar než vlna P vyvolaná sinusovým impulzem. Junkční extrasystoly vznikají v okolí AV uzlu. U tohoto typu předčasných stahů není vlna P obvykle vůbec patrná, protože je ukryta v QRS komplexu, který má normální tvar. Výskyt síňových a junkčních extrasystol závisí na klinickém stavu pacienta a může být přidružený k různým srdečním onemocněním,



stejně tak se mohou vyskytovat u zcela zdravých jedinců. Rizikem je, že mohou spouštět jiné závažnější arytmie.

Komorové extrasystoly (PVC = premature ventricular contraction) se na EKG projevují abnormálním tvarem QRS komplexu, který je většinou široký (nad 120 nm) a dává informaci o tom, kde extrasystola vznikla (zda v pravé nebo levé komoře). PVC je vždy následována úplnou kompenzační pauzou – tedy vzdálenost QRS komplexu před PVC a QRS komplexu za PVC je stejná jako vzdálenost dvojnásobku běžné vzdálenosti mezi QRS komplexy (jak je ilustrováno na Obr. 1.2).

Pokud pacient nemá žádné srdeční onemocnění, jsou jednotlivé PVC ve většině případů neškodné a nevyžadují léčbu. Pokud jde o více extrasystol za sebou, může se jednat o znak vážnějšího postižení srdce. Problém může nastat i u pacientů v časně fázi infarktu myokardu nebo po infarktu myokardu, kdy s výskytem PVC roste riziko náhlé smrti. [5,7,8]



Obr. 1.2: EKG – komorová extrasystola [8]

### 1.2.5 AV (nodální) reentry tachykardie

AV reentry tachykardie (AVRT) a AV nodální reentry tachykardie (AVNRT) jsou supraventrikulární tachykardie. V srdci jsou v určitém místě alespoň dvě dráhy vedení vzruchu, kdy jedna dráha má jednosměrný blok a druhá dráha má zpomalené vedení. Pokud tedy vzruch přijde v určitém „správném“ čase, zablokuje jednu dráhu a po pomalém průchodu druhou zpomalenou dráhou se vrátí zpět do první dráhy, čímž dojde ke kroužení vzruchu. Pro tento typ tachykardií je udávaným symptomem pravidelné a rychlé bušení srdce s náhlým začátkem a koncem. [9]

AVRT, která je součástí Wolff-Parkinson-Whiteova syndromu (WPW), se projevuje, pokud je přítomna přídavná dráha mezi síní a komorou. Této dráze ovšem nestojí v cestě AV uzel, který by frekvenci impulzů zpomalil, a proto je na EKG viditelný úzký QRS komplex s frekvencí 160 až 240 za minutu.

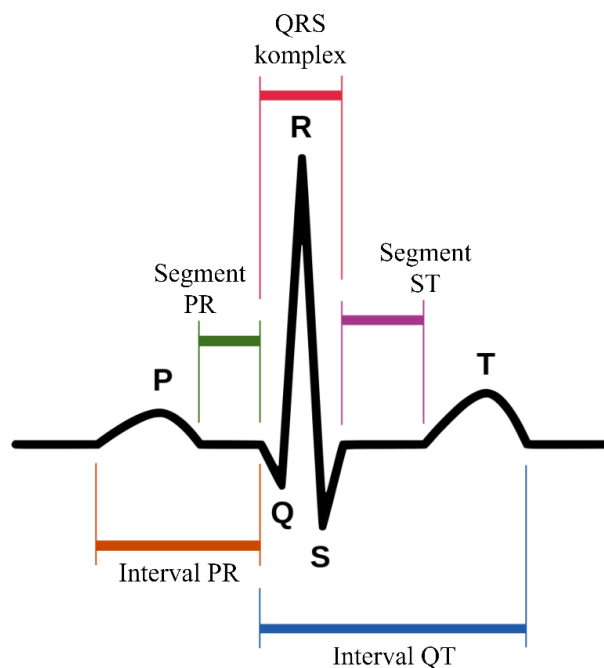
AVNRT je charakterizována přídavnou dráhou v oblasti AV uzlu a je typická pro ženy v mladším věku. Stimulace síní a komor probíhá těsně za sebou, proto jsou ve většině případů vlny P skryty do QRS komplexů. [7]

## 1.3 EKG

EKG je v oblasti kardiologie základní vyšetřovací metodou. Pomocí elektrod přiložených na povrch těla pacienta jsou snímány elektrické změny, které podávají informaci o kontrakci srdečního svalu. Ze záznamů EKG je tak možné určit, zda pacient netrpí nějakým druhem arytmie či jiným onemocněním srdce. [6]

Na Obr. 1.3 je zobrazena křivka fyziologického EKG, jehož základní vlny, kmity a linie jsou následující: [5,6]

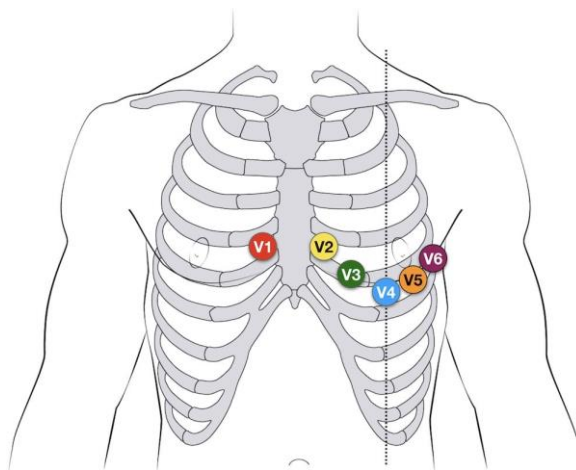
- Vlna P představuje kontrakci (tedy depolarizaci) síní, kdy první část vlny P ilustruje kontrakci pravé síně, protože se v ní nalézá sinusový uzel, a druhá část vlny P stáhne levé síně.
- Interval PR je úsekem od začátku depolarizace síní do začátku depolarizace komor.
- Segment PR označuje pauzu mezi převodem ze síní do komor. Během této doby se vyprázdňují síně a mohou se začít kontrahovat komory.
- Komplex QRS skládající se z kmitů Q, R a S zobrazuje kontrakci komor a jeho amplituda je oproti vlně P značně vyšší, a to z důvodu větší mohutnosti myokardu komor. V komplexu QRS je rovněž vizuálně schována repolarizace síní.
- Interval QT označuje délku trvání depolarizace a repolarizace komor.
- Segment ST je doba od konce depolarizace komor po začátek jejich repolarizace.
- Vlna T následně zaznamenává repolarizaci komor.



Obr. 1.3: Křivka EKG [10]

Záznam se ve většině případů získává pomocí 12svodového EKG, které vyžaduje umístění 10 elektrod na tělo pacienta a srdce, tedy jeho elektrická aktivita, je snímáno z různých směrů, čímž se vytváří celkový obraz o stavu tohoto orgánu.

Z 10 elektrod se čtyři umístí na každou z končetin, čímž se zajistí záznam končetinových svodů tvořený třemi standardními svody (I, II, III) a třemi zvětšenými unipolárními svody (aVL, aVR, aVF). Zbývajících šest elektrod se umístí na hrudník, jak je zobrazeno na Obr. 1.4, a je tak vytvořeno šest hrudních svodů. [5]



Obr. 1.4: Umístění hrudních svodů [11]

### 1.3.1 Rušení EKG

V záznamech EKG se objevují různé typy šumů a pro správnou interpretaci dat je nutné tato rušení filtrovat. Obvyklými typy rušení jsou drift nulové izolinie, síťové rušení nebo rušení myopotenciály.

Drift nulové izolinie je jev, kdy se základní linie signálu pohybuje nahoru a dolů. Je způsoben ději na rozhraní kůže-elektroda, dýcháním a pohyby pacienta. Frekvence driftu bývá obvykle méně než 0,5 Hz, proto se pro jeho filtraci používají např. filtry typu horní propust s mezní frekvencí 0,5 Hz.

Síťové rušení, v České republice tedy na frekvenci 50 Hz, je vyvoláno elektromagnetickými poli elektrovodné sítě, kterou je napájen přístroj samotný a filtrace se provádí využitím pásmové zádrže v oblasti 50 Hz.

Rušení myopotenciály pochází z aktivity svalů pacienta a největší problém představuje při pořizování záznamu EKG za pohybu člověka. Filtrace tohoto typu rušení je poměrně obtížná, protože se nejedná o úzkopásmový šum, ale jeho spektrální obsah se silně překrývá s užitečným spektrem EKG. Určitým řešením může být použití filtrace prostřednictvím klouzavého průměru. [12]

## 2. NEURONOVÉ SÍTĚ

### 2.1 Úvod ke strojovému učení

Neuronové sítě jsou jednou z technik strojového učení. Strojové učení se vyvinulo v druhé polovině 20. století jako podoblast umělé inteligence. Zahrnuje algoritmy, které získávají informace z dat, vytváří tak predikce a samy se učí. Lidé tak nemusejí manuálně analyzovat velká množství dat a vytvářet z nich modely. [13]

Existují čtyři základní typy strojového učení:

- učení s učitelem,
- učení bez učitele,
- částečně řízené učení,
- a zpětnovazebné učení. [14]

Učení s učitelem (používaný je rovněž anglický termín supervised learning) využívá učení prostřednictvím propojování vztahu mezi proměnnými a známými výstupy. Do algoritmu tedy vstupují data s různými příznaky a jejich správný výstup, kdy tento výstup může být prvkem určitého konečného počtu tříd, reálné číslo nebo složitější struktura, jako např. matice nebo vektor. Cílem algoritmu je najít vzorce mezi vstupy a výstupy a vytvořit takový model, který pro určitý vstupní vektor příznaků odvodí výstup, tedy třídu. [14,15]

Učení bez učitele (unsupervised learning) nemá k dispozici klasifikovaná data. Cílem tohoto typu učení je tedy vytvořit takový model, který na vstupu přijme vektor příznaků, najde skryté vzorce a vektor transformuje do jiného vektoru nebo hodnoty, které mohou být použity pro řešení určitého praktického problému. Příkladem učení bez učitele je algoritmus shlukové analýzy k-means. [14,15]

Částečně řízené učení (semi-supervised learning) spojuje učení s učitelem a bez učitele. Soubor dat tedy obsahuje klasifikovaná i neklasifikovaná data, kdy neklasifikovaných je mnohem více. Předpokladem je, že když se použije větší množství neklasifikovaných dat s menším množstvím klasifikovaných dat, je možné vytvořit kvalitnější model. [14]

Posledním základním typem učení je zpětnovazebné učení (reinforcement learning), které neustále zlepšuje model tím, že využívá zpětnou vazbu ve formě odměn z předchozích iterací. Nejprve tedy probíhá průzkum prostředí a jednotlivé aktivity jsou hodnoceny kladně nebo záporně. Pomocí takovéto zpětné vazby se model metodou „pokus-omyl“ sekvenčně učí, co je správně a co je špatně a zlepšuje tak postupně svou funkci. [16]

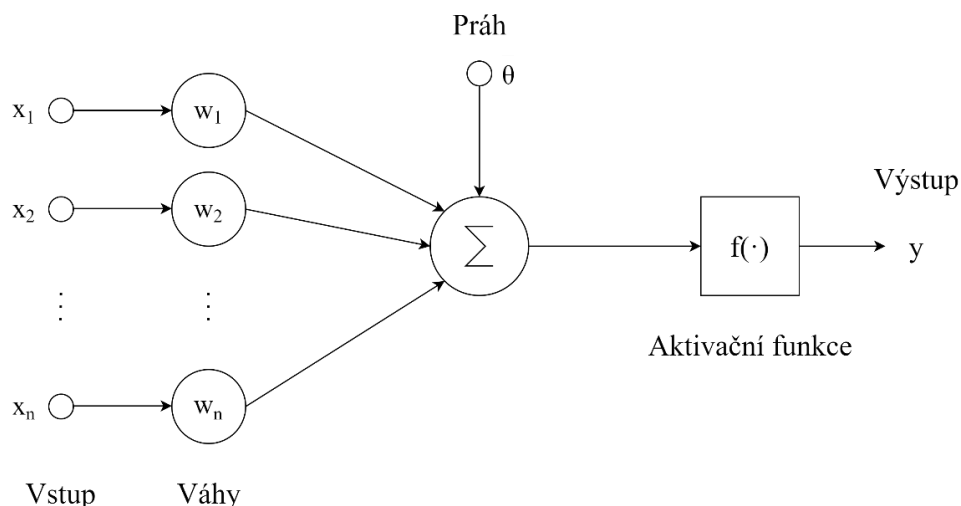
## 2.2 Úvod k neuronovým sítím

Umělé neuronové sítě jsou technikami strojového učení, které simulují mechanismus učení u člověka. Lidská nervová soustava se skládá z neuronů, které jsou k sobě navzájem připojeny pomocí axonů a dendritů, kdy místa spojení se nazývají synapse. Proces učení je ovlivněn silou tohoto synaptického spojení, které se může měnit v závislosti na externím stimulu.

Umělé neuronové sítě tedy obsahují výpočetní jednotky, známé také jako neurony, které odpovídají velmi zjednodušenému matematickému modelu neuronů v nervové soustavě. Tyto výpočetní jednotky jsou propojeny pomocí vah, které odpovídají síle synaptického spojení neuronů v lidském těle a změny velikosti těchto vah zajišťují proces učení.

Trénovací data při využití učení s učitelem zahrnují vstupní data (např. pixelová reprezentace obrázku) a jejich třídy (např. kočka, pes apod.) jako výstupy. Trénovací data jsou vložena do neuronové sítě predikující jejich třídy (např. na obrázku je kočka) a je zjišťována správnost těchto predikcí, kdy je porovnávána predikovaná třída s tou v trénovacích datech. V závislosti na výsledné chybě mezi těmito dvěma hodnotami jsou změněny váhy a celý proces je opakován. Pokud má tedy model dostatečný počet trénovacích dat, po několika opakováních by měl být naučen tak, aby co nejsprávněji určil třídu daného vstupního vektoru. Velkou výhodou neuronových sítí je schopnost generalizace – učí se tedy pomocí daných trénovacích dat, po naučení jsou ale schopny predikovat výstupy neznámých dříve nezpracovaných dat. [17]

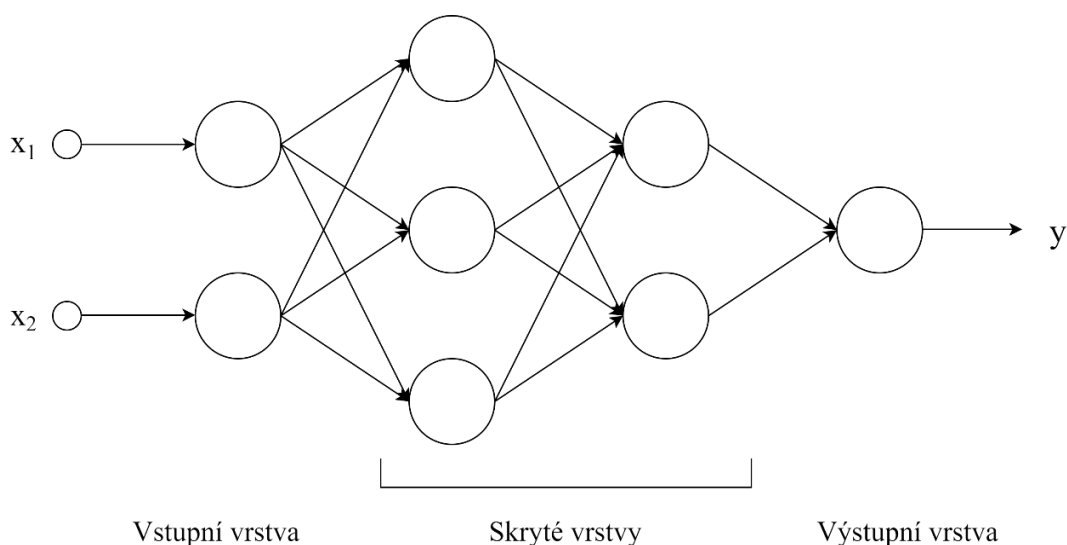
V tomto odstavci bude pro lepší porozumění popsán klasifikační problém s využitím učení s učitelem na nejjednodušší neuronové síti – perceptronu. Perceptron se skládá z jedné vrstvy a jeho schéma je zobrazeno na Obr. 2.1, je schopen provést pouze klasifikaci dat do 2 tříd. Vektor vstupů je v tomto případě mapován rovnou na výstup. Vstupní hodnoty jsou váhovány váhami, které se v průběhu trénování mění a ovlivňují tak proces učení. Váhované hodnoty jsou sečteny a následně je od tohoto čísla odečten práh. Výsledné číslo je argumentem aktivační funkce, která určuje výstup perceptronu, čímž proběhne klasifikace dat do tříd (např. výstup ve formě 0 nebo 1, kdy 0 označuje jednu třídu a 1 druhou třídu). Následně je porovnávána predikovaná třída s třídou z trénovacích dat a na základě chyby jsou upraveny váhy a práh. [18,19]



Obr. 2.1: Schéma perceptronu, převzato z [20]

Perceptron má velmi omezené použití, a to pro problémy, které jsou lineárně separabilní, proto se ve většině aplikací používají neuronové sítě obsahující více neuronů a více vrstev, podobně jako v procesu učení člověka nestačí pouze jeden biologický neuron, ale celé sítě neuronů.

Vícevrstvé sítě obsahují kromě vstupní a výstupní vrstvy také další vrstvy, které se nazývají skryté, kdy počet těchto vrstev se volí v závislosti na řešeném problému. Neuronové sítě obsahující maximálně dvě skryté vrstvy se označují jako mělké neuronové sítě, pokud obsahují skrytých vrstev více, jedná se o hluboké neuronové sítě. Příklad mělké neuronové sítě se dvěma skrytými vrstvami je zobrazen na Obr. 2.2. Celá tato práce bude pojednávat o hlubokých neuronových sítích. [19]



Obr. 2.2: Vícevrstvá neuronová síť, převzato z [20]

Proces učení u vícevrstvých sítí je samozřejmě složitější než u jednovrstvého perceptronu. Na začátku je nutné inicializovat váhy, např. náhodně z normálního rozložení se střední hodnotou 0 a směrodatnou odchylkou 0,01 [21]. Nejdříve probíhá dopředné vybavení, kdy je na vstup přiveden vektor z trénovací množiny a prostřednictvím vah a aktivační funkce je vypočten výstup ve formě skrytého aktivačního vektoru  $h$ . Vstup může být ignorován nebo přenesen na aktivační funkci v závislosti na velikosti váhy a podobnou možnost má následně i aktivační funkce, která může v závislosti na jejím průběhu rovněž filtrovat data (pokud poskytuje jako výstup 0). Výstupy jedné vrstvy jsou následně předány na vstupy následující vrstvy sítě.

Dále je spočtena chyba ve výstupní vrstvě pomocí vybrané ztrátové funkce – je tedy zjištěn rozdíl mezi predikovaným výstupem a skutečným výstupem, kdy se model snaží tento rozdíl (a tedy ztrátovou funkci) minimalizovat.

Jakmile je určena chyba na výstupu poslední vrstvy, využije se k úpravě vah skrytých vrstev tím, že se postupně šíří celou sítí až k jejímu začátku, k čemuž se využívá algoritmu zpětného šíření chyby. Jeho účelem je zjistit gradienty ztrátové funkce s ohledem na jednotlivé váhy, kdy tyto gradienty jsou následně použity na aktualizování daných vah. A protože je toto prováděno směrem od výstupní vrstvy ke vstupní, nazývá se tento proces jako zpětné šíření chyby. Tímto způsobem je tak možné upravit jednotlivé váhy všech neuronů a pokračovat v procesu učení. Za výpočet gradientů (a tak i aktualizaci velikosti vah) jsou zodpovědné optimalizační algoritmy, také nazývané učební algoritmy, které jsou více popsány v kapitole 2.6. [17,19,21,22]

## 2.3 Hluboké učení

Hluboké neuronové sítě jsou sítě obsahující více skrytých vrstev neuronů a mají široké spektrum využití - často se využívají např. při klasifikaci obrazů [23], v robotice [24], v biomedicínských oblastech [25] nebo při rozpoznávání hlasu [26]. Jejich velkou výhodou je to, že samy extrahují příznaky ze vstupních dat oproti mělkým neuronovým sítím, které na vstupu vyžadují již nalezené příznaky a data je tak před vstupem do sítě nutné složitě zpracovávat a příznaky extrahovat. Hluboké učení také lépe nachází globální vztahy a vzory v datech. Kapacita učení těchto sítí je úměrná počtu dostupných dat – se stoupajícím počtem dat se tedy zvyšuje výkon sítě. V případě mělkých neuronových sítí nebo jiných tradičních algoritmů strojového učení dosáhne výkon sítě při zvyšování počtu dat fáze, kdy se již výkon nezlepšuje. [20]

## 2.4 Konvoluční neuronové sítě

Umělé neuronové sítě ve smyslu vícevrstvého perceptronu jsou tvořeny neurony, které jsou propojeny každý s každým vždy ve dvou sousedních vrstvách, což přináší nevýhodu ve velkém počtu optimalizovaných vah. S rostoucí velikostí dat a vysokým počtem vrstev tedy dochází k tomu, že je výpočetní náročnost celé operace vysoká a může rovněž

docházet k přeučení modelu. Počet vah v jedné plně propojené vrstvě je roven součinu délky vstupu a počtu neuronů v dané vrstvě, tedy v případě vstupních dat ve formě 2D matice o velikosti 100x100 a 100 neuronech ve vrstvě by počet parametrů byl jeden milion. [17,21]

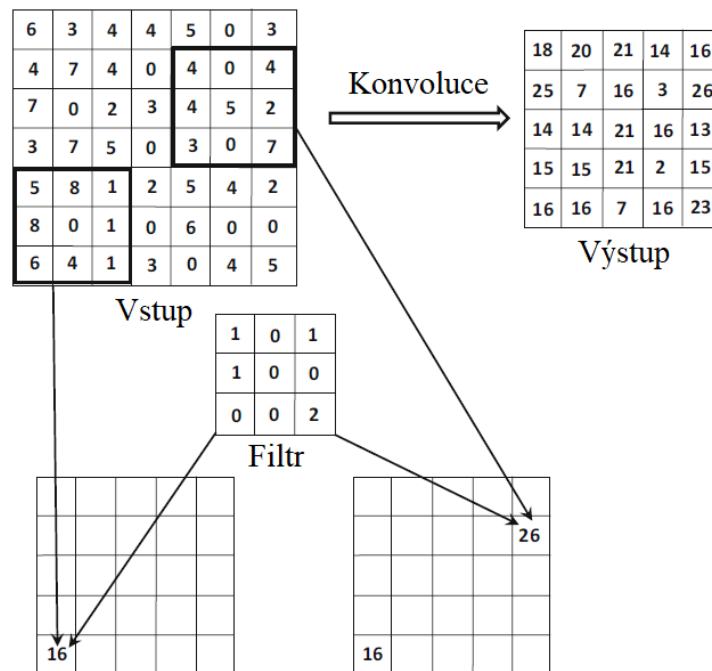
S řešením přichází konvoluční neuronové sítě (CNN = convolutional neural network), které pracují s prostorovou strukturou dat. Neurony určité konvoluční vrstvy jsou propojeny vždy jen s určitou malou oblastí neuronů předcházející konvoluční vrstvy (a následující konvoluční vrstvy), čímž se uchovávají prostorové závislosti mezi jednotlivými regiony vstupu. Vstup do CNN není zploštěn na jednorozměrný vektor, ale pracuje se s více dimenzemi. Např. v případě 12svodového EKG by jednou dimenzí byly samotné signály EKG ve formě vektoru a druhou dimenzí jednotlivé svody, tedy počet kanálů vstupu by byl 12. Výhodou těchto neuronových sítí je i nízký počet vah ve vrstvě, jelikož jsou sdíleny mezi neurony. Počet vah ve vrstvě závisí na velikosti filtru, počtu kanálů a počtu filtrů a nebývá větší než několik stovek, což je v porovnání s počtem vah u plně propojených vrstev (FCN = fully connected layer) velký rozdíl. Nižší počet učících se vah přináší velkou časovou úsporu při trénování, a tedy i možnost trénovat rozsáhlejší modely. [17,21]

Konvoluční neuronové sítě se skládají z jednotlivých vrstev sestavených do určité architektury sítě, kdy některé vrstvy jsou typické přímo pro tento typ modelů (tedy CNN vždy musí obsahovat konvoluční vrstvy). Většina vrstev provádí extrakci příznaků a až poslední vrstva nebo několik posledních vrstev slouží pro klasifikaci. Příznaky extrahované na začátku sítě jsou nízkourovňové a později se stávají komplexnějšími a jedná se tedy o příznaky vyšší úrovně. Existují 1D, 2D i 3D konvoluční neuronové sítě v závislosti na vstupních datech. V následujících kapitolách budou popsány jednotlivé základní vrstvy CNN. [25]

### **2.4.1 Konvoluční vrstva**

Konvoluční vrstva je základním stavebním prvkem CNN. Prvním vstupem do konvoluční vrstvy je matice číselných hodnot, kdy pro názornost bude v této kapitole pojednáváno o vstupu ve formě barevného obrazu s tím, že přizpůsobení algoritmu pro jinou dimenzionalitu vstupní matice není obtížné. Obraz má tedy určitou výšku, šířku a hloubku – např. RGB obraz by byl složen ze tří 2D matic, tedy jednoho kanálu pro každou základní barvu. Následně je na vstup aplikován filtr (= konvoluční jádro) o stejné hloubce (počtu kanálů) jako má obraz a je provedena konvoluční operace – filtr je postupně posouván po celém vstupním obraze a je počítán skalární součin mezi filtrem a danou oblastí vstupní matice. Výstupem konvoluční vrstvy jsou příznakové mapy o hloubce stejné, jako je počet extrahovaných příznaků, tedy počet filtrů. Schéma procesu je zobrazeno na Obr. 2.3. Klíčovou vlastností CNN je tedy sdílení vah – konvoluční jádro je ve všech místech obrazu stejné. [17,25]





Obr. 2.3: Schéma konvoluční vrstvy, převzato z [17]

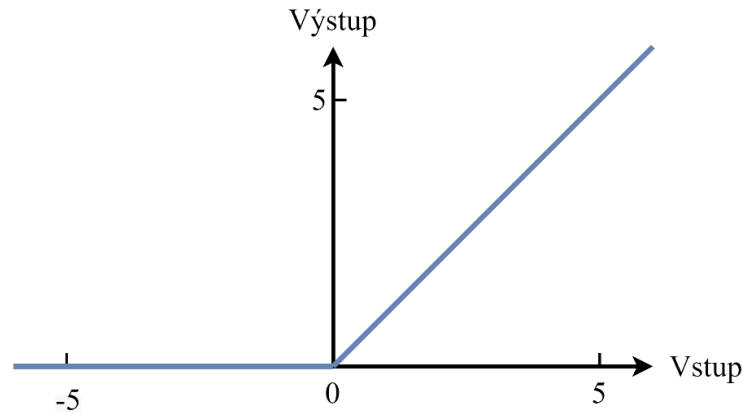
Je možné nastavit různé parametry konvoluční vrstvy, a to velikost filtru, počet filtrů, krok filtru (stride) a doplnění vstupu nulami. Filtry jsou ve většině případů symetrické, čím menší je jejich velikost, tím jemnější detaily obrazu odhalují. Např. 2D filtry jsou časté ve velikosti 3x3, 5x5 nebo 7x7. Počet filtrů udává počet příznakových map na výstupu z dané vrstvy. Krok filtru udává velikost kroku při posunu filtru přes obraz a většinou bývá zvolen jako 1 nebo 2. Pokud je na vstupu velký obraz s komplexními příznaky, vyšší krok filtru pomůže tyto příznaky extrahovat. Zároveň vyšší krok filtru snižuje výpočetní náročnost, ale také se tímto snižuje schopnost detekce jemných detailů. Při doplnění vstupu nulami (zero-padding) je ovlivňována velikost výstupu – tedy je provedeno doplnění nulami, pokud je žádoucí zachovat stejnou velikost výstupu jako vstupu. [17,22]

#### 2.4.2 Aktivační vrstva

Aktivační funkce je u CNN samostatnou vrstvou, která většinou následuje po konvoluční vrstvě. Výstupy konvoluční vrstvy ve formě příznakové mapy jsou převedeny přes nelineární aktivační funkci prostřednictvím bodové transformace. Výstupem aktivační vrstvy je aktivační mapa se stejnými rozměry jako vstupní mapa. V poslední vrstvě části neuronové sítě určené pro klasifikaci je vhodné použít např. funkci SoftMax, která převádí výstupy na pravděpodobnosti příslušnosti k určité třídě. [17,21,25]

Funkce tanh či sigmoida sice lépe reprezentují biologický neuron, velmi oblíbenou aktivační funkcí je ovšem ReLU (= rectified linear unit), která je zobrazena na Obr. 2.4. Tato funkce zachovává pozitivní hodnoty (její průběh je v tomto úseku lineární), a je

nulová při záporných vstupních hodnotách. Rovněž její derivace je vždy 1 pro pozitivní vstupní hodnoty a 0 pro záporné hodnoty, což redukuje problém mizějícího gradientu. V poslední řadě je ReLU funkci také jednoduché implementovat. [17,21]

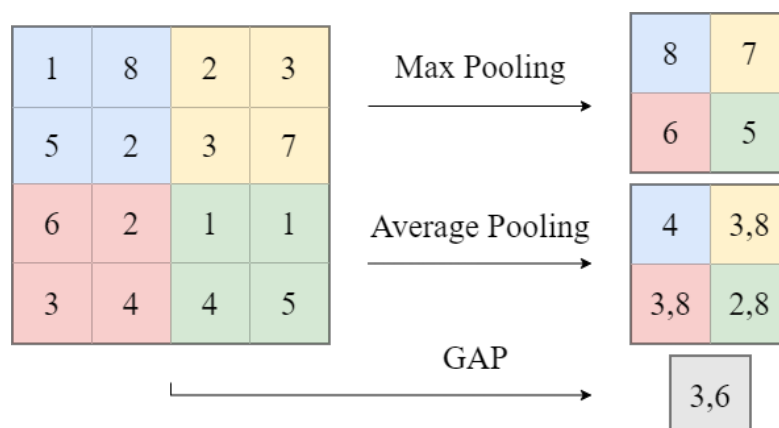


Obr. 2.4: Aktivační funkce ReLU

### 2.4.3 Pooling vrstva

Pooling vrstvy provádějí podvzorkování dat – snižují výšku a šířku vrstvy prostřednictvím komprese dat, hloubka zůstává neměnná. Používají se ze dvou hlavních důvodů. Prvním je samozřejmě samotné snížení velikosti vrstvy. Druhým důvodem je, že použitím pooling vrstev se model stává robustnějším ke změně pozic příznaků a lépe tak generalizuje.

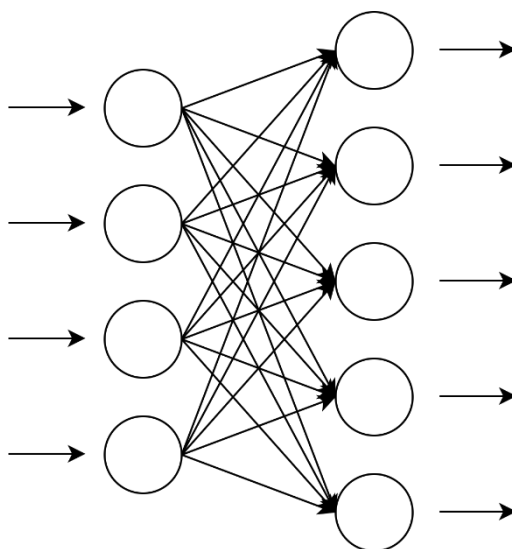
Typ komprese dat definuje i typ samotné pooling vrstvy. Max pooling vybírá vždy maximální hodnotu z určitého okruhu a tu uchovává na výstupu. Dalším typem této vrstvy je např. average pooling, tedy využívá se průměru hodnot z určité oblasti. Jiným příkladem je i global average pooling (GAP), který převede každou příznakovou mapu (každý kanál) na jedno číslo, které je průměrem celé oblasti. Příklad použití všech těchto typů pooling vrstev je zobrazen na Obr. 2.5. [21,25]



Obr. 2.5: Příklad použití pooling vrstev

#### 2.4.4 Plně propojená vrstva

Plně propojená vrstva bývá umístěna na konci celé architektury sítě a funguje jako vrstva vícevrstvého perceptronu, tedy jsou propojeny všechny neurony dvou sousedních vrstev prostřednictvím učicích se vah tak, jak je zobrazeno schématem na Obr. 2.6. Vstupem je zploštěná příznaková mapa vystupující z poslední vrstvy části sítě extrahující příznaky. Většinou se využívá více plně propojených vrstev k dosažení většího výkonu výpočtů a jejich povaha závisí na dané aplikaci – např. zda se jedná o klasifikační nebo regresní problém. V případě klasifikace má poslední plně propojená vrstva stejné množství výstupních uzlů jako je počet tříd. [17,25]



Obr. 2.6: Schéma plně propojené vrstvy

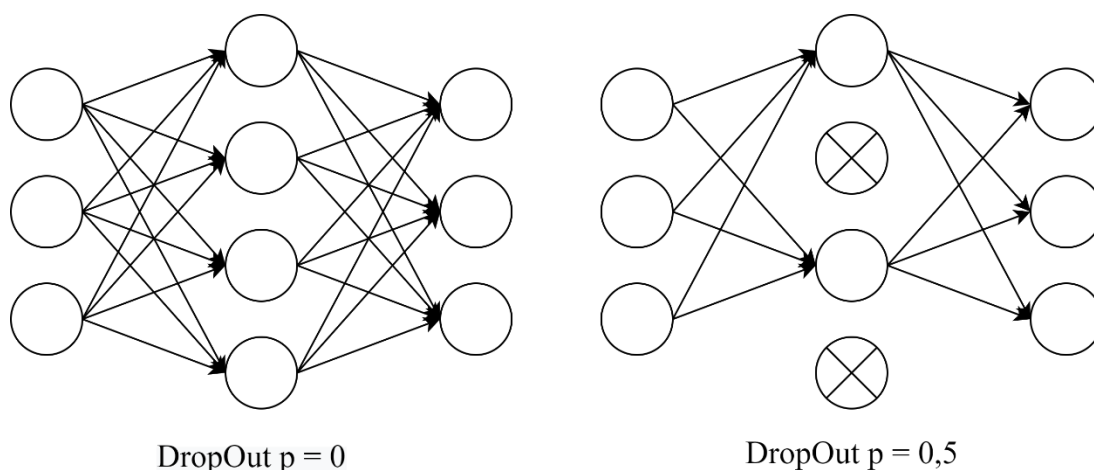
#### 2.4.5 BatchNorm vrstva

Batch normalizace (batch označuje dávku dat, která vstupuje do sítě) slouží k potlačení projevů mizejícího a explodujícího gradientu (více popsáno v kapitole 2.7), které způsobují náhlé snižování či zvyšování aktivačních gradientů. Když je trénován model, dochází k tomu, že proměnné ve vnitřních částech sítě mají velmi odlišné hodnoty, což ohrožuje konvergenci dané sítě a BatchNorm vrstva tedy provádí normalizaci mezi skrytými vrstvami v rámci jedné dávky. Každá BatchNorm jednotka obsahuje učicí se parametry  $\gamma$  (směrodatná odchylka dávky) a  $\beta$  (střední hodnota dávky), které regulují úroveň normalizace v dané jednotce. Batch normalizace může být provedena před aplikováním aktivační funkce, nebo po aplikování aktivační funkce. [17,21]

#### 2.4.6 DropOut vrstva

DropOut je způsob regularizace sítě u plně propojených vrstev, který se využívá pouze během učení modelu. Parametrem je pravděpodobnost, s jakou bude vypnut každý jednotlivý neuron v síti. To znamená, že pokud by byla tato pravděpodobnost 0,5, bude

50% šance, že neuron bude vypnut, tedy v síti nakonec zůstane aktivních 50 % neuronů. Tímto způsobem je provedeno učení pomocí mnoha menších sítí, kdy menší síť mají nižší tendenci přeučovat. Tato metoda také napomáhá tomu, aby na sobě nebyly jednotlivé neurony tolik závislé. Ve fázi predikce je Dropout inaktivován. Na Obr. 2.7 vlevo je možné vidět schéma sítě tvořené plně propojenými vrstvami (vstupní, skrytá a výstupní vrstva) s pravděpodobností Dropout 0, tedy jsou zachovány všechny neurony ve skryté vrstvě. Na Obr. 2.7 vpravo se jedná o stejnou síť, ale s pravděpodobností Dropout 0,5 - je tedy vypnuta polovina neuronů ve skryté vrstvě. [17,27]



Obr. 2.7: Vlevo: Dropout s  $p = 0$ , vpravo Dropout s  $p = 0,5$ , převzato z [21]

## 2.5 Ztrátová funkce

Ztrátová funkce (z angl. loss function) je formální měření, jak dobře klasifikuje daný model. Účelem trénování modelu je minimalizovat rozdíl mezi predikcí a skutečnou třídou, kdy tento rozdíl je spočítán právě pomocí ztrátové funkce (funkce se tedy nazývá ztrátová, protože je minimalizována). [21]

Výběr ztrátové funkce je důležitou součástí procesu před trénováním modelu a závisí na řešeném problému, vstupních datech i žádaném výstupu. V klasifikačních problémech se běžně využívá jako poslední aktivační vrstva SoftMax, tedy výstupem je pravděpodobnost příslušnosti k určité třídě a za ztrátovou funkci je tak vybrána křížová entropie, která je pro každou instanci definována jako

$$L = -\log(\hat{y}_r), \quad (2.1)$$

kde  $\hat{y}_r$  označuje pravděpodobnost příslušnosti k třídě  $r$ , kdy třída  $r$  je skutečnou třídou daného vzorku. Tedy při ukázce na příkladu je hodnota ztrátové funkce  $L = 2,3$ , pokud je pravděpodobnost příslušnosti daného vzorku k dané třídě  $0,1$  a  $L = 0$  při

pravděpodobnosti příslušnosti daného vzorku k dané třídě 1. Výsledkem je tak vysoká hodnota ztrátové funkce při špatné kvalitě predikce a nízká až nulová hodnota ztrátové funkce při správné klasifikaci. [17]

## 2.6 Učební algoritmy

Existuje několik různých učebních algoritmů, kdy tím nejjednodušším je metoda gradientního sestupu, ve které se používá gradient chybové funkce k aktualizování parametrů (vah). Využívá učebního kroku o dané délce a může dojít k tomu, že se nepohybuje do optimálního směru nejstrmějšího sestupu, protože je např. učební krok nastaven moc velký a algoritmus se nedostane do globálního minima. Tato metoda je navíc pomalá a nevhodná pro velké soubory dat. Stochastický gradientní sestup, tedy úprava základního algoritmu, využívá k aktualizování parametrů vždy pouze jeden vzorek ze souboru dat a výpočet je tak rychlejší než u dříve zmíněného algoritmu. Využívá stochastického gradientu, tedy odhadu gradientu. Střední cestou mezi těmito dvěma algoritmy je „minibatch“ gradientní sestup, kdy se pro výpočet používá vždy dávky (minibatch) vzorků. [14,17,21]

Výše zmíněné nevýhody se snaží kompenzovat některé další modifikace metody gradientního sestupu, mezi které patří např. stochastický gradientní sestup s hybností. Směr pohybu je zprůměrovaným směrem několika předchozích kroků, tedy několika předchozích gradientů. Rychlost konvergence ke globálnímu minimu je v tomto případě vyšší než u dříve zmíněných metod. [17]

Dalším typem učebních algoritmů jsou ty s úpravou učebního kroku pro jednotlivé parametry. AdaGrad využívá sumu kvadratických velikostí gradientů pro každý parametr, problémem je ale to, že tato suma postupem času vysoce narůstá a učení se tak stává pomalejším, až se prakticky zastaví. Pro většinu modelů se tedy nehodí. [28]

RMSProp optimalizuje parametry v závislosti na rychlosti jejich změny – pokud se parametr rychle mění, a je tedy nestabilní, rychlost této změny se snižuje a pokud se parametr mění pomalu, rychlost změny se zvýší, aby nedošlo k zastavení učení parametru. Oproti AdaGrad má výhodu, že se jednotlivé gradienty exponenciálně zapomínají a nedojde tak k akumulaci gradientu a následnému zastavení učení. [17,28]

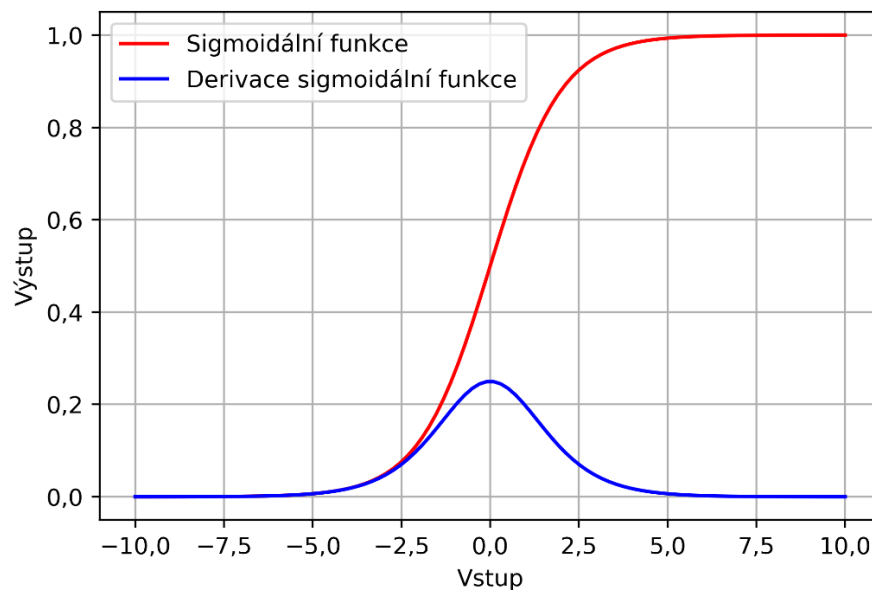
Metoda Adam kombinuje RMSProp a použití hybnosti, kdy se rychlost učení vždy určuje exponenciálně zapomínanými průměry gradientu a kvadratického gradientu pro každý parametr. Algoritmus je jednoduchý na implementaci a nemá vysoké nároky na paměť, rovněž funguje dobře u obsáhlých souborů dat. Může mít ovšem problémy s konvergencí, pokud pracuje s gradienty o velkém rozptylu. [21,28,29]

## 2.7 Problémy konvolučních neuronových sítí

Konvoluční neuronové sítě mají mnoho vrstev, což se pojí s určitými problémy, které budou rozebrány níže. Může tak být náročné je učit kvůli tomu, jaký vztah mezi sebou mají gradienty ve vrstvách na začátku a ke konci sítě.

Při zpětném šíření chyby se váhy upravují dle velikosti a směru gradientu. Pokud jsou ovšem gradienty nestabilní (neočekávaně vysoké nebo nízké), úpravy vah se mohou v dalších vrstvách exponenciálně zvyšovat nebo snižovat, což způsobuje dva základní typy problémů: problém explodujícího gradientu a problém mizejícího gradientu.

Problém mizejícího gradientu nastává, když jsou váhy upravovány pouze nepatrně a učení modelu je tak prakticky nemožné. Zda nastane tato situace, se dá např. dobře ovlivnit výběrem vhodné aktivační funkce. Dříve byla oblíbenou aktivační funkcí sigmoidální funkce (Obr. 2.8), a to proto, že silně připomínala funkci biologického neuronu a první umělé neuronové sítě byly inspirovány právě funkcí biologických nervových soustav. Ze své podstaty nemá nikdy sigmoidální funkce gradient vyšší než 0,25. Tedy při zpětném šíření chyby se gradient postupně snižuje, až je tak malý, že se síť přestane učit. Právě tento jev se označuje jako problém mizejícího gradientu a v tomto případě je vhodné ho vyřešit použitím jiné aktivační funkce, např. ReLU, která je stabilnější (její derivace je 0 nebo 1).



Obr. 2.8: Sigmoidální funkce a její derivace

Problém explodujícího gradientu nastává, když jsou inicializované váhy vysoké, aktualizace vah jsou následně také vysoké a učební algoritmus tak nekonverguje. Jak je tedy zřejmé, CNN jsou v ohledu velikosti gradientu poměrně nestabilní, v dnešní době už však existují techniky, které tyto problémy potlačují, např. batch normalizace. [17,21]

## 2.8 Hyperparametry sítě

Hyperparametry sítě jsou parametry, které nejsou optimalizovány algoritmem samotným (jako parametry, tedy váhy jednotlivých propojení sítě), ale jejich optimální hodnoty musí být nalezeny před započítáním samotného učení. Jedná se např. o krok učení nebo počet epoch. Pokud nejsou hodnoty hyperparametrů nastaveny správně, učení sítě nebude probíhat s dobrými výsledky – např. se síť bude přeučovat, nedoučovat nebo bude učení moc pomalé, popřípadě bude učení velmi výpočetně náročné a zdlouhavé, což se musí při optimalizaci hyperparametrů také zohlednit. Obecně je ale vždy jeden hyperparametr závislý na hyperparametrech ostatních a při hledání jeho optimální velikosti je nutné se zamyslet nad konkrétním řešeným problémem a použitou sítí. Existuje několik základních metod, jak je možné ty nejvhodnější hyperparametry najít. [14]

Prohledávání stavového prostoru je nejjednodušším typem optimalizování hyperparametrů. Je určen rozsah hodnot jednotlivých hyperparametrů a následně jsou vyzkoušeny všechny jejich kombinace a je nalezena ta nejlepší. Nevýhoda prohledávání stavového prostoru je ovšem zřejmá – při větším počtu hyperparametrů a/nebo při větším rozsahu prohledávaných hodnot je tato metoda velmi časově náročná, protože počet bodů prohledávání se exponenciálně zvyšuje s rostoucím počtem hyperparametrů. Proto se také používá modifikace této metody, kdy jsou nejdříve prohledávány hodnoty hyperparametrů, které mezi sebou mají větší rozestupy. Jakmile se najde optimální kombinace, je provedeno jemnější prohledávání s menšími rozdíly mezi danými hodnotami. Časová náročnost je stále ale poměrně vysoká, proto se užívají i jiné, časově méně náročné, techniky optimalizace hyperparametrů zmíněné níže. [17]

Při použití náhodného prohledávání je poskytnuto statistické rozložení pro každý hyperparametr, ze kterého jsou náhodně vybrány hodnoty k otestování, následně jsou vyzkoušeny různé kombinace hodnot jednotlivých hyperparametrů a je vybrána ta optimální. [14]

Další metodou optimalizace hyperparametrů je Bayesovská optimalizace. Hlavní myšlenkou je snižovat počet kombinací hyperparametrů k otestování tím, že se využívají minulé výsledky učení daných hyperparametrů a výběr pro další učení se tak provádí v okolí hodnot s nejvyšší úspěšností učení v minulosti. Tato metoda optimalizace je tedy schopna najít vhodné hyperparametry poměrně rychle, protože je snížen počet kombinací, pro který je nutno model trénovat. [17,30,31]

V další části kapitoly budou popsány základní hyperparametry neuronových sítí, popřípadě i jejich běžné hodnoty.

### 2.8.1 Počet skrytých vrstev

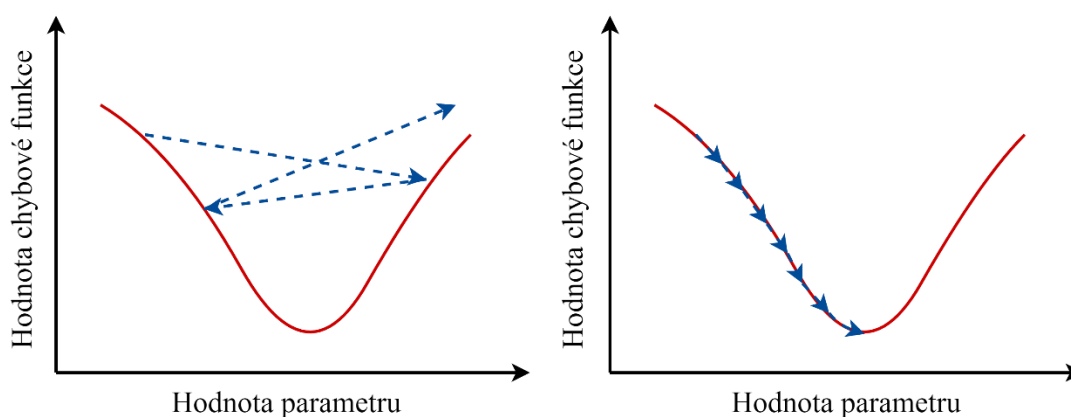
Se zvyšováním výpočetní kapacity počítačů bylo v posledních letech možné navrhovat sítě s větším a větším počtem skrytých vrstev. Obecně více vrstev v síti umožní extrahovat abstraktnější příznaky vyšší úrovně, snadněji ale dojde k přeučení modelu.

U hlubokých sítí je častý i problém mizejícího gradientu, který se s vyšším počtem vrstev prohlubuje, a velká výpočetní náročnost učení. Vždy je nutné hloubku sítě přizpůsobit řešenému problému a dostupným datům. [17]

### 2.8.2 Učební krok

Učební krok ovlivňuje velikost, o jakou jsou upravovány váhy během jejich optimalizace, za účelem snižovat chybu učení neuronové sítě. To znamená, že pokud je učební krok moc vysoký, délka trvání učení je sice nízká, ale zároveň to vede k tomu, že může být nemožné se dostat do globálního minima a algoritmus se tak zacyklí (zobrazeno na Obr. 2.9 vlevo). Pokud je učební krok moc nízký, výsledky algoritmu jsou přesnější (i když je zde riziko spadení do lokálního minima), ale učení trvá velmi dlouho a je výpočetně náročné a neefektivní (zobrazeno na Obr. 2.9 vpravo).

Hodnota učebního kroku se pohybuje přibližně v rozmezí 0,01 a 0,001 [17]. Jako vhodné řešení se tedy jeví použití měnícího se učebního kroku v závislosti na fázi učení. Na začátku učení bude krok větší, aby se zvýšila efektivita algoritmu a následně se bude učební krok snižovat a bude tak vhodný k hledání přesného řešení problému. [22]



Obr. 2.9: Porovnání velkého/malého učebního kroku, převzato z [22]

### 2.8.3 Pravděpodobnost DropOut

Hyperparametrem DropOut vrstvy (viz kapitola 2.4.6) je pravděpodobnost, s jakou bude vypnut každý jednotlivý neuron v síti a běžně se pohybuje v rozmezí 0,2 až 0,5 [17]. Hledá se tedy taková pravděpodobnost, která bude zaručovat nejlepší chování sítě. [27]

### 2.8.4 Velikost dávky

Velikosti dávky (batch) je počet vzorků, které zároveň prochází neuronovou sítí. Pokud je velikost dávky vyšší, je lépe využita paměť počítače, gradienty jsou plynulé a učení je tak efektivnější z hlediska využití výpočetní kapacity. Velikost dávky se často pohybuje v řádu několika málo desítek až po nižší stovky vzorků [21]. [22]



### 2.8.5 Počet epoch

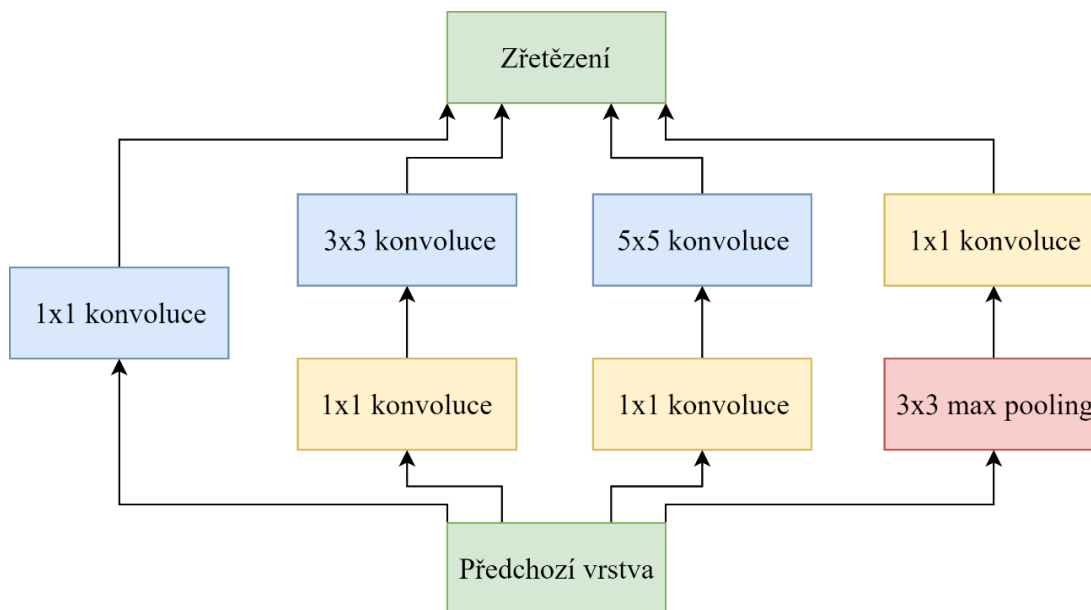
Epocha je jeden úplný přechod přes všechny trénovací vzorky v souboru dat. Se zvyšujícím se počtem epoch, se zvyšuje přesnost učení. Ovšem jakmile se použije určitý větší počet epoch, který je nutné zjistit experimentálně, model se začíná přeučovat. [14]

## 2.9 Neuronová síť Inception

Původní síť Inception vznikla v roce 2014 jakožto dílo výzkumníků z Google za účelem klasifikace obrazů. Hlavní myšlenkou této architektury je vytvoření hluboké neuronové sítě s vylepšeným využitím výpočetní kapacity tak, aby vůbec bylo možné takto hlubokou síť efektivně natrénovat.

V jiných architekturách se běžně využívá jedné konkrétní velikosti filtru v každé konvoluční vrstvě, kdy její velikost je nutné vyzkoušet a zjistit, jaký má tato volba následek při učení. Podobně se zkouší i umístění pooling vrstev – zda se umístí za každou konvoluční vrstvou, nebo každou druhou, třetí apod. Architektura Inception toto řeší tak, že je vytvořen inception modul, který obsahuje několik konvolučních vrstev o různých velikostech konvolučního jádra. Vrstvy s menším filtrem tedy dobře detekují malé detaily v obraze, zatímco vrstvy s větší velikostí filtru umí analyzovat větší celky obrazu. Výsledná architektura je tak tvořena konvoluční a pooling vrstvou na začátku, dále skládáním inception modulů a pooling vrstev a na konci jsou použity plně propojené vrstvy jako klasifikátory. [23,32]

Naivní inception modul se skládá ze tří různých konvolučních vrstev – o velikostech filtru 1x1, 3x3, 5x5 a max pooling vrstvy o velikosti filtru 3x3. Výstupy těchto vrstev jsou zřetězeny do jednoho výstupu, který je následně vstupem do další vrstvy. Problémem ovšem je, že při vyšší velikosti filtru u konvolučních vrstev se zvyšuje počet prováděných operací a tím výpočetní náročnost, proto je před vrstvami s velikostí filtru 3x3 a 5x5 provedena redukce dimenzí, což sníží následnou výpočetní náročnost a vznikne tak inception modul s redukcí dimenzí, jehož schéma je zobrazeno na Obr. 2.10. Tohoto je dosaženo přidáním 1x1 konvoluční vrstvy (také nazývané bottleneck vrstva), která zachovává prostorové dimenze (výšku a šířku), ale snižuje hloubku, tedy počet kanálů. Také je bottleneck vrstva přidána po provedení 3x3 max pooling, protože pooling sám o sobě neredukuje počet kanálů. [23]



Obr. 2.10: Inception modul s redukcí dimenzí, převzato z [32]

Konečná architektura je tvořena sérií konvolučních a max pooling vrstev na začátku, následně bloky inception modulů s redukcí dimenzí – tedy 2 moduly následované max poolingem, 5 modulů následovaných max poolingem a 5 modulů, a na konci klasifikátorem tvořeným vrstvou GAP a plně propojenými vrstvami. Jde o architekturu, která je nazývána GoogLeNet (nebo také Inception v1) a jedná se o první generaci modelů založených na inception modulech. V roce 2014 vyhrála soutěž ImageNet Large Scale Visual Recognition Challenge a šlo o průlomovou architekturu. Od té doby již vzniklo několik dalších verzí architektur založených na inception modulech a to jak pro použití v oblasti 2D dat, tak 1D dat. [23,32]

## 2.10 Multiple-Instance Learning

Multiple-Instance Learning (MIL) je metoda částečně řízeného učení. Obrazy nebo signály, tedy vstupní data, se zde označují jako jeden koš (angl. bag), který je složen z jednotlivých instancí (konkrétních míst obrazu nebo signálu) a třídy klasifikace jsou přiřazeny globálně celému koši. Základní myšlenkou tedy je, že negativní koš neobsahuje ani jednu pozitivní instanci a pozitivní koš obsahuje alespoň jednu pozitivní instanci, i když existují různé MIL algoritmy, které tuto podmínku mění a pro klasifikování koše jakožto pozitivního je tedy např. nutné, aby obsahoval určitý vyšší počet pozitivních instancí. [33,34]

Hlavním úkolem MIL je učit model, který predikuje třídu koše (např. medicínskou diagnózu na základě záznamu EKG nebo snímků pořízených pomocí magnetické rezonance), a dalším úkolem je nalézt ty instance, na základě kterých došlo k rozhodnutí modelu zařadit koš do určité třídy (tedy v případě EKG by se mohlo jednat o úsek, kde

se vyskytuje arytmie a v oblasti snímkování pomocí magnetické rezonance např. o nádorovou tkáň). Tyto instance se označují jako klíčové. [33,35]

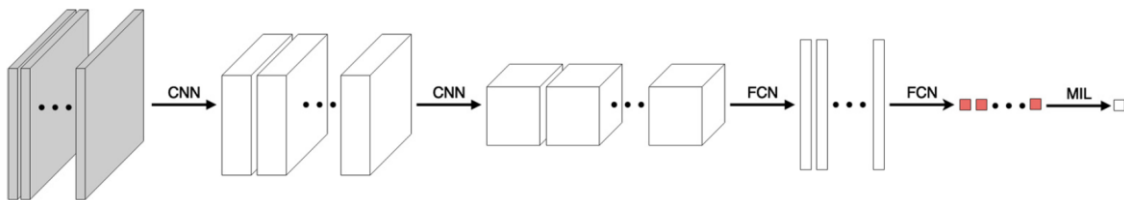
Vstupními daty k učení modelu jsou obrazy nebo signály (koše) mající globální třídu. Třídy jednotlivých instancí nejsou dostupné a to zejména proto, že by se jednalo o velmi časově náročnou práci vhodnou pouze pro kvalifikované pracovníky. Naučený model slouží pro přiřazení globální třídy k danému signálu a v ideálním případě také pro vyznačení místa signálu, které je zodpovědné za přiřazení dané třídy (např. v případě záznamu EKG, kterému byla přiřazena třída určité arytmie, by byl vyznačen úsek dané arytmie). [33,36]

Existují tři základní přístupy k problematice MIL – přístup založený na instancích, přístup založený na koši a přístup založený na vnoření (angl. embedding). [33]

### 2.10.1 Přístup založený na instancích

V tomto typu algoritmu se pracuje se skóre jednotlivých instancí. Dochází tedy k tréninku hluboké neuronové sítě, která má za účel spočítat skóre všech instancí (červené čtverce na Obr. 2.11 – schématu příkladu modelu přístupu založeném na instancích), která jsou následně využita v druhém kroku využívajícím MIL pooling vrstvu kombinující skóre všech instancí pro určení třídy celého koše.

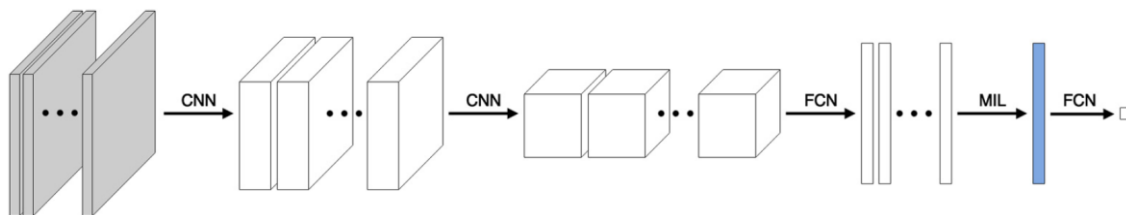
Výhodou tohoto přístupu je možnost zjištění klíčových instancí, což zvyšuje interpretovatelnost výsledků. V porovnání se dvěma dalšími přístupy ovšem dosahuje obecně horších výsledků při klasifikaci. [33]



Obr. 2.11: Schéma přístupu založeném na instancích [33]

### 2.10.2 Přístup založený na vnoření

Tento přístup je podobný výše zmíněnému typu algoritmu, kdy největší rozdíl spočívá v pořadí plně propojených vrstev a MIL pooling vrstvy. Nejdříve je provedeno vnoření instancí do nízkodimenzionálních reprezentací. Následně je využitím MIL pooling vrstvy provedeno zkombinování nízkodimenzionálních reprezentací instancí do jedné nízkodimenzionální reprezentace, neboli vnoření, koše (modrý obdélník na Obr. 2.12 – schématu příkladu modelu přístupu založeném na vnoření). Vstup do MIL pooling vrstvy je tedy vektor, oproti tomu u přístupu založeném na instancích je to skalár. Na konci modelu jsou umístěny plně propojené vrstvy, které provedou klasifikaci koše. [33,36]



Obr. 2.12: Schéma přístupu založeném na vnoření [33]

### 2.10.3 Přístup založený na koši

Tento přístup využívá rozdílů/podobností mezi jednotlivými koši a MIL problém tak transformuje na klasickou úlohu učení s učitelem. Tento způsob není možné přímo použít pro klasifikaci instancí. [33]

### 2.10.4 MIL pooling operátory

Existují různé typy MIL pooling operátorů v MIL pooling vrstvách přístupu založeného na instancích a přístupu založeného na vnoření, a jejich volba je velmi důležitou součástí řešení daného problému. V závislosti na přístupu tato vrstva slouží ke kombinování skóre jednotlivých instancí nebo vnoření instancí.

Max pooling operátor vybírá ze skóre instancí/nízkodimenzionálních reprezentací maximální hodnotu. Jedná se o operátor výhodný zejména v přístupu založeném na instancích – každá instance má své skóre a pomocí max pooling operátoru se vybírá ta s nejvyšším skóre, tedy klíčová instance.

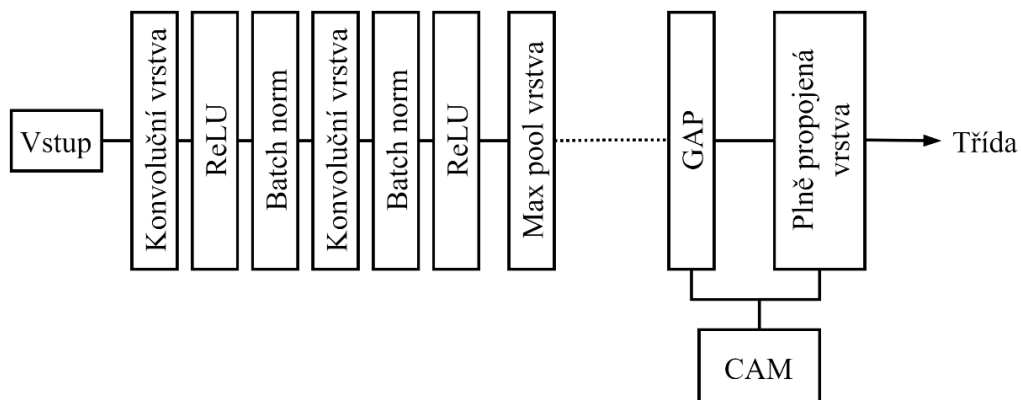
Průměrovací pooling operátor u přístupu založeném na instancích počítá průměr skóre jednotlivých instancí, u přístupu založeném na vnoření je výstupem průměrná nízkodimenzionální reprezentace ze všech nízkodimenzionálních reprezentací, které vznikly vnořením instancí.

Pooling operátory představené v předchozím odstavci jsou ovšem předem definované a není možné je trénovat, což snižuje flexibilitu modelů. Tyto nevýhody řeší pooling založený na pozornostním mechanismu, kdy takováto pooling „vrstva“ se tak skládá ze dvou plně propojených vrstev. V první vrstvě se využívá hyperbolické tangenty jako aktivační funkce (důvodem je symetričnost výstupů), ve druhé vrstvě SoftMax funkce, takže suma pozornostních vah je 1. Výhodou takovéto pooling vrstvy je, že je možné snadno interpretovat pozornostní váhy (popřípadě podobně využít gradient) – tedy čím vyšší tato váha je, tím vyšší důležitost má daná oblast. V případě váhy skutečné třídy je tak model schopen určit i klíčovou instanci. Příklad využití této funkce je v oblasti medicíny při diagnostice nádorového onemocnění z histologických vzorků – MIL model je schopen určit region vzorku, který obsahuje určité typy buněk. Grafické znázornění je možné pomocí teplotních map, které vizuálně zobrazují, kde se dle modelu nachází klíčová instance. [33]

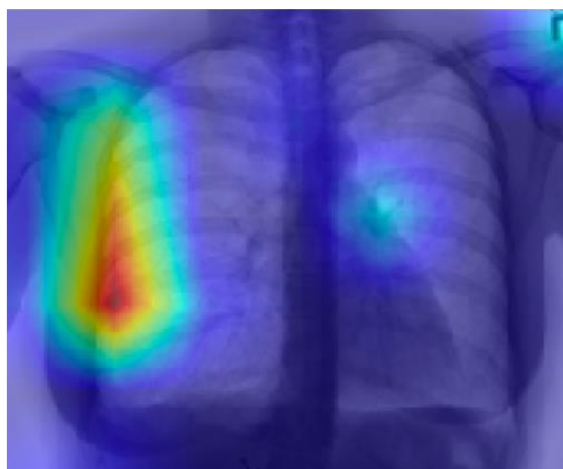
## 2.11 Class Activation Mapping

Class Activation Mapping (CAM, česky mapování třídivých aktivací) je metoda, která umožňuje označení regionu v signálu, který největší měrou přispívá ke klasifikaci signálu do určité třídy, tedy vlastně zobrazuje lokaci klíčové instance, jak je známo z MIL přístupů. CAM vytváří teplotní mapy korespondující s používanými daty, které ukazují vztah mezi výstupní predikcí a vstupem. Pozitivní třídivá aktivace znamená, že daný znak pozitivně přispívá ke klasifikaci do dané třídy a negativní aktivace označuje znak, který má nízkou důležitost pro klasifikaci do stejné třídy.

Při implementaci CAM je za poslední konvoluční vrstvu vložena vrstva GAP, která je následována plně propojenou vrstvou. Příklad takovéto architektury je zobrazen na Obr. 2.13. Výstupem GAP jsou prostorové průměry příznakových map, které jsou vynásobeny vahami tříd a následně jsou tato čísla sečtena a vznikne tak mapa aktivace tříd. Obr. 2.14 ukazuje snímek plic pořízený pomocí rentgenu, u kterých byla zjištěna diagnóza pneumotoraxu, spolu s teplotní mapou jakožto výstupem CAM zobrazující lokalizaci pneumotoraxu. [37]



Obr. 2.13: Schéma architektury umožňující CAM, převzato z [38]



Obr. 2.14: Teplotní mapa, pneumotorax [39]

### 2.11.1 Grad-CAM

Gradientem vážený CAM (Grad-CAM) je generalizací CAM a je tak možné tuto metodu využít na širší spektrum konvolučních neuronových sítí. Grad-CAM je vhodný i pro předučný model, jehož architektura zůstává zachována.

Prvním krokem tohoto přístupu je spočtení gradientu skóre pro určitou třídu s ohledem na aktivace příznakové mapy konvoluční vrstvy. Tyto gradienty jsou následně vloženy na vstup vrstvy GAP, čímž jsou získány váhy alfa důležitosti neuronu. Zmíněné váhy vyjadřují důležitost určité příznakové mapy pro danou třídu. V posledním kroku je provedena vážená kombinace aktivačních map a na výsledek je aplikována ReLU funkce (z toho důvodu, že objektem zájmu jsou pouze příznaky, které pozitivně ovlivňují třídu zájmu). [37,40]

## 2.12 Metriky pro zhodnocení kvality modelu

Pro hodnocení průběhu učení je možné použít různé metriky, které jsou zvoleny na základě řešeného problému.

Základním nástrojem pro evaluaci modelu je matice záměn (Obr. 2.15), která obsahuje tabulku predikcí a skutečných tříd. Je možné tedy zjistit, v kolika případech došlo ke správné klasifikaci a v kolika případech k různým typům chyb. Falešně pozitivní (FP = false positive) výsledek je ten, jehož predikce byla pozitivní, skutečná třída byla ovšem negativní. Označuje se také jako chyba typu I. Falešně negativní (FN = false negative) výsledek byl ten, jehož predikce byla negativní, ale skutečná třída byla pozitivní. Označuje se jako chyba typu II. Pokud je klasifikace provedena správně, jde o výsledky pravdivě pozitivní (TP = true positive) nebo pravdivě negativní (TN = true negative). [22]

		Skutečná třída	
		Pozitivní	Negativní
Predikce	Pozitivní	Pravdivě pozitivní	Falešně pozitivní
	Negativní	Falešně negativní	Pravdivě negativní

Obr. 2.15: Matice záměn

Vhodnou kombinací výsledků z matice záměn je možné určit mimo jiné senzitivitu (neboli také recall) a specificitu. Senzitivita kvantifikuje, jak dobře model detekuje pravdivě pozitivní případy a specificita udává to stejné ale pro pravdivě negativní případy. [22]

$$\text{Senzitivita} = \frac{TP}{TP+FN} \quad (2.2)$$

$$\text{Specifická} = \frac{TN}{TN+FP} \quad (2.3)$$

Přesnost (angl. accuracy) označuje podíl vzorků, které byly správně klasifikovány (tedy jako pravdivě pozitivní a pravdivě negativní). Použití výpočtu přesnosti není vhodné pro nevyvážené datasety – např. pokud by data obsahovala vzorky tříd v poměru 9:1 a došlo by ke správné klasifikaci pouze vzorků z první třídy, stále by byla přesnost neúměrně vysoká tomu, že model neumí klasifikovat vzorky z druhé třídy. [22]

$$\text{Přesnost (acc)} = \frac{TP+TN}{TP+FN+FP+TN} \quad (2.4)$$

Přesnost (angl. precision) označuje podíl vzorků, které byly predikovány jako pozitivní a jejich skutečná třída opravdu pozitivní je. [22]

$$\text{Přesnost (prec)} = \frac{TP}{TP+FP} \quad (2.5)$$

F skóre kombinuje dohromady dvě metriky a to přesnost (precision) a senzitivitu. Při řešení určitých problémů může být větší důraz kladen na vysokou přesnost, při řešení jiných problémů to může být právě vysoká senzitivita, proto je možné určit, jakou váhu bude mít v F skóre přesnost a jakou senzitivita, a metriku si tak nastavit dle konkrétních kritérií.

Příkladem je F1 skóre, které je harmonickým průměrem přesnosti a senzitivity a je využíváno, když je přikládána stejná důležitost oběma zahrnutým metrikám. [22,41]

$$F1 = \frac{2 \cdot TP}{2 \cdot TP + FP + FN} \quad (2.6)$$

### 3. NAVRŽENÝ PROGRAM

Program pro detekci abnormálních EKG segmentů, konkrétně komorových extrasystol, byl navržen v programovacím jazyce Python (verze 3.7.10) použitím prostředí Google Colab Pro. V tomto prostředí je dostupných 27,3 GB RAM a uživateli nemusí vždy být přiřazena stejná grafická karta, v naprosté většině běhů jde ale o kartu NVIDIA Tesla P100 16GB, méně často o NVIDIA Tesla V100 SXM2 16GB. Pro Bayesovskou optimalizaci hyperparametrů byl použit počítač s procesorem AMD Ryzen 5 (6 jader/12 vláken) a grafickou kartou NVIDIA GeForce RTX 2080 Ti.

Vytvořený program využívá zejména následujících knihoven:

- tsai (verze 0.2.17) – jedná se o knihovnu pro hluboké učení se zaměřením na moderní techniky pro klasifikaci 1D signálů (časových řad). Je založena na knihovnách Pytorch a fastai. [42]
- pandas (verze 1.1.5) – nástroj pro analýzu dat a manipulaci s nimi. [43]
- NumPy (verze 1.19.5) – slouží pro práci s vektory nebo maticemi, obsahuje matematické funkce. [44]
- BayesianOptimization – implementace Bayesovské optimalizace pro Python. [30]
- a další knihovny a balíčky, které jsou použity pouze ojediněle.

#### 3.1 Data

Data pro účely diplomové práce pochází z firmy BTL Medical a byla získána prostřednictvím vedoucího práce Ing. Jakuba Hejče. Jedná se o 6877 záznamů 12svodového EKG ve formátu json, kdy každý záznam má dobu trvání 10 sekund. Signály byly vzorkovány frekvencí 500 Hz a nebyly předem filtrovány.

Záznamy obsahují anotace, které vznikly konsenzem tří hodnotitelů z Ústavu biomedicínského inženýrství FEKT VUT a nesou informaci o tom, jaká diagnóza byla přiřazena danému signálu. K jednomu záznamu může být přiřazena i více než jedna třída, což je zřejmé i z Tab. 3.1, která zobrazuje třídy jednotlivých signálů a jejich počet v rámci celého datasetu. Je možné pozorovat, že některé třídy obsahují pouze malé množství záznamů, které tak nemusí být dostatečné pro kvalitní naučení modelu a následnou klasifikaci dat z testovací množiny. Skutečnou třídou zájmu (komorové extrasystoly, tedy PVC) bylo označeno celkem 412 záznamů.



Tab. 3.1: Třídy vyskytující se v datech a jejich počty

Třída	Počet
Sinusový rytmus	5956
EKG se šumem	2167
Tachykardie	1045
Infarkt myokardu	771
Bradykardie	714
Změny STT	669
Fibrilace síní	586
Síňové extrasystoly	418
Komorové extrasystoly	412
Blokáda levého Tawarova raménka	262
AV blokáda 1. stupně	258
Blokáda pravého Tawarova raménka	179
Supraventrikulární rytmus	179
Stimulace komor	96
Flutter síní	44
Stimulace síní	15
Komorová preexcitace	10
Síňový rytmus	8
AV blokáda 2. stupně	6
AV blokáda 3. stupně	6
Nespecifická intraventrikulární blokáda	2
Komorový rytmus	2
SA blokáda	1
Komorová trigeminie	1

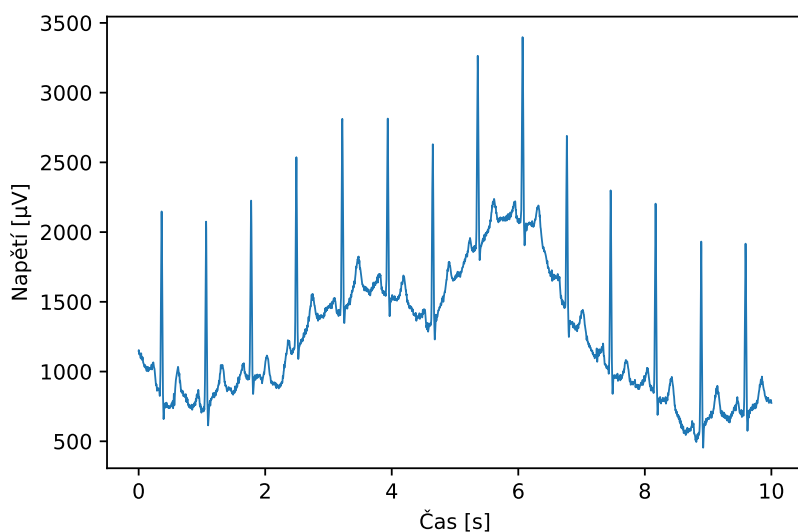
## 3.2 Předzpracování dat

Aby bylo možné využít záznamy EKG při učení modelu hluboké neuronové sítě, je nutné je nejdříve vhodně předzpracovat.

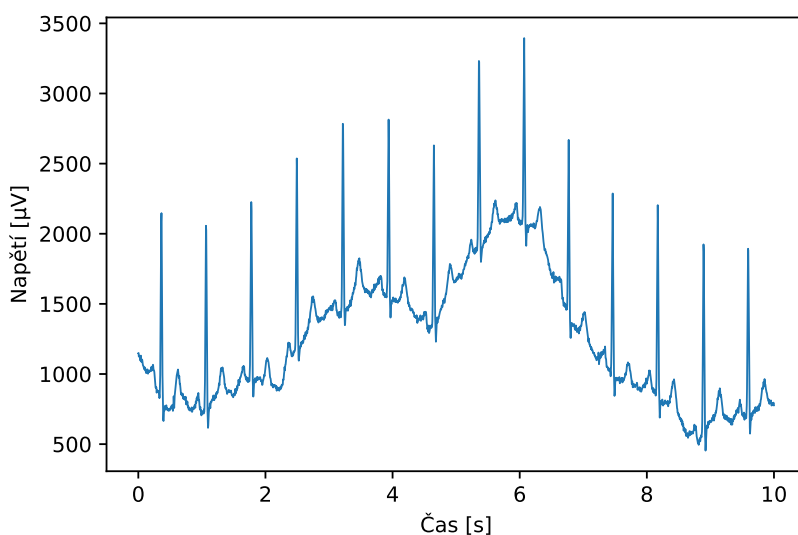
Prvním krokem předzpracování bylo uložení dat do vhodného formátu. Původním formátem byl json, kdy jeden záznam (signál a další informace včetně anotací) tvořil jeden json soubor. Z každého json souboru byl tedy postupně extrahován samotný signál (12 svodů), diagnózy (skutečné třídy) a id záznamu a tyto informace byly uloženy do datové struktury DataFrame, která je vhodná pro další zpracování pomocí využívané knihovny Pandas.

### 3.2.1 Decimace

Dalším krokem bylo podvzorkování (decimace) signálů. Nejdůležitější informace nese EKG v pásmu do přibližně 100 Hz [45], a proto bylo provedeno podvzorkování signálů z původních 500 Hz na vzorkovací frekvenci 250 Hz, aby byla práce s daty rychlejší a při tom nedošlo ke ztrátě užitečné informace. Obr. 3.1 zobrazuje signál (jeden svod) před decimací a Obr. 3.2 signál po decimaci a je možné zhodnotit, že při podvzorkování nedošlo k odstranění důležité složky EKG signálu, jelikož mezi signály na obrázcích není zřetelná změna.



Obr. 3.1: Signál (jeden svod) před decimací



Obr. 3.2: Signál (jeden svod) po decimaci

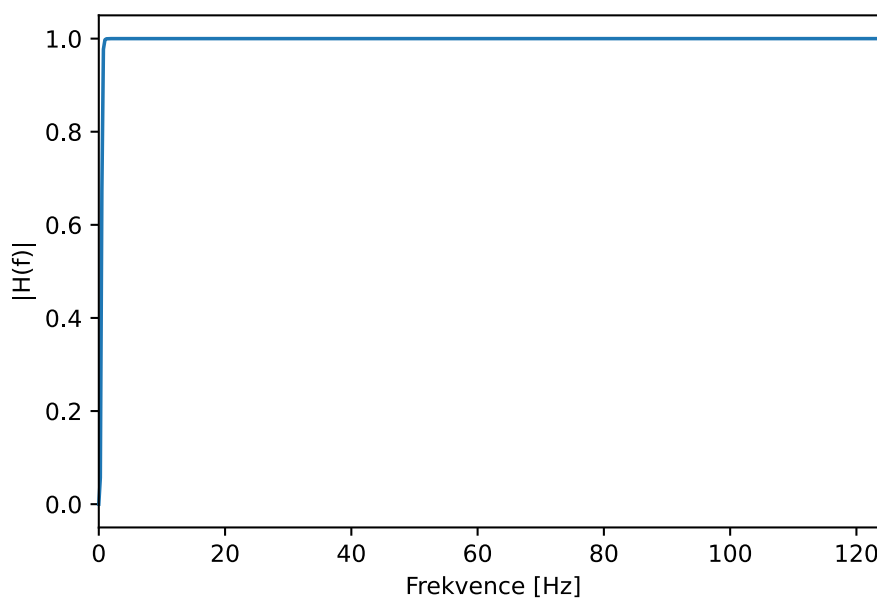
### 3.2.2 Filtrace

Jakmile došlo k decimaci signálu, byla provedena jeho základní filtrace pomocí Butterworthových filtrů 4. řádu. Butterworthův filtr je jedním z nejpoužívanějších filtrů v oboru, a to z důvodu jeho ploché amplitudové charakteristiky v propustném pásmu a také plynulé fázové odpovědi v propustném pásmu. [46]

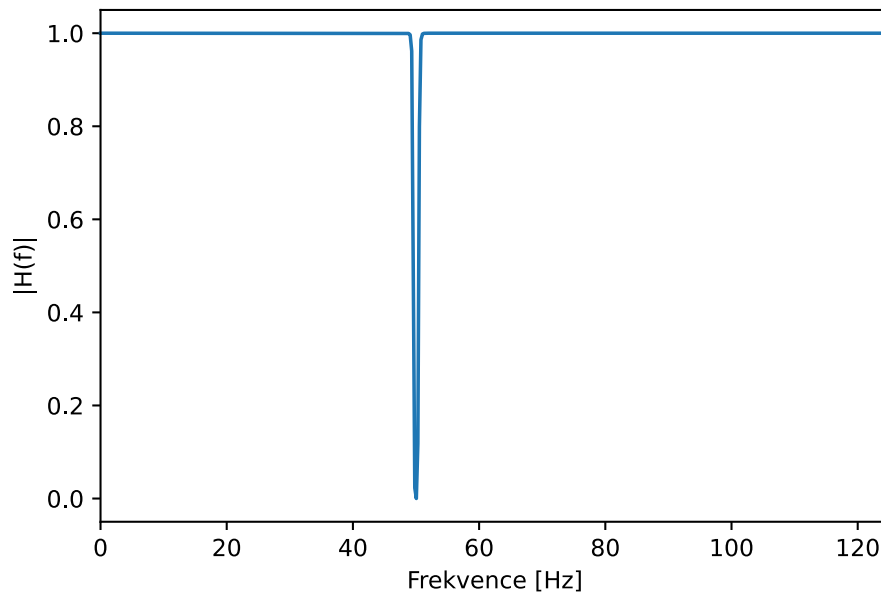
Drift nulové izolinie byl potlačen pomocí filtru typu horní propust s mezním kmitočtem 0,5 Hz, jehož frekvenční charakteristika je na Obr. 3.3. Síťový brum (50 Hz) byl potlačen pomocí filtru typu pásmová zádrž s mezními kmitočty 49,5 Hz a 50,5 Hz, jak je zřejmé i z jeho frekvenční charakteristiky zachycené na Obr. 3.4. Signál po filtraci (stejný jako na Obr. 3.2) je možné pozorovat na Obr. 3.5. Shrnutí nastavení filtrace je možné shlédnout v Tab. 3.2.

Tab. 3.2: Nastavení filtrace

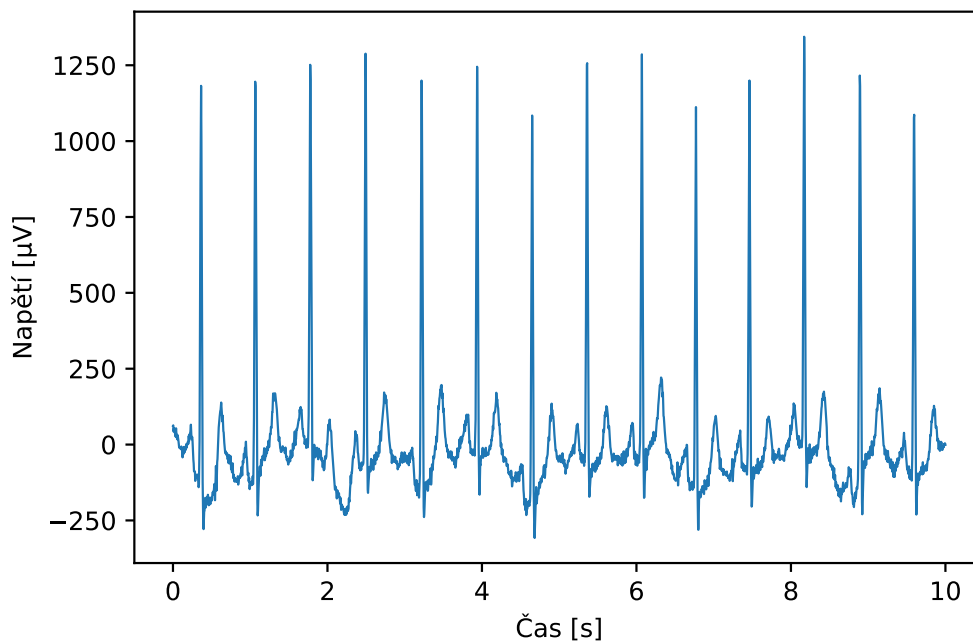
Filtrovaný šum	Typ filtru	Název filtru	Řád	Mezní kmitočty [Hz]
Drift nulové izolinie	Horní propust	Butterworthův	4	0,5
Síťový brum	Pásmová zádrž	Butterworthův	4	49,5 a 50,5



Obr. 3.3: Frekvenční charakteristika Butterworthova filtru (horní propust)

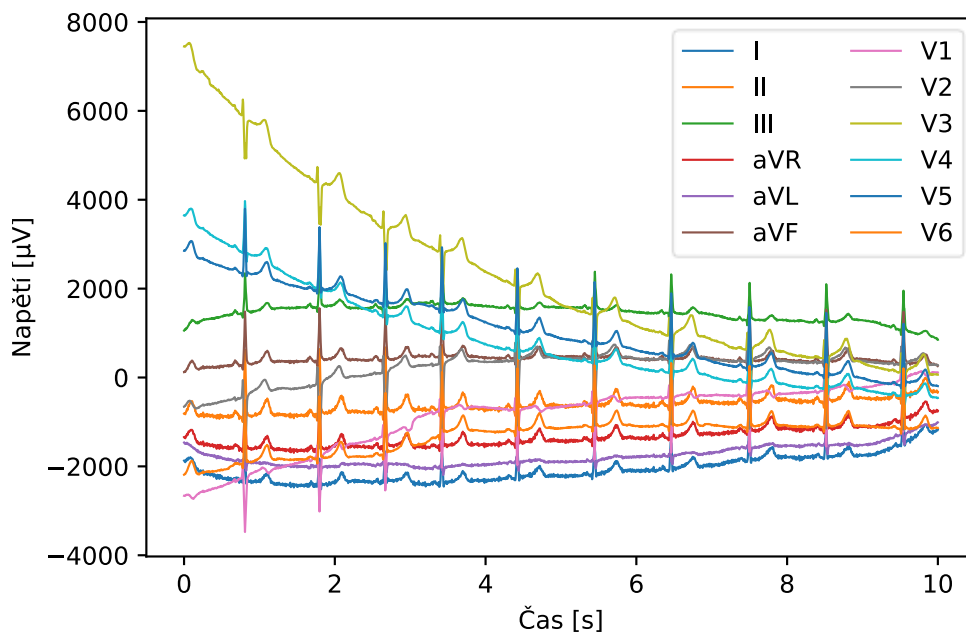


Obr. 3.4: Frekvenční charakteristika Butterworthova filtru (pásmová zádrž)

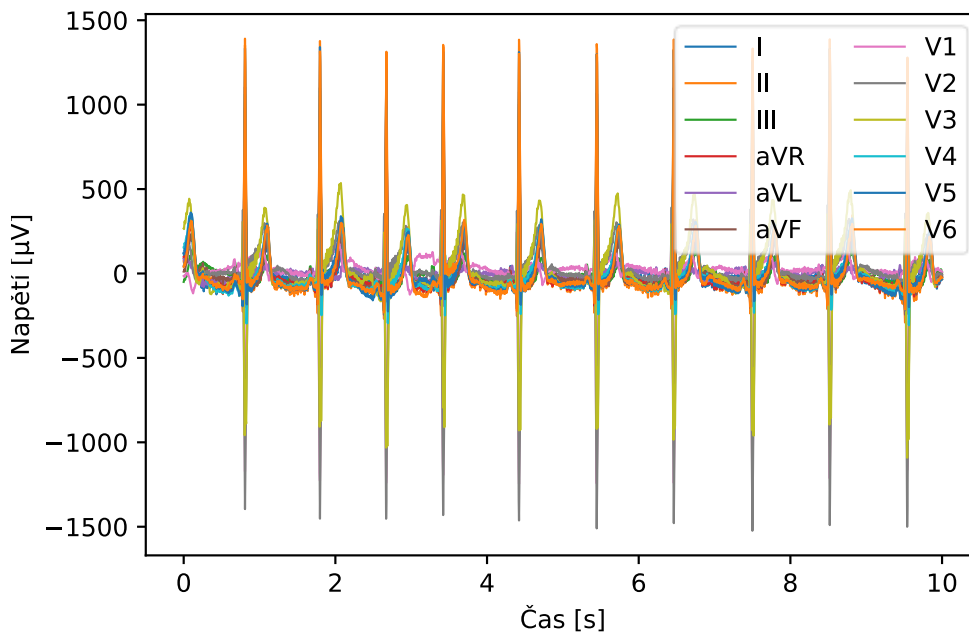


Obr. 3.5: Signál (jeden svod) po filtraci

Porovnání jednoho záznamu 12svodového EKG před filtrací a po filtraci je zobrazeno na Obr. 3.6 (před) a Obr. 3.7 (po).



Obr. 3.6: Signály (12 svodů) před filtrací



Obr. 3.7: Signály (12 svodů) po filtraci

### 3.2.3 Rozdělení dat do množin

Skutečnou třídou dle diagnózy jsou označeny signály, ke kterým je přiřazena pouze jedna třída z klasifikovaných tříd (mimo třídu ostatní) a zároveň jsou tyto diagnózy označeny jako diagnózy zájmu, všechny ostatní signály jsou označeny jako skutečná třída ostatní. Signály jsou následně rozděleny do trénovací, validační a testovací množiny. Rozdělení probíhá vždy tak, aby se v každé množině nacházel stejný poměr jednotlivých diagnóz, tedy přesněji skutečných tříd (stratifikované vzorkování) – např. v případě klasifikace do 2 tříd (klasifikace mezi PVC a ostatními) bude stejný poměr signálů s diagnózou PVC a signálů s ostatními diagnózami v trénovací, validační i testovací množině. Záznamy jsou před rozdělením do jednotlivých množin vždy promíchány.

Byly otestovány tři různé poměry rozdělení dat do množin vypsanych výše – šlo o běžně používané poměry a to:

- 70 % signálů trénovací množina, 15 % signálů validační množina a 15 % signálů testovací množina
- 80 % signálů trénovací množina, 10 % signálů validační množina a 10 % signálů testovací množina
- 60 % signálů trénovací množina, 20 % signálů validační množina a 20 % signálů testovací množina

Pro otestování všech rozdělení byl použit model popsany v kapitole 3.5.1 s použitím hyperparametrů z Tab. 4.4 a pro porovnání bylo použito F1 skóre klasifikace PVC u validační množiny (průměr ze tří běhů). Byla provedena klasifikace do 2 tříd (PVC, ostatní). V Tab. 3.3 jsou zobrazeny výsledky, ze kterých vyplývá, že nejlepšími výsledky klasifikace PVC model dosahuje při rozdělení dat v poměru 80 % signálů v trénovací množině, 10 % signálů ve validační množině a 10 % signálů v testovací množině.

Tab. 3.3: Výsledky testování poměrů rozdělení dat

Rozdělení	F1 skóre
70 : 15 : 15	0,87
80 : 10 : 10	0,90
60 : 20 : 20	0,83

Po rozdělení dat do jednotlivých množin je provedena jejich normalizace, a to pomocí průměru a směrodatné odchylky pouze trénovací množiny.

### 3.3 Řešení nevyvážených dat a jejich augmentace

Jak je možné odvodit z Tab. 3.1 v kapitole 3.1, data použitá v této práci nemají rovnoměrně rozdělené počty signálů v jednotlivých třídách, kdy komorových extrasystol, na jejichž klasifikaci je zaměřena tato práce, je 412 z celkových 6877 signálů. Tento jev je běžný u medicínských dat, protože incidence problematických jevů (např. určitého typu arytmie) je nižší než fyziologických záznamů.

Většina klasifikačních algoritmů ovšem pracuje s předpokladem, že jednotlivé třídy jsou vyvážené a pokud nejsou, může docházet k upřednostňování majoritní třídy při klasifikaci a modelu se snižuje schopnost generalizace. Pro odstranění, nebo alespoň zmenšení, problému existuje několik různých metod.

Běžnými metodami pro snížení nevyváženosti 1D dat jsou oversampling a undersampling dat, tedy kopírování jednotlivých signálů méně zastoupené třídy pro snížení rozdílu mezi počtem signálů v majoritní třídě a minoritní třídě (třídách), respektive odstraňování signálů z více zastoupené třídy.

Používané jsou také metody, které data přímo augmentují (tedy vytváří nové jedinečné signály ze signálů již existujících), nejen mění jejich relativní zastoupení v datasetu. Jedná se o využití generativních kompetitivních sítí nebo techniky SMOTE (synthetic minority oversampling technique = syntetická technika minoritního vzorkování). Pokud jsou klasifikována obrazová data, jsou používány různé transformace typu rotace nebo škálování obrazu, v případě 1D signálů, potažmo EKG, ovšem takovéto transformace nedávají smysl nebo jejich vliv není dostatečně prozkoumán.

Přímo pro EKG existují speciální metody augmentace, ale ne všechny tyto techniky jsou vhodné pro použití při klasifikaci jakýchkoliv diagnóz. Jednou z technik je např. window slicing, kdy se záznam EKG dělí na menší kousky, tak zvané snippets, se kterými se následně nakládá jako s novými záznamy. Předpokládá se ovšem, že všechny menší nastříhané úseky patří do stejné třídy jako původní záznam, což není pravda u arytmií, které se vyskytují velmi často jednotlivě (např. právě PVC). [47]

V rámci této práce byla pro řešení nevyvážení dat použita kombinace oversamplingu a undersamplingu. Tedy zvyšoval se počet signálů v méně zastoupených třídách a zároveň se snižoval počet signálů ve více zastoupených třídách. Aby bylo možné určit, jaká nejlepší kombinace těchto dvou úkonů je ideální (= klasifikace PVC tak dosahuje nejlepších výsledků), byly určeny dva hyperparametry, jejichž vhodné hodnoty jsou hledány pomocí Bayesovské optimalizace. Jeden z hyperparametrů určuje, kolikrát může být zduplikována nejméně zastoupená třída v datasetu (max\_duplicate). Druhý hyperparametr říká, kolikrát maximálně větší mohou být více zastoupené třídy vůči té nejméně zastoupené (max\_extra, tedy přesněji součet max\_duplicate a max\_extra). Na testovací a validační množinu dat nebyl oversampling a undersampling aplikován.

### 3.3.1 Mixup

Augmentace dat proběhla pomocí metody mixup. Prostřednictvím této techniky jsou konstruovány nové vzorky

$$\begin{aligned}\tilde{x} &= \lambda x_i + (1 - \lambda)x_j, \\ \tilde{y} &= \lambda y_i + (1 - \lambda)y_j,\end{aligned}$$

kde  $x_i$  a  $x_j$  jsou původní signály,  $y_i$  a  $y_j$  původní one-hot zakódované třídy, a  $\lambda$  je v rozmezí 0 až 1. Jsou tedy vytvářeny nové vzorky pomocí lineární kombinace dvou náhodně vybraných původních vzorků a také k nim příslušející nové třídy, které jsou stejně tak kombinací dvou daných původních tříd. Hodnota  $\lambda$  pochází z distribuce  $\text{Beta}(\alpha, \alpha)$ , kde  $\alpha$  je hyperparametrem kontrolujícím sílu interpolace mezi dvěma vzorky, jehož hodnota je optimalizována pomocí Bayesovské optimalizace.

Samotná implementace metody mixup je poměrně jednoduchá a představuje pouze minimální zátěž na výpočetní kapacitu, přesto tento typ augmentace dat velmi dobře regularizuje model. [48]

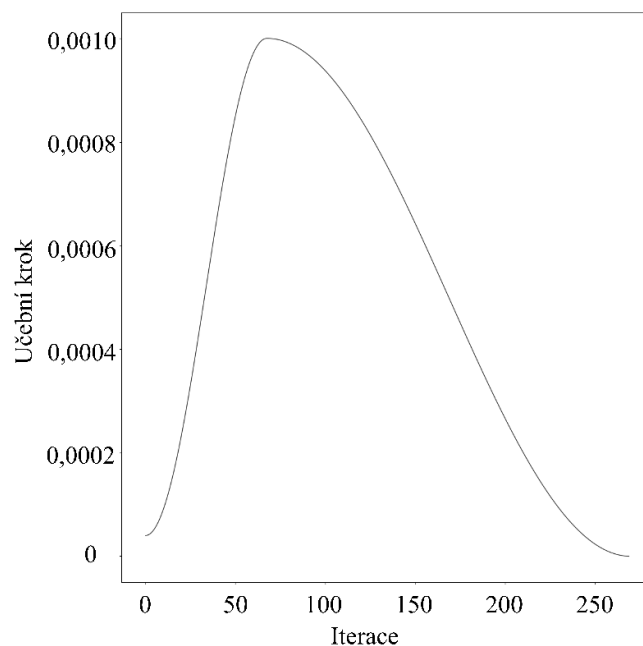
## 3.4 Učení modelu

Při trénování modelu byl použit učební algoritmus Adam, který je blíže popsán v kapitole 2.6. Dle literatury ([14,17,21]) se jedná o robustní algoritmus, který v sobě efektivně spojuje více optimalizačních technik. Nespornou výhodou je, že využívá exponenciálně zapomínané průměry k výpočtu jak gradientu, tak kvadrátu gradientu.

Jako ztrátová funkce byla zvolena křížová entropie, která je nejčastěji používanou ztrátovou funkcí v klasifikačních problémech a je detailně popsána v kapitole 2.5. [21]

Data jsou vždy při učení promíchána. Učení modelu probíhá v cyklech a epochách, kde je vždy určitý stejný počet epoch ve všech cyklech (počet epoch je hyperparametrem, jehož optimální hodnota je vyhledávána pomocí Bayesovské optimalizace). Tento typ průběhu učení je založen na článku [49], který představuje tzv. 1cycle přístup a řeší problematiku nastavení kroku učení, což je velmi důležitý parametr, který tento proces silně ovlivňuje (více v kapitole 2.8.2). Pokud je učební krok moc malý, může dojít k přeučení modelu. Vyšší učební kroky pomáhají model regularizovat, pokud je všem učební krok moc vysoký, trénování diverguje. Když je využit 1cycle přístup, dochází k tomu, že je nastaven určitý maximální učební krok (další z hyperparametrů). V průběhu učení se ovšem učební krok mění pro každou dávku – na začátku je učební krok malý, postupně se zvyšuje až na danou maximální hodnotu a následně znovu klesá. Zvyšování učebního kroku přibližně uprostřed cyklu regularizuje model a zabraňuje tak přeučení. Vývoj hodnoty učebního kroku během jednoho cyklu je zobrazen na Obr. 3.8. [49,50]





Obr. 3.8: Vývoj hodnoty učebního kroku za 1 cyklus

Jako hlavní metrika pro posouzení kvality učení bylo zvoleno F1 skóre klasifikace PVC, které kombinuje přesnost (precision) i senzitivitu. Také je vhodné pro nevyvážené datasey, u kterých může přesnost (accuracy) vyvolávat dojem přehnaně optimistických výsledků.

Model byl trénován za účelem klasifikace do 2 tříd (PVC, ostatní), do 3 tříd (PVC, sinusový rytmus, ostatní) a do 8 tříd (PVC, PAC, sinusový rytmus, fibrilace síní, tachykardie, bradykardie, infarkt myokardu a ostatní). Klasifikace do 8 tříd byla vybrána proto, aby bylo možné porovnat výsledky klasifikace do méně tříd (2 a 3) i více tříd (8) a zjistit, zda a jaký vliv toto má na metriky klasifikace PVC. Jako diagnózy vhodné pro klasifikaci do 8 tříd (7 tříd daných diagnóz a třídy ostatní) byly vybrány ty, které byly přiřazeny jako skutečná třída alespoň 400 záznamům, aby existoval dostatečný počet signálů k učení modelu i následné testování.

### 3.5 Výběr architektury

Pro výběr vhodné architektury bylo otestováno celkem 7 architektur založených na konvolučních neuronových sítích z knihovny tsai [42], kde je také možné nalézt detaily jednotlivých architektur. Byly vybrány na základě získaných teoretických i praktických znalostí tak, aby bylo dostupné porovnání jak jednoduché plně konvoluční sítě, reziduálních sítí, tak i pokročilejších architektur s prvky paměti (LSTM) nebo obsahujících inception bloky.

Jednalo se o následující architektury vhodné pro klasifikaci 1D signálů:

- FCN – plně konvoluční síť složená ze tří bloků: konvoluční vrstva + BatchNorm vrstva + ReLU aktivační funkce, filtry konvolučních vrstev jsou o velikosti 8, 5 a 3 a počet filtrů v jednotlivých vrstvách je 128, 256 a 128 [51]
- LSTM\_FCNet – jedná se o FCN jako výše s přidáním LSTM (long short-term memory) bloku [52]
- ResNet – využívá konvolučních bloků jako architektura FCN (jen s rozdílným počtem filtrů v konvolučních vrstvách a to 64, 128 a 128) ve formě reziduálních bloků, které jsou propojeny reziduálním spojením, kterým může přímo plynout gradient a síť tedy může být hlubší [51]
- Xresnet1d34 – upravená architektura ResNet z článku [53]
- ResCNN – tato architektura kombinuje ResNet a CNN, kdy je složena ze tří konvolučních bloků (počet filtrů v konvolučních vrstvách je 64) ve formě reziduálního bloku, který je následován třemi konvolučními vrstvami (počet filtrů je 128, 256, 128) [54]
- InceptionTime – využití inception modulů detailně popsanych v kapitole 2.9, více informací v kapitole 3.5.1 [55]
- Xception\_time – architektura odvozená z InceptionTime, kde jsou inception moduly nahrazeny hloubkovou separabilní konvolucí [56]

Jako metriky pro zhodnocení bylo využito přesnosti (accuracy) klasifikace ve validační množině (průměru ze tří učení) – jedná se pouze o porovnání výsledků mezi sebou. Toto testování (klasifikace do 2 tříd – PVC a ostatní) proběhlo s rozdělením do trénovací, validační a testovací množiny v poměru 70 : 15 : 15 a s hyperparametry uvedenými v Tab. 3.4. Tyto úvodní hyperparametry byly zvoleny na základě informací z teoretické části práce, popřípadě prostřednictvím praktických zkušeností získaných při tvorbě programu.

Tab. 3.4: Nastavení hyperparametrů pro výběr architektury

Hyperparametr	Hodnota
Maximální učební krok	0,001
Dávka (trénování)	32
Počet epoch	30
Počet cyklů	1
Max_extra	2
Max_duplicate	2
Mixup (alfa)	0,2

Výsledky testů různých architektur jsou zapsány v Tab. 3.5 včetně délky učení a počtu učících se parametrů (vah). Architektury jsou seřazeny sestupně od té s nejlepší přesností po tu s nejhorsí přesností klasifikace. Dle výsledků je tedy možné usoudit, že nejvyšší přesnost klasifikace má architektura InceptionTime, která také byla použita pro další práci. Nejvyšší počet učících se parametrů má architektura Xresnet1d34 (více než 7 milionů), což má následek vysokou výpočetní náročnost. Zároveň výpočetní náročnost ale není ovlivněna jen počtem učících se parametrů, ale rovněž samotnou stavbou architektury. Proto např. doba učení Xception\_time (147 s) je nejvyšší ze všech vyzkoušených architektur i přes nižší počet parametrů. Vybraná architektura InceptionTime má 457 218 učících se parametrů a doba učení je 120 s.

Tab. 3.5: Výsledky učení za použití různých architektur

Architektura	Přesnost	Počet parametrů	Doba učení [s]
InceptionTime	0,83	457 218	120
ResCNN	0,78	261 891	57
ResNet	0,78	484 098	87
Xresnet1d34	0,77	7 229 314	93
FCN	0,76	274 178	47
Xception_time	0,76	400 980	147
LSTM_FC	0,69	1 315 178	50

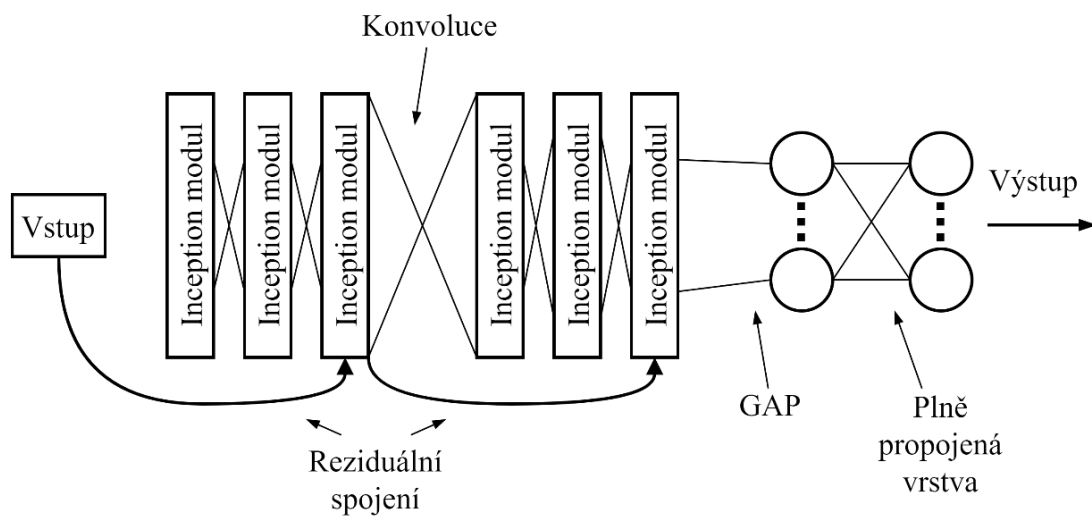
### 3.5.1 InceptionTime

Architektura InceptionTime (schéma na Obr. 3.9) je založena na inception architektuře obecně popsané v kapitole 2.9 a je vhodná přímo pro 1D signály (časové řady), kdy pomocí různě dlouhých filtrů v konvolučních vrstvách umožňuje extrahovat příznaky z kratších i delších časových úseků.

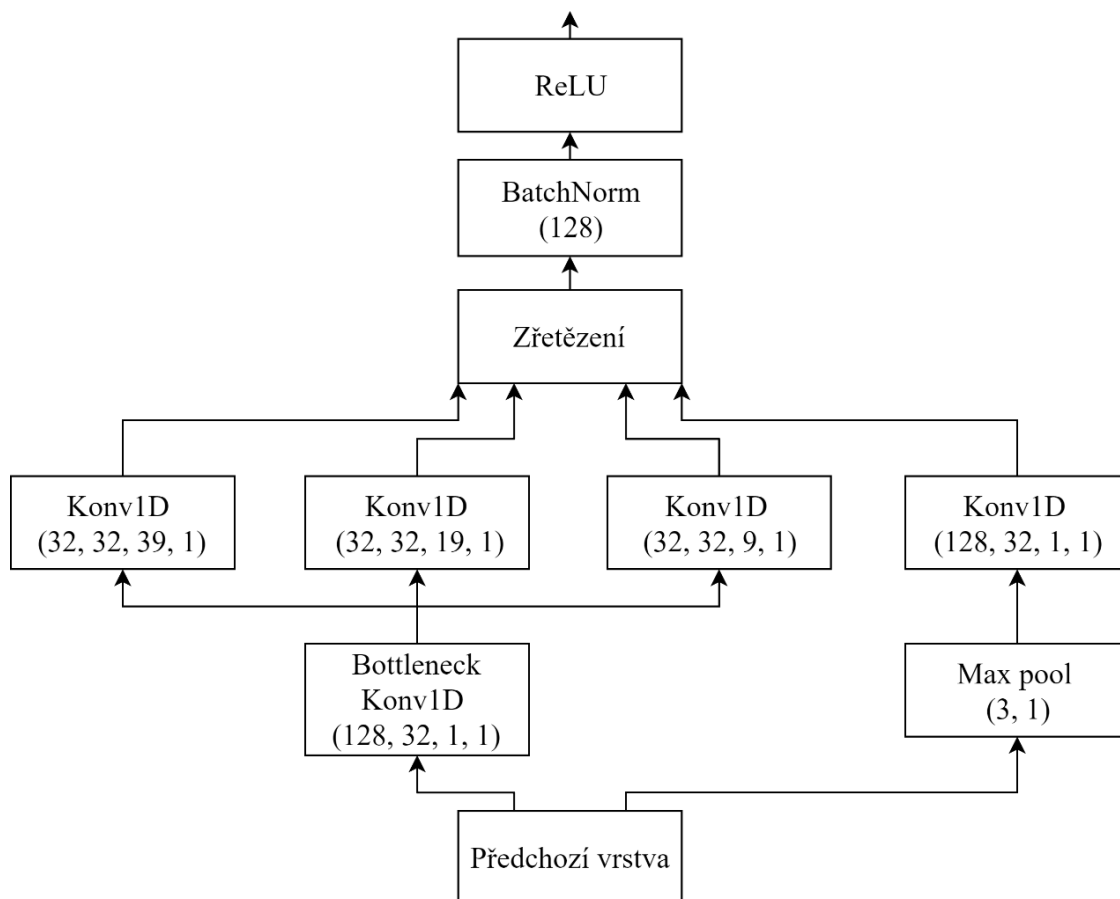
Model je vytvořen ze dvou reziduálních bloků, které jsou každý složeny ze tří inception modulů, kdy každý modul obsahuje tři různé konvoluční vrstvy s velikostí filtrů 9, 19 a 39. Vstup každého reziduálního bloku je převeden přes reziduální spojení na vstup dalšího bloku, čímž se potlačuje problém mizejícího gradientu.

Schéma inception modulu (všech šest modulů je stejných, až na počet vstupních kanálů bottleneck vrstvy prvního modulu, který je roven počtu svodů, tedy 12) je na Obr. 3.10. U každé konvoluční vrstvy je v závorkách uveden (v tomto pořadí): počet vstupních kanálů, počet výstupních kanálů, velikost filtru a krok filtru. U max poolingové vrstvy jde o velikost filtru a krok filtru.

Následně po reziduálních blocích je umístěna GAP vrstva a model je zakončen plně propojenou vrstvou, do které proudí vstup o velikosti 128 a výstupní počet neuronů je roven počtu tříd, do kterých je prováděna klasifikace. [55]



Obr. 3.9: Schéma architektury InceptionTime, převzato z [55]



Obr. 3.10: Schéma Inception modulu

### 3.6 Hyperparametry a jejich optimalizace

Pro zajištění co nejlepších výsledků učení modelu je nutné optimalizovat hyperparametry, jejichž seznam a rozsah prohledávání je uveden v Tab. 3.6. Cílem je najít takovou kombinaci hyperparametrů, aby bylo F1 skóre klasifikace PVC validační množiny co nejvyšší (jedná se o průměr F1 skóre ze tří učení se stejnými hyperparametry). Použitím hodnot maximální hranice rozsahu ve formě desetinného čísla u dávky, počtu epoch, max\_extra a max\_duplicate se zaručí, že všechny hodnoty z rozsahu mají stejnou pravděpodobnost zvolení (čísla jsou vždy před učením převáděna na celé číslo pomocí příkazu int). Na validační množinu dat byl aplikován oversampling a undersampling. Počet cyklů byl ustálen na 15, protože algoritmus vybírá nejlepší F1 skóre ze všech cyklů a není tedy třeba toto číslo optimalizovat. Při finálním učení modelu je uložen model s nejlepším F1 skóre PVC ve validační množině a tento model je následně použit pro klasifikaci vzorků testovací množiny.

Velikost prohledávaného prostoru byla zvolena na základě praktických zkušeností i znalostí z literatury, stejně tak jako nutnosti omezit prohledávaný prostor kvůli dostupnosti výpočetní kapacity. Rozdělení dat do trénovací, validační a testovací množiny proběhlo v poměru 70 : 15 : 15, jelikož testování různých poměrů rozdělení proběhlo až po Bayesovské optimalizaci.

Optimalizace hyperparametrů byla provedena pomocí Bayesovské optimalizace za použití knihovny BayesianOptimization [30] a to pro klasifikaci do 2 tříd (PVC, ostatní), do 3 tříd (PVC, sinusový rytmus, ostatní) a do 8 tříd (PVC, PAC, sinusový rytmus, fibrilace síní, tachykardie, bradykardie, infarkt myokardu a ostatní). Bylo zvoleno, že se provede 40 kroků optimalizace a bude 10 výchozích bodů prohledávání. Metoda Bayesovské optimalizace byla vybrána, protože vykazuje efektivitu ve smyslu počtu trénování modelu, který je nutný pro nalezení optimálního řešení [30,31].

Tab. 3.6: Hyperparametry a jejich rozsahy, Bayesovská optimalizace

Hyperparametr	Rozsah prohledávání
Maximální učební krok	0,0001-0,005
Dávka (trénování)	8-64,999
Počet epoch	1-12,999
Max_extra	0-4,999
Max_duplicate	1-4,999
Mixup (alfa)	0-1

### 3.7 Lokalizace PVC

Pro získání oblastí signálu, kde se vyskytuje PVC (tedy klíčová instance, jak je známo z přístupu MIL), byla použita metoda Grad-CAM blíže popsaná v kapitole 2.11.1. Aktivace a gradienty na výpočet Grad-CAM pochází z poslední vrstvy sítě před vrstvou GAP, tedy z výstupu celého inception bloku po průchodu ReLU aktivační funkcí. Jedná se o poslední možné místo, odkud je možné aktivace čerpat, model je plně naučen a aktivace tak nesou největší množství užitečné informace.

Tímto je možné nalézt místo signálu, na které se daná síť při klasifikaci zaměřuje, v případě této práce ovšem není možné posouzení přesnosti této metody přímo z dodaných informací v databázi, jelikož dostupná data nejsou podrobně anotována. Proto byla pro zhodnocení výsledků oslovena odbornice v oboru a výsledky jsou tedy jak ve formě ukázky lokalizace PVC na obrázcích, tak podrobnějšího zhodnocení úspěšnosti detekce PVC.

V programovém řešení této práce jsou nejdříve spočteny a vizualizovány mapy vystupující z metody Grad-CAM – tyto mapy existují jako vektor pro každou třídu a čím vyšší hodnotu má určitá hodnota ve vektoru, tím více se na toto místo zaměřuje model při učení, a tedy následně klasifikaci. Předpokladem tedy je, že tato místa jsou lokací klasifikovaných arytmií.

Nabízí se tedy, aby bylo zjištěno, zda jsou místa s nejvyšší hodnotou v Grad-CAM mapě opravdu důvodem, proč se model rozhodne klasifikovat záznam do dané třídy. Byly tedy vytvořeny Grad-CAM mapy třídy PVC validační množiny dat třídy PVC a byla zjištěna jejich maximální hodnota, pomocí níž se určil práh v rozmezí 0 až 110 % této hodnoty. Mapa každého záznamu byla následně prahována a v mapě tak zůstaly pouze hodnoty vyšší než daný práh. Toto bylo provedeno pro každý práh vždy u všech záznamů PVC a byla vypočtena přesnost (precision), senzitivita a F1 skóre klasifikace pouze pomocí prahování Grad-CAM mapy (tedy signály s mapou, která obsahuje hodnoty vyšší než práh, byly zařazeny do třídy PVC). Předpokládáno je, že pokud dosahuje klasifikace i tímto způsobem vysokých hodnot metrik, tak se model zaměřuje na správná místa signálu, která jsou tak klíčovými instancemi pro klasifikaci do dané třídy (v těchto místech je vysoká hodnota vážené aktivace).

Výstupem této části programu jsou prahované aktivační mapy záznamů skutečné třídy PVC zobrazující místo v signálu, na které se zaměřuje model při jejich klasifikaci (klíčové instance).

## 4. VÝSLEDKY

### 4.1 Bayesovská optimalizace hyperparametrů

#### 4.1.1 Klasifikace do 2 tříd

Výsledky Bayesovské optimalizace hyperparametrů pro klasifikaci do 2 tříd (PVC a ostatní) jsou uvedeny v Tab. 4.1. Tabulka zahrnuje kombinace hyperparametrů, za jejichž použití bylo dosaženo nejvyšších pěti F1 skóre klasifikace PVC na validační množině dat. Je možné zhodnotit, že každá pětice hodnot vždy stejného hyperparametru je poměrně různorodá a není tak možné konstatovat, že by např. nízký počet epoch vedl vždy k lepším výsledkům nebo bylo výhodné použít vyšší mixup alfa, a to až na hodnotu max\_extra, která se pohybuje na spodní hranici prohledávaného rozsahu. Maximální F1 skóre klasifikace PVC, kterého bylo dosaženo při optimalizaci hyperparametrů pro klasifikaci do 2 tříd, je nejvyšší (0,860) ze všech tří provedených optimalizací (pro klasifikaci do 2,3 a 8 tříd). Tato Bayesovská optimalizace trvala 19 hodin.

Tab. 4.1: Výsledek Bayesovské optimalizace, klasifikace do 2 tříd

	Maximální učební krok	Dávka (trénování)	Počet epoch	Max_extra	Max_duplicate	Mixup (alfa)	F1 skóre
1	0,0040	28	4	0	3	0,07	0,860
2	0,0016	53	12	0	1	0,41	0,829
3	0,0023	52	9	0	3	0,57	0,828
4	0,0022	46	6	1	3	0,24	0,818
5	0,0036	26	4	1	2	0,12	0,813

#### 4.1.2 Klasifikace do 3 tříd

Výsledky Bayesovské optimalizace hyperparametrů pro klasifikaci do 3 tříd (PVC, sinusový rytmus a ostatní) jsou uvedeny v Tab. 4.2. Tabulka zahrnuje kombinace hyperparametrů, za jejichž použití bylo dosaženo nejvyšších pěti F1 skóre klasifikace PVC na validační množině dat. Z dat je možné vyčíst, že pro klasifikaci do 3 tříd je výhodné použití trénovací dávky kolem 27 vzorků, spíše nižšího počtu epoch v cyklu, nulové hodnoty max\_extra a střední až vyšší hodnoty max\_duplicate. Alfa metody mixup se pohybuje zejména v jednotkách setin. Maximální F1 skóre klasifikace PVC (0,846) je nižší než maximální F1 skóre klasifikace PVC (0,860), kterého bylo dosaženo optimalizováním hyperparametrů pro klasifikaci do 2 tříd. Tato Bayesovská optimalizace trvala 20 hodin.

Tab. 4.2: Výsledek Bayesovské optimalizace, klasifikace do 3 tříd

	Maximální učební krok	Dávka (trénování)	Počet epoch	Max_extra	Max_duplicate	Mixup (alfa)	F1 skóre
1	0,0043	27	4	0	3	0,02	0,846
2	0,0011	27	3	0	2	0,01	0,837
3	0,0040	28	4	0	3	0,07	0,831
4	0,0026	27	4	0	3	0,01	0,830
5	0,0033	27	4	0	3	0,10	0,816

#### 4.1.3 Klasifikace do 8 tříd

Výsledky Bayesovské optimalizace hyperparametrů pro klasifikaci do 8 tříd (PVC, PAC, sinusový rytmus, fibrilace síní, tachykardie, bradykardie, infarkt myokardu a ostatní) jsou uvedeny v Tab. 4.3. Tabulka zahrnuje kombinace hyperparametrů, za jejichž použití bylo dosaženo nejvyšších pěti F1 skóre klasifikace PVC na validační množině dat. Oproti výsledkům Bayesovské optimalizace hyperparametrů pro klasifikaci do 3 tříd je možné určit, že v případě optimálních hyperparametrů pro klasifikaci do 8 tříd je výhodný spíše vyšší počet epoch v cyklu a až na jednu výjimku i vysoké alfa metody mixup. Rovněž hyperparametr max\_extra je ustálen na hodnotě 1 a max\_duplicate dosahuje nižších hodnot v porovnání s předcházejícími výsledky optimalizace. F1 skóre klasifikace PVC optimální kombinace hyperparametrů dosahuje nejnižší hodnoty ze všech tří typů klasifikací a to 0,79. Tato Bayesovská optimalizace trvala 25 hodin.

Tab. 4.3: Výsledek Bayesovské optimalizace, klasifikace do 8 tříd

	Maximální učební krok	Dávka (trénování)	Počet epoch	Max_extra	Max_duplicate	Mixup (alfa)	F1 skóre
1	0,0050	32	12	1	2	1,0	0,790
2	0,0050	40	9	1	2	1,0	0,780
3	0,0028	8	8	1	2	0,75	0,775
4	0,0037	13	9	1	1	0,21	0,774
5	0,0050	12	10	1	1	1,0	0,772



## 4.2 Klasifikace

Jakmile byla dokončena optimalizace hyperparametrů, byla vybrána jejich kombinace, při které dosahoval model nejlepších výsledků (nejvyšší F1 skóre PVC) při klasifikaci validační množiny dat do daného počtu tříd. Následně byl model naučen za použití daných hyperparametrů a rozdělení dat do trénovací, validační a testovací množiny v poměru 80 : 10 : 10, jelikož toto rozdělení dosahovalo nejlepších výsledků klasifikace PVC, viz kapitola 3.2.3. Model byl uložen při zachycené nejvyšší hodnotě F1 skóre PVC ve validační množině a následně byl použit pro klasifikaci dat testovací množiny.

Hlavní metrikou posuzující kvalitu výsledků klasifikace testovací množiny dat bylo F1 skóre – je spočteno F1 skóre pro každou skutečnou třídu a následně je nalezen průměr těchto skóre, tedy každé třídě je přiřkládána stejná důležitost bez ohledu na to, kolik vzorků obsahuje. Podobným způsobem byla spočtena i přesnost (precision) a senzitivita. Při hodnocení je ale nutno se dívat i na metriky jednotlivých tříd, jelikož schopnost modelu klasifikovat různé třídy se může silně lišit. Pokud bude dále v textu pojednáváno o přesnosti, jedná se o přesnost ve smyslu precision.

Jelikož v řešeném problému jde o klasifikaci srdečních arytmií, je možné považovat senzitivitu klasifikace arytmiie za důležitější metriku než přesnost – pokud je záznam falešně negativní, nebude zachycen pacient trpící srdečním onemocněním, což může být potenciálně nebezpečné.

### 4.2.1 Klasifikace do 2 tříd

Jako hyperparametry pro klasifikaci do 2 tříd (PVC, ostatní) byly použity ty z kombinace hyperparametrů, která měla v rámci Bayesovské optimalizace nejvyšší F1 skóre klasifikace PVC ve validační množině, a jsou uvedeny v Tab. 4.4. Učení probíhalo po dobu 11 cyklů, kdy bylo dosaženo nejvyššího F1 skóre PVC ve validační množině a model byl uložen pro klasifikaci testovacích dat. Matici záměn a F1 skóre jednotlivých tříd (validační množina) z poslední epochy učení je možné vidět v Tab. 4.5 a Tab. 4.6. Klasifikace do 2 tříd byla provedena na testovací množině čítající 689 vzorků, jejichž rozdělení do tříd je obsahem Tab. 4.7.

Tab. 4.4: Hyperparametry použité pro klasifikaci do 2 tříd

Hyperparametr	Hodnota
Maximální učební krok	0,004
Dávka (trénování)	28
Počet epoch	4
Max_extra	0
Max_duplicate	3
Mixup (alfa)	0,07

Tab. 4.5: Matice záměn, klasifikace do 2 tříd, validační množina

		Skutečná třída	
		PVC	Ostatní
Predikce	PVC	31	30
	Ostatní	10	616

Tab. 4.6: F1 skóre, klasifikace do 2 tříd, validační množina

Skutečná třída	F1 skóre
PVC	0,61
Ostatní	0,97
Celkové	0,79

Tab. 4.7: Počet záznamů ve třídách, klasifikace do 2 tříd, testovací množina

Skutečná třída	PVC	Ostatní
Počet záznamů	42	647

Tab. 4.8 ukazuje matici záměn a Tab. 4.9 shrnuje jednotlivé metriky klasifikace testovací množiny. Celkové F1 skóre je 0,74, kdy vyššího skóre (0,95) dosahuje třída ostatní a nižšího skóre (0,53) dosahuje třída PVC. Celková senzitivita je 0,85 a přesnost 0,70. Vyšších hodnot těchto metrik vždy dosahuje třída ostatní oproti PVC.

Tab. 4.8: Matice záměn, klasifikace do 2 tříd, testovací množina

		Skutečná třída	
		PVC	Ostatní
Predikce	PVC	32	47
	Ostatní	10	600

Tab. 4.9: Metriky, klasifikace do 2 tříd, testovací množina

Skutečná třída	F1 skóre	Přesnost	Senzitivita
PVC	0,53	0,41	0,76
Ostatní	0,95	0,98	0,93
Celkové	0,74	0,70	0,85

Při porovnání F1 skóre validačních dat v Tab. 4.6 a testovacích dat v Tab. 4.9 je zřejmé, že model dosahuje vyšších F1 skóre při validaci. Konkrétně rozdíl mezi F1 skóre PVC je 0,08 (0,61 validace a 0,53 test) a mezi celkovými F1 skóre 0,05 (0,79 validace a 0,74 test).

#### 4.2.2 Klasifikace do 3 tříd

Jako hyperparametry pro klasifikaci do 3 tříd (PVC, sinusový rytmus, ostatní) byly použity ty z kombinace hyperparametrů, která měla v rámci Bayesovské optimalizace nejvyšší F1 skóre klasifikace PVC ve validační množině, a jsou uvedeny v Tab. 4.10. Učení probíhalo po dobu 12 cyklů. Matici záměn a F1 skóre jednotlivých tříd (validační množina) z poslední epochy učení je možné vidět v Tab. 4.11 a Tab. 4.12. Klasifikace do 3 tříd byla provedena na testovací množině čítající 690 vzorků, jejichž rozdělení do tříd je obsahem Tab. 4.13.

Tab. 4.10: Hyperparametry použité pro klasifikaci do 3 tříd

Hyperparametr	Hodnota
Maximální učební krok	0,0043
Dávka (trénování)	27
Počet epoch	4
Max_extra	0
Max_duplicate	3
Mixup (alfa)	0,02

Tab. 4.11: Matice záměn, klasifikace do 3 tříd, validační množina

		Skutečná třída		
		PVC	Sinusový r.	Ostatní
Predikce	PVC	29	0	26
	Sinusový r.	3	200	92
	Ostatní	9	23	305

Tab. 4.12: F1 skóre, klasifikace do 3 tříd, validační množina

Skutečná třída	F1 skóre
PVC	0,68
Sinusový r.	0,77
Ostatní	0,82
Celkové	0,75

Tab. 4.13: Počet záznamů ve třídách, klasifikace do 3 tříd, testovací množina

Skutečná třída	PVC	Sinusový r.	Ostatní
Počet záznamů	42	224	424

Tab. 4.14 ukazuje matici záměn a Tab. 4.15 shrnuje jednotlivé metriky klasifikace testovací množiny. Celkové F1 skóre je 0,72 a skóre jednotlivých tříd jsou vyrovnanější než v případě klasifikace do 2 tříd, u PVC se jedná o hodnotu 0,63. Celková senzitivita je 0,81 a přesnost 0,70. Klasifikace do tříd PVC a sinusový rytmus dosahují vyšších hodnot senzitivity a nižších hodnot přesnosti, oproti tomu klasifikace do třídy ostatní proběhla s vyšší přesností, ale nižší senzitivitou. Pokud byl záznam třídy PVC nebo sinusový rytmus klasifikován nesprávně, jednalo se zejména o klasifikaci do třídy ostatní. Při klasifikaci ostatních záznamů se častěji objevovala nesprávná klasifikace do třídy sinusový rytmus.

Tab. 4.14: Matice záměn, klasifikace do 3 tříd, testovací množina

		Skutečná třída		
		PVC	Sinusový r.	Ostatní
Predikce	PVC	35	0	35
	Sinusový r.	1	212	110
	Ostatní	6	12	279

Tab. 4.15: Metriky, klasifikace do 3 tříd, testovací množina

Skutečná třída	F1 skóre	Přesnost	Senzitivita
PVC	0,63	0,50	0,83
Sinusový r.	0,76	0,66	0,95
Ostatní	0,77	0,94	0,66
Celkové	0,72	0,70	0,81

Porovnáním F1 skóre validačních dat v Tab. 4.12 a testovacích dat v Tab. 4.15 je zjištěno, že model dosahuje mírně vyšších F1 skóre při validaci. F1 skóre PVC validační množiny je 0,68 a testovací množiny 0,63, rozdíl je tedy 0,05. Celkové F1 skóre se liší o 0,03 (0,75 validace a 0,72 test).

### 4.2.3 Klasifikace do 8 tříd

Jako hyperparametry pro klasifikaci do 8 tříd (PVC, PAC, sinusový rytmus (Sinus), fibrilace síní (FiS), tachykardie (Tachy), bradykardie (Brady), infarkt myokardu (IM) a ostatní) byly použity ty z kombinace hyperparametrů, která měla v rámci Bayesovské optimalizace nejvyšší F1 skóre klasifikace PVC ve validační množině, a jsou uvedeny v Tab. 4.16. Učení probíhalo po dobu 10 cyklů. Matici záměn a F1 skóre jednotlivých tříd (validační množina) z poslední epochy učení je možné vidět v Tab. 4.17 a Tab. 4.18. Klasifikace do 8 tříd byla provedena na testovací množině čítající 620 vzorků, jejichž rozdělení do tříd je obsahem Tab. 4.19.

Tab. 4.16: Hyperparametry použité pro klasifikaci do 8 tříd

Hyperparametr	Hodnota
Maximální učební krok	0,005
Dávka (trénování)	32
Počet epoch	12
Max_extra	1
Max_duplicate	2
Mixup (alfa)	1,0

Tab. 4.17: Matice záměn, klasifikace do 8 tříd, validační množina

		Skutečná třída							
		PVC	PAC	Sinus	FiS	Tachy	Brady	IM	Ostatní
Predikce	PVC	10	0	0	0	0	0	2	1
	PAC	0	4	1	1	1	0	1	1
	Sinus	3	7	201	0	0	3	11	51
	FiS	0	2	0	33	3	1	0	8
	Tachy	0	0	5	3	68	0	0	6
	Brady	1	3	2	0	2	54	2	2
	IM	1	1	4	1	2	0	14	8
	Ostatní	1	6	10	4	2	0	3	67

Tab. 4.18: F1 skóre, klasifikace do 8 tříd, validační množina

Skutečná třída	F1 skóre
PVC	0,69
PAC	0,25
Sinus	0,81
FiS	0,74
Tachy	0,85
Brady	0,87
IM	0,44
Ostatní	0,56
Celkové	0,63

Tab. 4.19: Počet záznamů ve třídách, klasifikace do 8 tříd, testovací množina

Skutečná třída	PVC	PAC	Sinus	FiS	Tachy	Brady	IM	Ostatní
Počet záznamů	17	23	224	43	78	58	33	144

Tab. 4.20 ukazuje matici záměn a Tab. 4.21 shrnuje jednotlivé metriky klasifikace testovací množiny. Celkové F1 skóre je 0,56 a podobnému výsledku se rovnala i přesnost (0,57) a senzitivita (0,61). Klasifikace PVC měla v případě klasifikace do 8 tříd přesnost 0,63, senzitivitu 0,59 a F1 skóre 0,61. Z hlediska senzitivity jde tedy o horší výsledek než u předchozích typů klasifikací a z hlediska přesnosti jde o lepší výsledek. Metriky jednotlivých diagnóz se ovšem silně liší, kdy nejhorších výsledků bylo dosaženo při klasifikaci PAC s přesností, senzitivitou a F1 skóre shodně 0,13. Nejvyšší senzitivity bylo dosaženo u detekce bradykardie (0,93) a nejvyššího F1 skóre u tachykardie (0,81).

Tab. 4.20: Matice záměn, klasifikace do 8 tříd, testovací množina

		Skutečná třída							
		PVC	PAC	Sinus	FiS	Tachy	Brady	IM	Ostatní
Predikce	PVC	10	1	0	0	1	0	2	2
	PAC	0	3	10	1	0	1	0	8
	Sinus	1	8	175	1	2	1	1	30
	FiS	2	2	2	31	10	0	3	18
	Tachy	1	1	5	2	60	0	0	2
	Brady	2	4	16	7	2	54	3	17
	IM	1	3	14	1	2	1	22	24
	Ostatní	0	1	2	0	1	1	2	43

Tab. 4.21: Metriky, klasifikace do 8 tříd, testovací množina

Skutečná třída	F1 skóre	Přesnost	Senzitivita
PVC	0,61	0,63	0,59
PAC	0,13	0,13	0,13
Sinus	0,79	0,80	0,78
FiS	0,56	0,46	0,72
Tachy	0,81	0,85	0,77
Brady	0,66	0,51	0,93
IM	0,44	0,32	0,67
Ostatní	0,44	0,86	0,30
Celkové	0,56	0,57	0,61

I u klasifikace do 8 tříd je zřejmý rozdíl mezi F1 skóre validačních dat (Tab. 4.18) a testovacích dat (Tab. 4.21). U F1 skóre PVC se jedná o rozdíl 0,08 (0,69 validace a 0,61 test) a celkové F1 skóre je u validační množiny 0,63 a u testovací množiny 0,56, jedná se tedy o rozdíl 0,07.

#### 4.2.4 Srovnání výsledků klasifikace do 2, 3 a 8 tříd

Hlavní metriky klasifikace do 2, 3 a 8 tříd srovnává Tab. 4.22 a to pro validační (val) i testovací (test) množinu dat. U všech typů klasifikace bylo F1 skóre celkové i PVC vždy mírně vyšší u validační množiny dat než u testovací množiny.

Z hlediska výsledků klasifikace testovací množiny bylo nejlepšího celkového F1 skóre dosaženo při klasifikaci do 2 tříd (0,74), F1 skóre PVC bylo ovšem nejnižší (0,53). Nejvyšší F1 skóre PVC (0,63) měla klasifikace do 3 tříd, kdy bylo také dosaženo nejvyšší senzitivity detekce PVC a to 0,83. Nejhorší celkové F1 skóre (0,56) a senzitivita PVC (0,59) byly výsledkem klasifikace do 8 tříd.

Tab. 4.22: Srovnání hlavních metrik klasifikace do 2,3 a 8 tříd

Počet tříd klasifikace	Celkové F1 skóre (val)	F1 skóre PVC (val)	Celkové F1 skóre (test)	F1 skóre PVC (test)	Senzitivita PVC (test)
2 třídy	0,79	0,61	0,74	0,53	0,76
3 třídy	0,75	0,68	0,72	0,63	0,83
8 tříd	0,63	0,69	0,56	0,61	0,59

Pro porovnání jsou v Tab. 4.23 uvedeny délky učení modelu pro všechny typy klasifikace, a to od načtení filtrovaných dat (mimo samotné načtení) bez následné klasifikace dat testovací množiny. V tabulce je rovněž uveden počet cyklů a epoch učení modelu. Délka

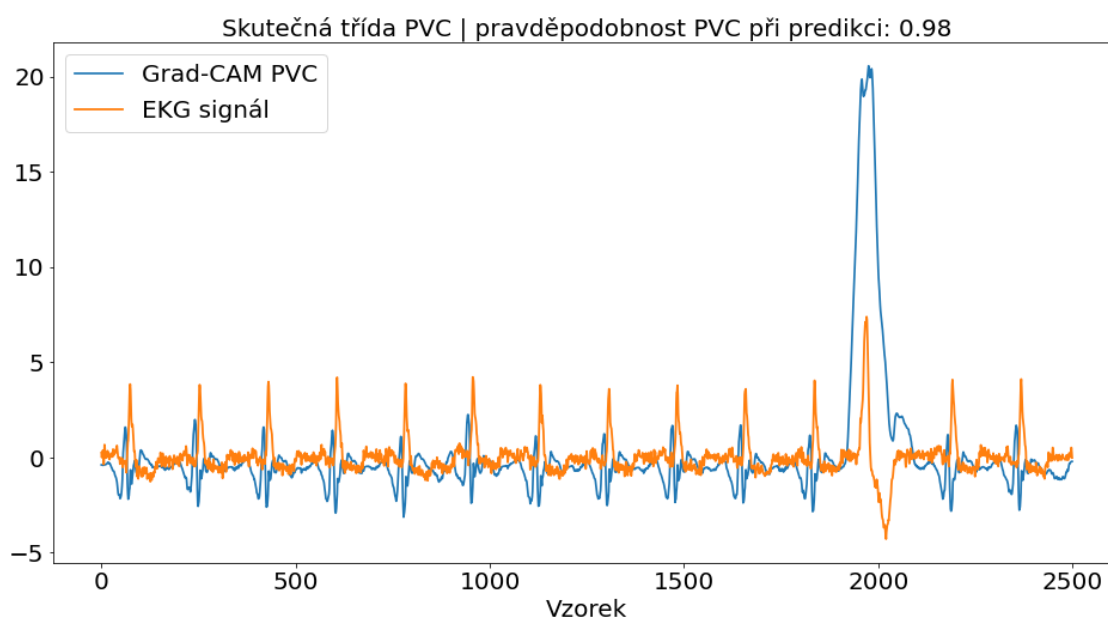
učení modelu pro klasifikaci do 2 a 3 tříd se liší pouze o dvě sekundy i přes vyšší počet cyklů trénování modelu pro klasifikaci do 3 tříd. Délka učení modelu pro klasifikaci do 8 tříd je výrazně vyšší (přibližně 6krát), a to zejména z důvodu vyššího počtu cyklů a epoch v jednotlivých cyklech.

Tab. 4.23: Délky učení

Počet tříd klasifikace	Délka učení	Počet cyklů/epoch v cyklu
2 třídy	1 min 57 s	11/4
3 třídy	1 min 59 s	12/4
8 tříd	11 min 52 s	10/12

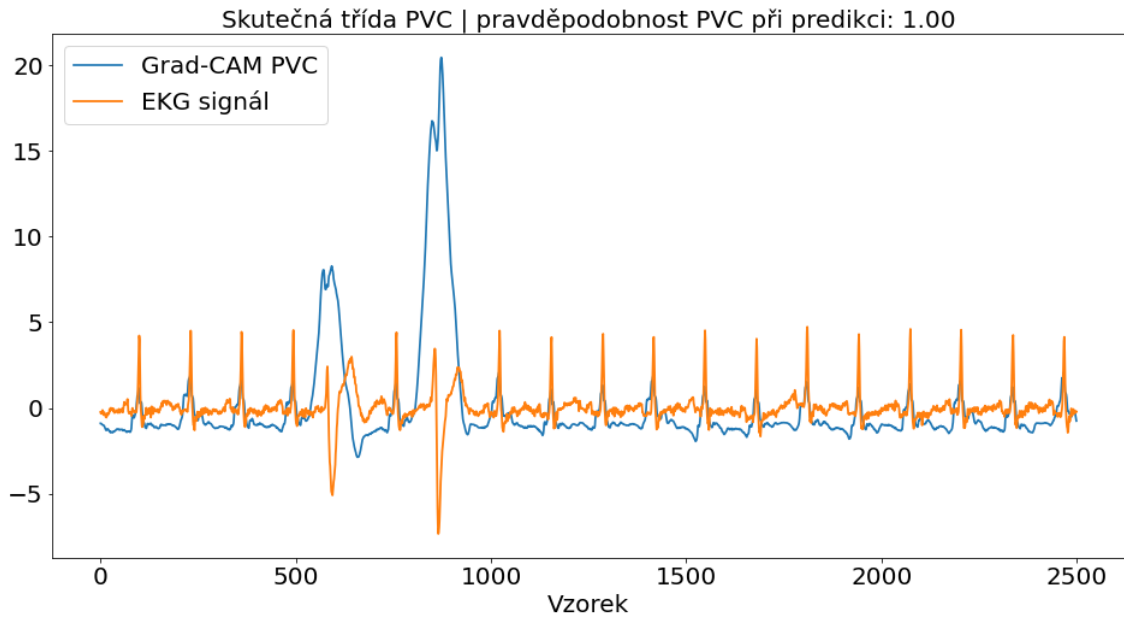
### 4.3 Grad-CAM a lokalizace PVC

Byla implementována metoda Grad-CAM, která je následně využita pro zjištění místa výskytu PVC v signálech EKG, tedy zjištění lokality klíčových instancí. Příklad výstupu metody Grad-CAM je na Obr. 4.1, Obr. 4.2 a Obr. 4.3, kde je vždy zobrazen jak původní signál EKG z validační množiny dat (pouze svod II pro zvýšení přehlednosti), tak výstup ve formě Grad-CAM mapy. V nadpisu jednotlivých obrázků jsou rovněž pravděpodobnosti příslušnosti ke třídě PVC. Na Obr. 4.1 je možné pozorovat výkyv hodnot mapy přímo v místě lokalizace PVC, stejně tak i na Obr. 4.2. Obr. 4.3 ukazuje signál, který nebyl správně klasifikován do třídy PVC a rovněž hodnoty mapy nejsou v žádné části signálu výrazně zvýšené.

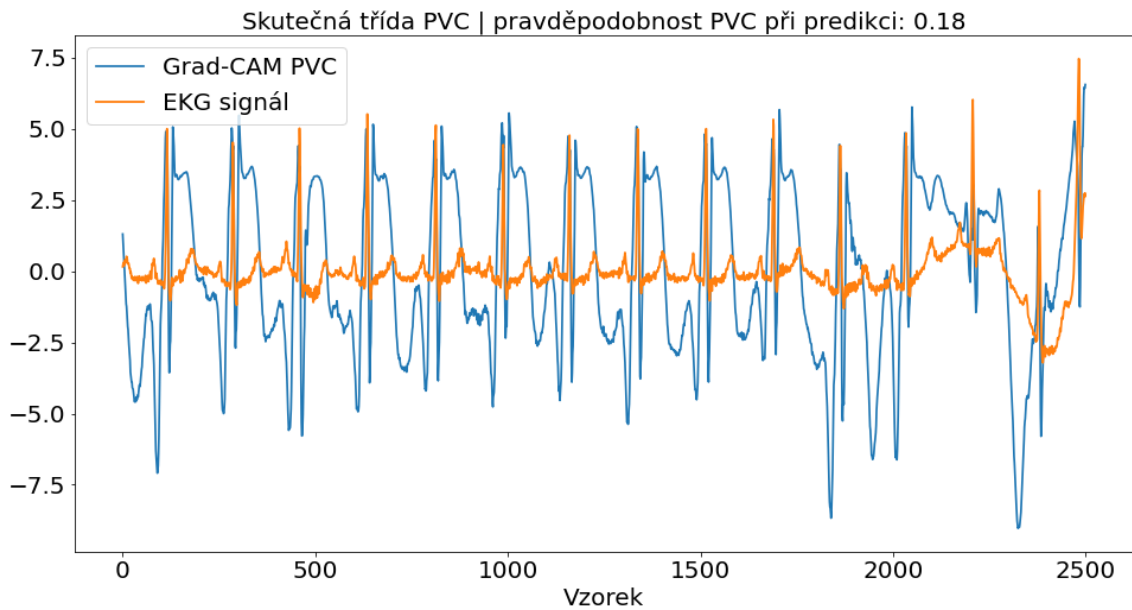


Obr. 4.1: Grad-CAM + signál EKG, správná klasifikace (0,98)



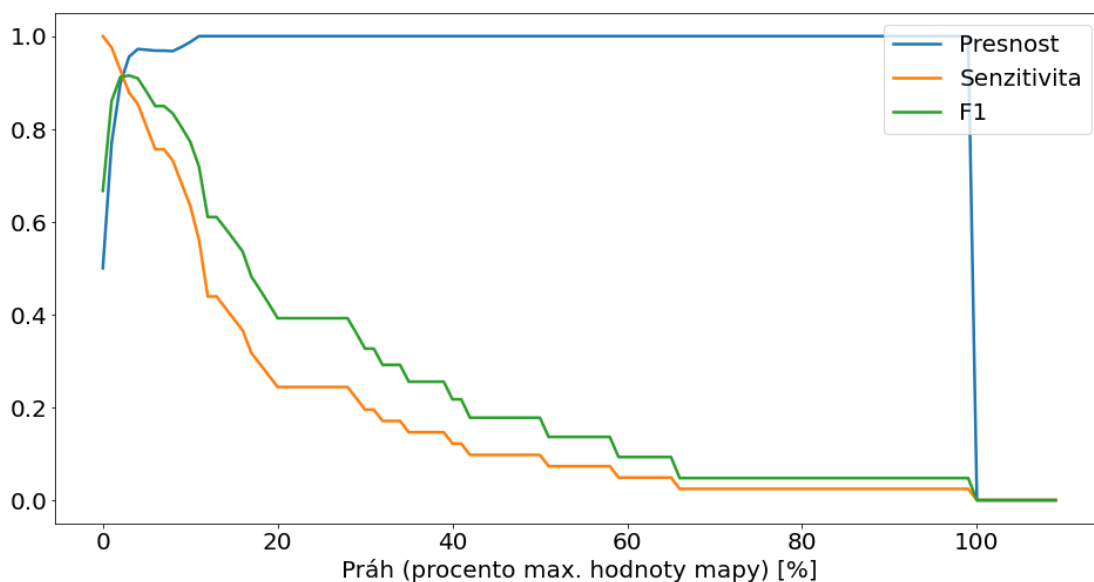


Obr. 4.2: Grad-CAM + signál EKG, správná klasifikace (1,00)



Obr. 4.3: Grad-CAM + signál EKG, nesprávná klasifikace (0,18)

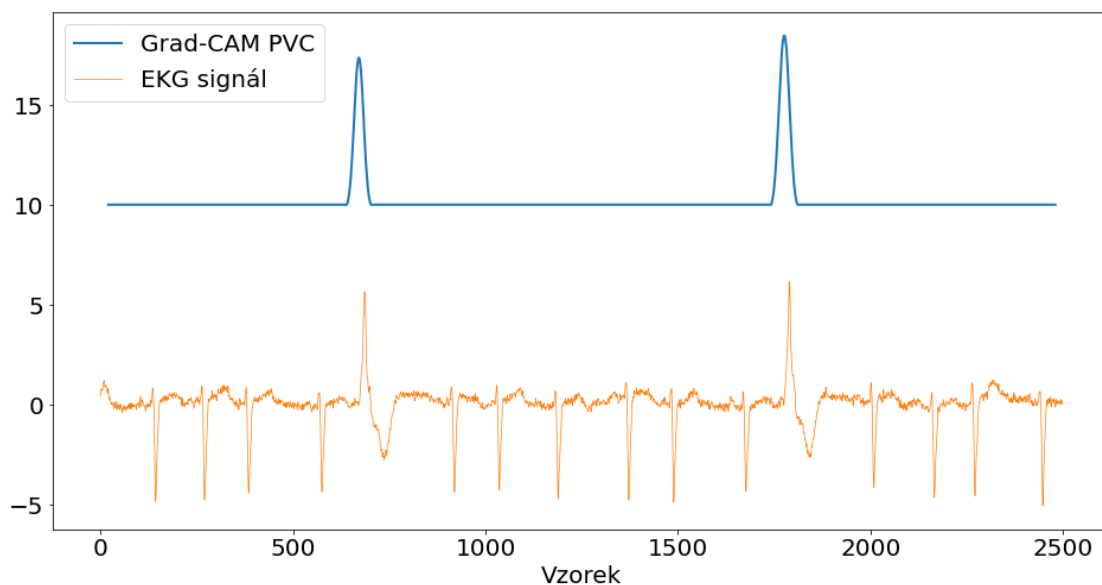
Následně byla provedena klasifikace signálů validační množiny (po naučení modelu pro klasifikaci do 3 tříd) pouze pomocí prahování Grad-CAM mapy třídy PVC (viz kapitola 3.7). Práh se rovnal hodnotě 0 až 100 % (po jednom procentu) z maximální hodnoty Grad-CAM map validační množiny. Pokud byla hodnota určité mapy vyšší než práh, odpovídající signál byl klasifikován jako PVC. Následně byla spočtena přesnost (precision), senzitivita a F1 skóre pro všechny záznamy v množině dat pro každou velikost prahu, čímž byl nalezen optimální práh dosahující nejvyššího F1 skóre klasifikace do třídy PVC. Na Obr. 4.4 je graf závislosti jednotlivých metrik na zvoleném prahu. Je tedy možné zhodnotit, že pouhým prahováním Grad-CAM map s vhodně nastaveným prahem je F1 skóre klasifikace PVC až 0,92 (validační množina). Model je tedy při učení a následně klasifikaci velmi často zaměřen na místa signálu, kde se vyskytuje daná arytmie.



Obr. 4.4: Graf závislosti jednotlivých metrik na zvoleném prahu

### 4.3.1 Vyhodnocení výsledků lokalizace

Poslední částí navrženého programu byla lokalizace PVC ve 42 záznamech testovací množiny dat skutečné třídy PVC po provedené klasifikaci do 3 tříd (jelikož při klasifikaci testovacích dat do 3 tříd bylo dosaženo nejvyššího F1 skóre PVC). Výstupem tedy byla prahovaná Grad-CAM mapa, kdy práh byl nastaven empiricky jako 98% kvantil Grad-CAM mapy daného záznamu, spolu s původním signálem. Příklad výsledku je na Obr. 4.5, kde jednotlivé kmity mapy (modrá linie) označují místa výskytu PVC.



Obr. 4.5: Záznam EKG (svod III) s lokalizací PVC

Evaluace výsledků (s využitím svodů I, III, aVF a V6 záznamů EKG) byla odborně provedena Ing. Monikou Míkovou, biomedicínskou inženýrkou na úseku Invazivní elektrofyziologie Interní kardiologické kliniky Fakultní nemocnice Brno, které tímto děkuji. Výsledky zhodnocení jsou v Tab. 4.24, která rozděluje záznamy do více kategorií včetně závislosti na predikované třídě. První z kategorií, ve které bylo obsaženo 62 % testovaných záznamů, zahrnuje záznamy, u kterých byla provedena správná lokalizace všech přítomných PVC. Další kategorie čítající 17 % ze všech záznamů popisuje signály, u kterých byla provedena správná lokalizace PVC, ale nebyly lokalizovány všechny PVC v záznamu (zejména se jednalo o vynechání jednoho PVC z mnoha). U poměrně velkého počtu záznamů (14 %) bylo shledáno, že signál nezobrazuje přítomnost PVC, ale v několika případech PAC, fibrilaci síní nebo sinusový rytmus se šumem. Kvalita některých záznamů byla velmi špatná, a proto nebylo možné 7 % záznamů hodnotit, což popisuje poslední kategorie evaluace.

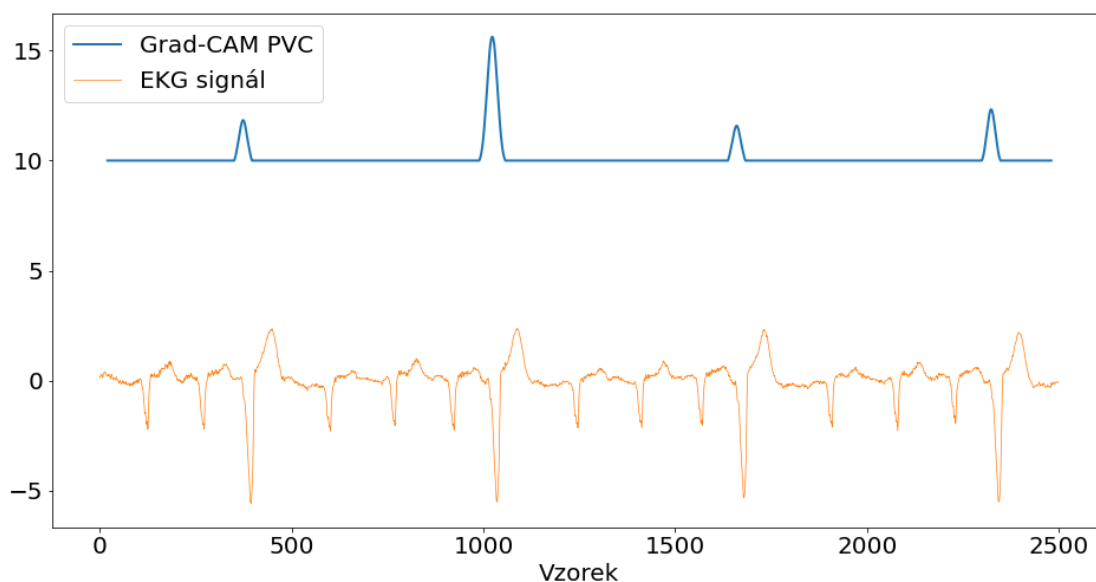
Z výsledků tedy vyplývá, že pokud záznam s anotací PVC skutečně obsahuje PVC, proběhne u 79 % takovýchto záznamů správná lokalizace všech PVC v signálu, u 21 % signálů se jedná o částečně správnou lokalizaci, tedy nejsou detekovány všechny PVC, které signál obsahuje. Také je možné konstatovat, že je určité množství případů (14 % testovací množiny), kdy je u záznamu přiřazena chybná anotace a u 7 % signálů je znemožněna evaluace výsledků kvůli velkému poměru šumu v signálu.

Při zaměření se na predikce tříd lze určit, že pokud proběhne alespoň částečně správná lokalizace PVC, naprostá většina takovýchto záznamů je správně zařazena do třídy PVC. Pokud signál neobsahuje PVC, tak u 3 případů ze 6 byl signál zařazen do třídy PVC a u 3 případů do třídy sinusový rytmus nebo ostatní. 2 ze 3 záznamů zobrazujících signál s velkým množstvím šumu byly zařazeny do třídy ostatní.

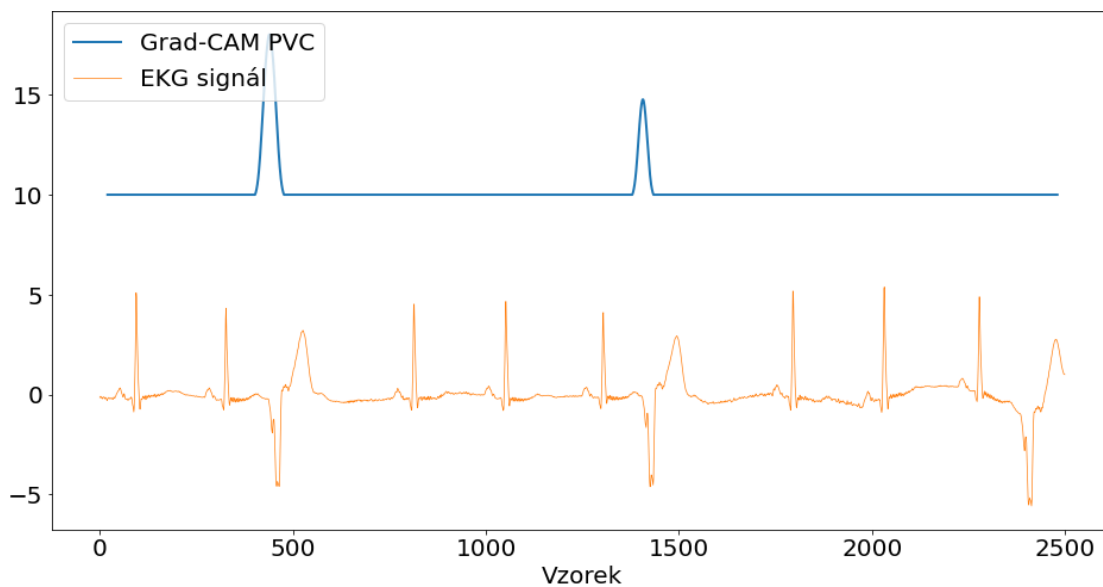
Tab. 4.24: Evaluace výsledků lokalizace PVC

Predikce při klasifikaci	Správná lokalizace 100 % PVC	Správná lokalizace méně než 100 % PVC	Záznam neobsahuje PVC	Záznam obsahuje velké množství šumu
PVC	25	6	3	1
Sinusový r.	0	0	1	0
Ostatní	1	1	2	2
Celkem	26	7	6	3
Procento z celku [%]	62	17	14	7

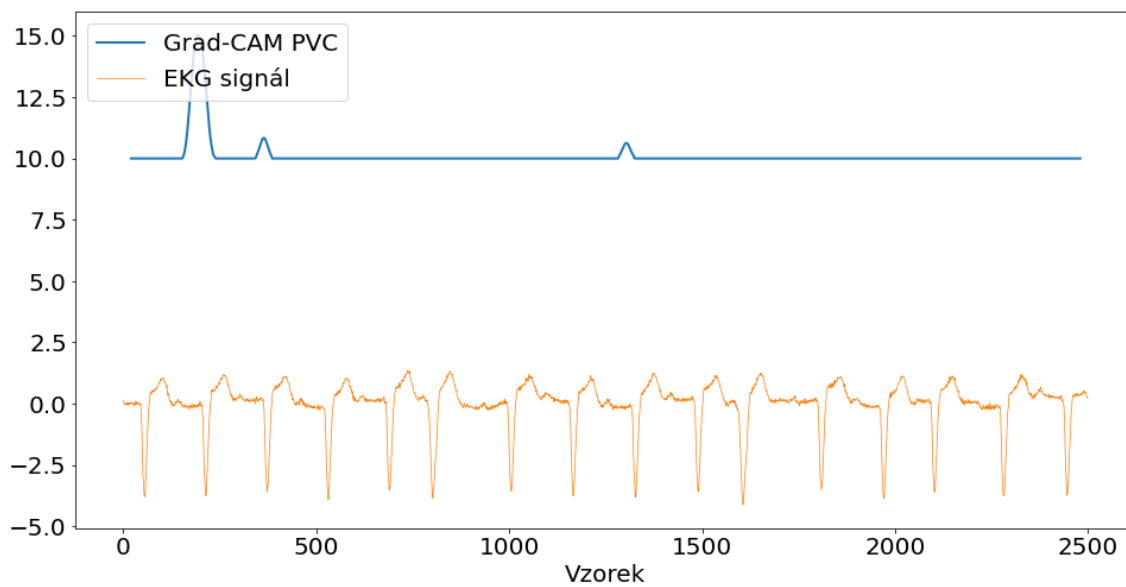
Na Obr. 4.6, Obr. 4.7, Obr. 4.8 a Obr. 4.9 jsou zobrazeny příklady záznamů jednotlivých kategorií. Na Obr. 4.6 je zobrazen příklad záznamu EKG se správnou lokalizací všech PVC. Obr. 4.7 zachycuje záznam, kde jsou lokalizovány dva PVC ze tří (chybí tedy lokalizace PVC na konci záznamu). Nesprávně anotovaný signál je obsahem Obr. 4.8, kde se nachází síňové extrasystoly (PAC), a na Obr. 4.9 je příklad poslední kategorie, tedy signálu s artefakty.



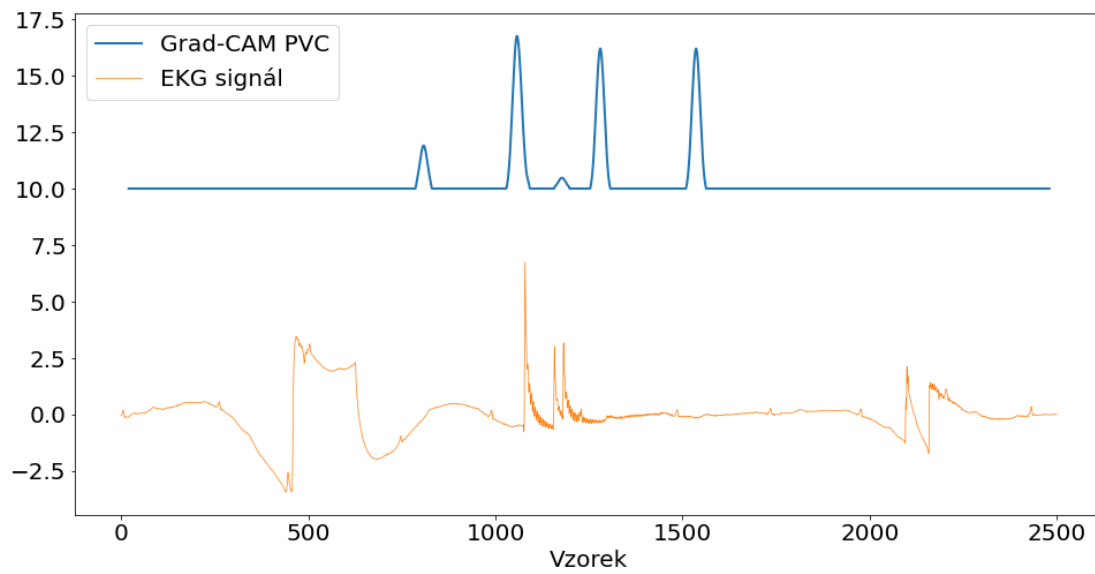
Obr. 4.6: Záznam EKG (svod III) se správnou lokalizací PVC



Obr. 4.7: Záznam EKG (svod III) s částečně správnou lokalizací PVC



Obr. 4.8: Záznam EKG (svod III) zobrazující PAC



Obr. 4.9: Záznam EKG (svod III) – šum, artefakty

## 5. DISKUZE

V předchozí kapitole byly uvedeny výsledky Bayesovské optimalizace hyperparametrů pro klasifikaci do 2, 3 a 8 tříd. Následně byly tyto hyperparametry využity ke trénování modelu, pomocí kterého byla klasifikována data z testovací množiny. Posledním krokem bylo využití metody Grad-CAM pro účely detekce místa výskytu arytmie v signálu. V této kapitole budou diskutovány výše zmíněné výsledky a budou představeny změny, které by mohly v budoucnu napomoci výsledkům lepším.

### 5.1 Bayesovská optimalizace hyperparametrů

Výsledky Bayesovské optimalizace byly uvedeny ve formě pěti nejlepších kombinací optimalizovaných hyperparametrů. Z Tab. 4.1, Tab. 4.2 a Tab. 4.3, které tyto informace obsahují, je možné vyčíst možné limitace metody a obecně optimalizace hyperparametrů – jedná se zejména o malé rozsahy prohledávaných hodnot. I když Bayesovská optimalizace pracuje efektivně (doba potřebná k nalezení optimálních hyperparametrů je nižší než např. u metody prohledávání stavového prostoru, viz kapitola 2.8), stále se jedná o časově a výpočetně náročný proces. Je tedy nutné poměrně výrazně omezit prohledávaný prostor a opravdu optimální kombinace hyperparametrů tak nemusí být nalezena.

Maximální učební krok dosáhl ve třech z pěti nejlepších kombinací hyperparametrů (klasifikace do 8 tříd) své hranice 0,0050, vhodné by tedy bylo do Bayesovské optimalizace zahrnout i vyšší maximální učební kroky (např. 0,01, viz kapitola 2.8.2), zvětšit tak rozsah prohledávaných hodnot a zjistit, zda klasifikace v tomto případě nedosahuje lepších výsledků (F1 skóre PVC).

Rozsah velikosti trénovací dávky by měl být zvýšen až na několik stovek (viz kapitola 2.8.4), jedná se ale o velký nárůst výpočetní náročnosti, proto musela být zvolena nižší maximální hodnota velikosti dávky. Bylo prakticky vyzkoušeno i prohledávání většího rozsahu hodnot, nestačila ovšem paměť počítače, Bayesovská optimalizace hyperparametrů selhala a bylo tak nutné maximální hodnotu dávky snížit.

Další možností by rovněž bylo zvýšení maximální hodnoty rozsahu počtu epoch, jelikož u optimalizace hyperparametrů pro klasifikaci do 2 a 8 tříd bylo vždy u jedné kombinace z pěti dosaženo maximální hodnoty z rozsahu počtů epoch.

Dle tabulek zmíněných na začátku této kapitoly je možné konstatovat, že rozsah hodnot `max_extra` a `max_duplicate` byl zvolen dostatečně velký, jelikož optimální hodnoty nedosahovaly maximálních hodnot uvedených rozsahů a nepředpokládá se, že by jejich navýšení přineslo zlepšení schopnosti modelu správně klasifikovat data.

Mixup alfa může nabývat hodnot pouze v rozsahu 0 až 1 a stejně tak byl nastaven prohledávaný prostor. Dle typu klasifikace (tedy do kolika tříd klasifikace probíhá) jsou zřetelné poměrně velké rozdíly ve velikosti optimálních hodnot alfa. V případě

klasifikace do 8 tříd bylo u pěti nejlepších kombinací hyperparametrů časté vysoké alfa, ve třech kombinacích šlo dokonce o alfa 1. Oproti tomu optimální alfa při klasifikaci do 3 tříd bylo velmi nízké, v řádech setin. Dle těchto výsledků je možné stanovit, že vyšší alfa metody mixup je výhodnější u klasifikace do vyššího počtu tříd. Důvodem může být to, že při klasifikaci do nižšího počtu tříd množina definovaná jako „ostatní“ signály obsahuje mnoho různých typů arytmií, jejich skutečná třída je ale vždy jedna a ta stejná. Naopak pokud existuje vyšší počet tříd, tak i vyšší počet arytmií má jako svou skutečnou třídu přiřazenou svou reálnou třídu a v metodě mixup je tak lépe definována směs dvou míchaných signálů s odpovídajícími skutečnými třídami.

Shrnutí předchozích odstavců ve formě doporučení, jak by mohly být upraveny rozsahy optimalizovaných hyperparametrů, je v Tab. 5.1. Je nutné podotknout, že po rozšíření prohledávaného prostoru se nemusí najít hodnoty hyperparametrů, které by zaručily vyšší F1 skóre klasifikace PVC než při použití nynějších optimálních hyperparametrů. Velkým omezením je ovšem nedostupnost dostatečně velké výpočetní kapacity, která zamezuje použití velkých rozsahů hyperparametrů při jejich Bayesovské optimalizaci.

Tab. 5.1: Doporučované úpravy rozsahů pro Bayesovskou optimalizaci hyperparametrů

Hyperparametr	Vhodná úprava
Maximální učební krok	Zvýšení maximální hodnoty v rozsahu (např. na 0,01)
Dávka (trénování)	Zvýšení maximální hodnoty v rozsahu (na několik stovek)
Počet epoch	Zvýšení maximální hodnoty v rozsahu (např. na 20)
Max_extra	Rozsah je dostatečný
Max_duplicate	Rozsah je dostatečný
Mixup (alfa)	Byl prohledán maximální možný rozsah

## 5.2 Klasifikace

Z výsledků klasifikace je možné určit, že nejvyššího F1 skóre a senzitivity klasifikace bylo dosaženo při klasifikaci do 3 tříd, tedy tříd PVC, sinusový rytmus a ostatní. Z matice záměn je také zřejmé, že model umí velmi dobře rozlišovat mezi signály PVC a sinusový rytmus, signály ostatní jsou ovšem často označeny jako PVC, což výrazně snižuje přesnost (precision) a tedy i F1 skóre klasifikace PVC. Méně často jsou rovněž signály skutečné třídy PVC označeny predikcí ostatní. Stejně chování sítě je možné pozorovat i u klasifikace do 2 tříd, tedy do tříd PVC a ostatní, kdy je velké množství záznamů



ostatní zařazeno do třídy PVC. Údaje ve formě konkrétních čísel se nachází v Tab. 4.8 a Tab. 4.14.

Je tedy zřejmé, že se model nedokázal plně naučit, jak PVC vypadá a které signály tedy do této třídy zařadit. Důvodem může být menší počet signálů PVC v datech, kdy tato arytmie má několik různých poměrně odlišných morfologií a model tedy nemusí mít dostatečné množství vzorků pro naučení každé morfologie dané arytmií.

Rovněž jak bylo popsáno v kapitole 4.3.1, množství záznamů v datasetu mělo nesprávně přiřazené anotace, což se může odrážet jak na kvalitě učení, tak na kvalitě klasifikace, jelikož záznam skutečné třídy PVC, který ovšem obsahuje jinou arytmií, bude považován za chybně klasifikovaný, pokud bude zařazen do jiné třídy než PVC, ale zároveň toto zařazení může být reálně správné. Doporučením pro další práci by tedy bylo použití jiné databáze EKG záznamů, popřípadě kontrola již anotovaných dat aktuálně používané databáze.

Záměrem učení modelu pro klasifikaci do 8 tříd bylo zjištění, zda má klasifikace do více tříd vliv na úspěšnost klasifikace arytmií, zejména PVC. Délka trvání učení modelu pro klasifikaci do 8 tříd byla přibližně 6krát delší než při učení modelu pro klasifikaci do 2 a 3 tříd. Výstupem bylo, že klasifikace do 8 tříd dosahuje podobného F1 skóre PVC jako klasifikace do 3 tříd (0,61, respektive 0,63), přičemž došlo ke zvýšení přesnosti (precision) a znatelnému snížení senzitivity z 0,83 na 0,59. Dle matice záměn (Tab. 4.20) největší nepřesnosti tvořila klasifikace do třídy ostatní, kdy signály skutečné třídy ostatní byly velmi často klasifikovány jako určitý druh arytmií. Návrhem na zlepšení by tedy bylo učení modelu pro klasifikaci do více tříd tak, aby i záznamy diagnóz, které vykazují podobné morfologie v EKG záznamech jako PVC, měly vlastní třídu (blokady Tawarova raménka apod.), čímž by mohlo dojít ke snížení četnosti záměn mezi zařazením do třídy PVC a ostatní. Problémem tohoto přístupu ovšem může být nižší počet záznamů v dalších třídách původně zařazených do třídy ostatní, a tedy méně dat k učení a následné klasifikaci testovací množiny. Také se zvýší počet tříd ke klasifikaci, ale zároveň se nezvýší kapacita modelu, což může vést k poklesu kvality klasifikace.

Pokud jsou srovnány výsledky, které se získaly klasifikací PVC ve validační množině dat při učení s výsledky získanými klasifikací testovací množiny dat (viz kapitola 4.2.4), je možné zhodnotit, že celkové F1 skóre klasifikace je vždy mírně vyšší u validační množiny než u testovací, kdy tento rozdíl činí maximálně 0,07. Stejný výsledek je i u F1 skóre PVC, pouze rozdíl je maximálně 0,08. Důvodem tohoto rozdílu může být přeučení sítě a tím nižší schopnost generalizace. Mohlo by tedy být výhodné změnit metodiku výběru bodu při učení modelu, kdy je uložen daný model pro klasifikaci testovací množiny. Nyní je tento bod vybrán jako cyklus, při kterém dosáhne klasifikace PVC ve validační množině nejvyššího F1 skóre, což může být příčinou přeučení modelu.

### 5.2.1 Srovnání s literaturou

Při porovnání výsledků práce s dostupnou literaturou (shrnuje v Tab. 5.2) bylo jako metrika použito F1 skóre klasifikace PVC. Nebylo využito celkové F1 skóre klasifikace, jelikož může být odlišné u klasifikace do různého počtu tříd (je rozdíl, zda probíhá klasifikace do 2 nebo 20 tříd) a také existují různé druhy výpočtů celkového F1 skóre, které by si nemusely odpovídat.

Pro srovnání výsledků klasifikace této práce s dostupnými výsledky jiných autorů se nabízí výstupy soutěže PhysioNet Challenge z roku 2020, jejíž zadáním bylo klasifikovat 12svodové záznamy EKG do 24 tříd. Jedním ze soutěžních příspěvků byl i příspěvek od Feng a kol. [57] s F1 skóre klasifikace PVC 0,21, což je výrazně méně než jaká jsou výsledná F1 skóre dosažená v této práci.

Dalším příkladem využití CNN, konkrétně 1D CNN s reziduálními bloky je článek od Zhang a kol. [58] provádějící klasifikaci 12svodových záznamů EKG do 9 tříd, kdy se F1 skóre PVC rovnalo 0,85 a bylo tedy výrazně vyšší než skóre uvedená v této práci (0,53-0,63 dle typu klasifikace).

Článek od Yao a kol. [59] představil klasifikaci do 9 tříd a to pomocí několika různých architektur, např. VGG-6 dosahující F1 skóre klasifikace PVC 0,69 a ATI-CNN dosahující F1 skóre PVC 0,86, tedy jde o F1 skóre PVC vyšší, než jaké jsou výsledky této práce.

Dle literatury je možné konstatovat, že metriky klasifikace PVC pomocí neuronových sítí se velmi liší a mimo zvolené architektury závisí na několika dalších faktorech, např. použití stejné/jiné databáze záznamů pro trénovací i testovací množinu dat nebo na kvalitě dostupných dat. Omezujícím prvkem klasifikace této arytmie je také nižší počet záznamů PVC v mnoha dostupných databázích, tedy model není dostatečně natrénovat a zároveň je tak běžný nízký počet záznamů s PVC v testovací množině dat, proto existují velké výkyvy ve výsledcích klasifikace. V porovnání s literaturou dosahuje představovaný model spíše horších výsledků zejména při klasifikaci do 2 tříd, při klasifikaci do 3 a 8 tříd se hodnota této metriky zvyšuje. Oproti článku soutěže PhysioNet Challenge ovšem dosahuje model výrazně lepších F1 skóre u všech typů klasifikace s rozdílem 0,3-0,4 mezi jednotlivými F1 skóre.

Tab. 5.2: Srovnání s literaturou

Práce	Architektura	F1 skóre PVC
Feng a kol. [57]	MIC-ResNet	0,21
Zhang a kol. [58]	1D CNN	0,85
Yao a kol. [59]	VGG-6	0,69
Yao a kol. [59]	ATI-CNN	0,86
Tato práce	InceptionTime	0,53/0,61/0,63

### 5.3 Lokalizace PVC

Metoda Grad-CAM byla využita pro vyhledání lokalit výskytu PVC v záznamech, kdy došlo ke zjištění, že pokud záznam anotovaný jako PVC opravdu obsahuje PVC, prahovaný Grad-CAM je schopen lokalizovat všechny či alespoň většinu PVC v záznamech. Největším problémem ovšem byly chybné anotace u některých signálů, popřípadě vysoký podíl šumu v záznamech, kdy nebylo možné diagnostikovat, jaká arytmie se v záznamu nachází nebo byla diagnostika velmi nejistá, i přestože došlo k odbornému zhodnocení výsledků.

Bylo by rovněž vhodné provést další testy, které by zjistily, zda neexistuje velikost prahu pro prahování Grad-CAM, která by zaručila lepší výsledky při lokalizaci PVC – tedy např. u záznamů, u kterých nebylo odhaleno 100 % všech PVC, by došlo k detekci 100 % PVC. Velikost prahu v této práci byla zvolena empiricky.

Zajímavou úpravou by rovněž mohlo být vytvoření prahovaných Grad-CAM map třídy, do které byl chybně zařazen signál jiné skutečné třídy, čímž by se mohly získat informace o klíčové instanci, tedy o tom, na co se síť při klasifikaci zaměřuje. Tímto způsobem by mohla být provedena investigace důvodů, proč došlo k chybné klasifikaci a tyto informace by mohly sloužit k dalšímu zlepšení průběhu předzpracování dat nebo ke změně průběhu učení.

## 6. ZÁVĚR

Cílem této práce bylo představit program pro klasifikaci PVC v záznamech EKG umožňující také lokalizovat jednotlivé PVC. V teoretické části práce bylo pojednáno o základní elektrofyziologii srdce, včetně popisu běžných arytmí, nebo EKG. Následující kapitoly se již zabývaly neuronovými sítěmi se zaměřením na konvoluční neuronové sítě a jejich učení. Tato část byla zakončena informacemi o přístupu Multiple-Instance Learning, kdy pro vytvoření map ukazujících na klíčové instance byla využita popsána metoda Grad-CAM.

Byl navržen program prostřednictvím programovacího jazyka Python, který vhodně předpracovává dostupnou databázi EKG signálů spolu s jejich anotacemi. Pro klasifikaci záznamů do 2 (PVC, ostatní), 3 (PVC, ostatní) a 8 tříd (PVC, PAC, sinusový rytmus, fibrilace síní, tachykardie, bradykardie, infarkt myokardu a ostatní) byl zvolen model architektury InceptionTime a byla provedena Bayesovská optimalizace hyperparametrů. Optimální hodnoty hyperparametrů byly použity při učení modelu, který následně klasifikoval data testovací množiny.

Výsledná celková F1 skóre jsou u klasifikace do 2 tříd 0,74, do 3 tříd 0,72 a do 8 tříd 0,56. F1 skóre klasifikace PVC jsou při klasifikaci do 2 tříd 0,53, do 3 tříd 0,63 a do 8 tříd 0,61. Největším problémem byla nízká přesnost (precision) klasifikace PVC, jelikož často docházelo ke klasifikaci signálu skutečné třídy ostatní do třídy PVC.

Poslední částí diplomové práce byla lokalizace PVC v záznamech EKG, k čemuž byla využita metoda Grad-CAM. Výstupem tohoto úkolu byla prahovaná Grad-CAM mapa pro určitou třídu spolu s daným signálem EKG testovací množiny ukazující na lokality výskytu PVC. Výsledky byly odborně ohodnoceny a bylo zjištěno, že některé záznamy jsou chybně anotovány, popřípadě mají silné zastoupení šumu, čímž se stávají obtížně diagnostikovatelné. Z výsledků ovšem také vyplynulo, že pokud se na záznamu EKG opravdu nalézá PVC, program úspěšně nalezne lokalitu většiny PVC v jednotlivých záznamech.

Hlavní cíle práce, tedy implementace modelu pro klasifikaci záznamů EKG a lokalizaci PVC v jednotlivých signálech, byly splněny. Možným pokračováním práce by byly změny v předzpracování dat (filtrace dalších typů šumů, změna rozdělení signálů do jednotlivých množin apod.), klasifikace dat do více tříd, hlubší prozkoumání vlivu techniky mixup na správnost a chování klasifikace nebo úprava stávajícího prahu pro mapy vystupující z Grad-CAM.

## LITERATURA

- [1] LINDBERG, T. a kol. Prevalence and Incidence of Atrial Fibrillation and Other Arrhythmias in the General Older Population: Findings From the Swedish National Study on Aging and Care. *Gerontology & geriatric medicine*, 2019, sv. 5. ISSN 2333-7214.
- [2] STEFAN, S. a AGAMEMNON, D. *Atlas fyziologie člověka*. 8. vyd. Grada Publishing, a.s., 2016. 448 s. ISBN 9788024742717.
- [3] Heart Conduction System. [online]. [cit. 2. 12. 2020]. Dostupné z: <<https://medmediasolutions.com/heart-conduction-system/>>.
- [4] WHITE, S. The Conducting System of the Heart. [online]. [cit. 2. 12. 2020]. Dostupné z: <<https://teachmeanatomy.info/thorax/organs/heart/conducting-system/>>.
- [5] THALER, M. S. *EKG a jeho klinické využití*. 6. vyd. Grada, 2013. 320 s. ISBN 9788024741932.
- [6] HAMPTON, J. R. *EKG stručně, jasně, přehledně*. 7. vyd. Grada, 2013. 192 s. ISBN 9788024742465.
- [7] O'ROURKE, R. A. a kol. *Kardiologie - Hurstův manuál pro praxi, překlad dvanáctého vydání*. 12. vyd. Grada, 2010. 800 s. ISBN 9788024731759.
- [8] ŠTEFÁNEK, J. Komorová extrasystola - EKG. [online]. [cit. 21. 3. 2021]. Dostupné z: <<https://www.stefajir.cz/komorova-extrasystola-ekg>>.
- [9] VOJÁČEK, J. a KETTNER, J. *Klinická kardiologie*. 1. vyd. Nucleus HK, 2009. 925 s. ISBN 9788087009581.
- [10] PNGWING. [online]. [cit. 2. 4. 2021]. Dostupné z: <<https://www.pngwing.com/en/free-png-pswcu>>.
- [11] CADOGAN, M. ECG Lead positioning. [online]. [cit. 2. 4. 2021]. Dostupné z: <<https://litfl.com/ecg-lead-positioning/>>.
- [12] KHER, R. Signal Processing Techniques for Removing Noise from ECG Signals. *Journal of Biomedical Engineering and Research*, 2019, sv. 3, č. 1. ISSN 2688-5093.
- [13] RASCHKA, S. a MIRJALILI, V. *Python Machine Learning: Machine Learning and Deep Learning with Python, Scikit-Learn, and TensorFlow 2*. 3. vyd. Packt Publishing, 2019. 770 s. ISBN 9781789955750.
- [14] BURKOV, A. *The Hundred-Page Machine Learning Book*. Andriy Burkov, 2019. 160 s. ISBN 9781999579517.
- [15] THEOBALD, O. *Machine Learning for Absolute Beginners: A Plain English Introduction*. 2018. ISBN 9781549617218.
- [16] WINDER, P. *Reinforcement Learning*. O'Reilly Media, 2020. ISBN 9781492072348.
- [17] AGGARWAL, C. C. *Neural Networks and Deep Learning: A Textbook*. 1. vyd. Springer International Publishing, 2018. 520 s. ISBN 9783319944630.
- [18] HOLMES, J. H. Knowledge Discovery in Biomedical Data: Theory and Methods. *V Methods in Biomedical Informatics*. Academic Press, 2014, s. 179-240.
- [19] HOLČÍK, J. a KOMENDA, M. *Matematická biologie: e-learningová učebnice*. Masarykova univerzita, 2015. ISBN 9788021080959.
- [20] BHARDWAJ, A. a kol. *Deep Learning Essentials: Your hands-on guide to the fundamentals of deep learning and neural network modeling*. Packt Publishing, 2018. 284 s. ISBN 9781785887772.

- [21] ZHANG, A. a kol. *Dive into Deep Learning*. 2020. 1007 s.
- [22] PATTERSON, J. a GIBSON, A. *Deep Learning: A Practitioner's Approach*. 1. vyd. O'Reilly Media, 2017. 532 s. ISBN 9781491914236.
- [23] ELGENDY, M. *Deep Learning for Vision Systems*. Manning Publications, 2020. 480 s. ISBN 9781617296192.
- [24] THUYET, D. Q. a kol. A robot system equipped with deep convolutional neural network for autonomous grading and sorting of root-trimmed garlics. *Computers and Electronics in Agriculture*, 2020, sv. 178. ISSN 0168-1699.
- [25] YAMASHITA, R. a kol. Convolutional neural networks: an overview and application in radiology. *Insights into Imaging*, 2018, sv. 9, č. 4, s. 611-629. ISSN 1869-4101.
- [26] ISSA, D. a kol. Speech emotion recognition with deep convolutional neural networks. *Biomedical Signal Processing and Control*, 2020, sv. 59. ISSN 1746-8094.
- [27] BERNICO, M. *Deep Learning Quick Reference: Useful hacks for training and optimizing deep neural networks with TensorFlow and Keras*. 1. vyd. Packt Publishing, 2018. 284 s. ISBN 9781788838917.
- [28] BONACCORSO, G. *Mastering Machine Learning Algorithms: Expert techniques to implement popular machine learning algorithms and fine-tune your models*. Packt Publishing, 2018. ISBN 9781788625906.
- [29] KINGMA, D. P. a BA, J. Adam: A Method for Stochastic Optimization. *ArXiv*, 2017.
- [30] NOGUEIRA, F. Bayesian Optimization: Open source constrained global optimization tool for Python. GitHub, 2020.
- [31] SNOEK, J. a kol. Practical Bayesian Optimization of Machine Learning Algorithms. *ArXiv*, 2012.
- [32] SZEGEDY, C. a kol. Going deeper with convolutions. V *Proceedings of the The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.
- [33] ILSE, M. a kol. Deep multiple instance learning for digital histopathology. V ZHOU, S.K., a kol. *Handbook of Medical Image Computing and Computer Assisted Intervention*. Academic Press, 2020, s. 521-546.
- [34] CARBONNEAU, M.-A. a kol. Multiple instance learning: A survey of problem characteristics and applications. *Pattern Recognition*, 2018, sv. 77, s. 329-353. ISSN 0031-3203.
- [35] QUELLEC, G. a kol. Multiple-Instance Learning for Medical Image and Video Analysis. *IEEE Reviews in Biomedical Engineering*, 2017, sv. 10, s. 213-234. ISSN 1937-3333.
- [36] ILSE, M. a kol. Attention-based Deep Multiple Instance Learning. V *Proceedings of the 35th International Conference on Machine Learning*, Stockholm, Švédsko, 2018 PMLR, 2127-2136.
- [37] BEDMUTHA, M. S. a RAMAN, S. *Computer Vision and Image Processing*. V SINGH, S.K.: Springer, 2021.
- [38] CHORARIA, A. Class Activation Mapping in Deep Learning. [online]. [cit. 7. 5. 2021]. Dostupné z: <<https://www.loginradius.com/blog/async/class-activation-mapping/>>.
- [39] LEARNING, M. S. I. T. D. MIT 6.S191: Introduction to Deep Learning. [online]. [cit. 17. 4. 2021]. Dostupné z: <<https://blog.tensorflow.org/2018/04/mit-6s191-introduction-to-deep-learning.html>>.

- [40] Conference Proceedings SELVARAJU, R. R. a kol. *Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization*. 2017 IEEE International Conference on Computer Vision (ICCV). 2380-7504.
- [41] BROWNLEE, J. *Imbalanced Classification with Python: Better Metrics, Balance Skewed Classes, Cost-Sensitive Learning*. Machine Learning Mastery, 2020. 463 s.
- [42] OGUIZA, I. tsai - A state-of-the-art deep learning library for time series and sequential data. [Verze 0.2.17]. GitHub, 2020.
- [43] REBACK, J. a kol. pandas-dev/pandas: Pandas. [Verze 1.1.5]. Zenodo, 2020.
- [44] HARRIS, C. R. a kol. Array programming with NumPy. *Nature*, 2020, sv. 585, č. 7825, s. 357-362. ISSN 1476-4687.
- [45] XIE, L. a kol. Computational Diagnostic Techniques for Electrocardiogram Signal Analysis. *Sensors*, 2020, sv. 20, č. 21. ISSN 1424-8220.
- [46] PODDER, P. a kol. Design and Implementation of Butterworth, Chebyshev-I and Elliptic Filter for Speech Signal Analysis. *International Journal of Computer Applications*, 2014, sv. 98, č. 7, s. 12-18.
- [47] *Feature Engineering and Computational Intelligence in ECG Monitoring*. 1. vyd. Springer Singapore, 2020. 268 s. ISBN 9789811538247.
- [48] Conference Proceedings ZHANG, H. a kol. *mixup: Beyond Empirical Risk Minimization*. International Conference on Learning Representations.
- [49] SMITH, L. N. A disciplined approach to neural network hyper-parameters: Part 1 - learning rate, batch size, momentum, and weight decay. *ArXiv*, 2018.
- [50] GUGGER, S. The 1cycle policy. [online]. [cit. 30. 4. 2021]. Dostupné z: <<https://sgugger.github.io/the-1cycle-policy.html>>.
- [51] WANG, Z. a kol. Time Series Classification from Scratch with Deep Neural Networks: A Strong Baseline. *ArXiv*, 2016.
- [52] FAZLE, K. a kol. LSTM Fully Convolutional Networks for Time Series Classification. *IEEE Access*, 2018, sv. 6, s. 1662–1669. ISSN 2169-3536.
- [53] HE, T. a kol. Bag of Tricks for Image Classification with Convolutional Neural Networks. *ArXiv*, 2018.
- [54] ZOU, X. a kol. Integration of residual network and convolutional neural network along with various activation functions and global pooling for time series classification. *Neurocomputing*, 2019, sv. 367, s. 39-45. ISSN 0925-2312.
- [55] FAWAZ, H. I. a kol. InceptionTime: Finding AlexNet for time series classification. *Data Mining and Knowledge Discovery*, 2020, sv. 34, č. 6, s. 1936-1962. ISSN 1573-756X.
- [56] RAHIMIAN, E. a kol. XceptionTime: A Novel Deep Architecture based on Depthwise Separable Convolutions for Hand Gesture Classification. *ArXiv*, 2019.
- [57] Conference Proceedings FENG, Y. a VIGMOND, E. *Deep Multi-Label Multi-Instance Classification on 12-Lead ECG*. 2020 Computing in Cardiology. ISSN 2325-887X.
- [58] ZHANG, D. a kol. Interpretable deep learning for automatic diagnosis of 12-lead electrocardiogram. *iScience*, 2021, sv. 24, č. 4. ISSN 2589-0042.
- [59] YAO, Q. a kol. Multi-class Arrhythmia detection from 12-lead varied-length ECG using Attention-based Time-Incremental Convolutional Neural Network. *Information Fusion*, 2020, sv. 53. ISSN 1566-2535.