



# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

## FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

## ÚSTAV AUTOMATIZACE A MĚŘICÍ TECHNIKY

DEPARTMENT OF CONTROL AND INSTRUMENTATION

## IMPLEMENTACE A PRAKTICKÉ OVĚŘENÍ METOD PRO PREDIKTIVNÍ IDENTIFIKACI PORUCH VALIVÝCH LOŽISEK

IMPLEMENTATION AND PRACTICAL VERIFICATION OF METHODS FOR PREDICTIVE IDENTIFICATION OF ROLLING BEARINGS FAILURES

### DIPLOMOVÁ PRÁCE

MASTER'S THESIS

### AUTOR PRÁCE

AUTHOR

Bc. Martin Bár

### VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Stanislav Klusáček, Ph.D.

BRNO 2022

# Diplomová práce

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

Ústav automatizace a měřicí techniky

**Student:** Bc. Martin Bár

**ID:** 200459

**Ročník:** 2

**Akademický rok:** 2021/22

**NÁZEV TÉMATU:**

## **Implementace a praktické ověření metod pro prediktivní identifikaci poruch valivých ložisek**

### **POKYNY PRO VYPRACOVÁNÍ:**

Cílem diplomové práce je implementace a ověření vybraných metod pro identifikaci a predikci poruch valivých ložisek využitelných v diagnostice točivých strojů na základě dat získaných z vibračních snímačů.

Zadání diplomové práce:

1. Nastudujte a zpracujte rešerši týkající se valivých ložisek, diagnostiky jejich poruch, metod pro identifikaci a predikci těchto poruch na základě signálů získaných z vibračních snímačů.
2. Na základě této rešerše implementujte vybrané diagnostické a prediktivní metody ve vybraných prostředích (např. LabVIEW, MATLAB, BK Connect).
3. Pomocí rotor modelu pro valivá ložiska změřte reálnou sadu dat pro ložiska bez poruchy a s různými typy poruch (např. vnějšího/vnitřního kroužku, klece, kuliček) a na těchto měřených datech ověřte vámi implementované metody.
4. Porovnejte implementované metody s ohledem na vybrané parametry - například rychlost výpočtu, požadavky na výpočetní hardware, kvalitu měřených dat (SNR, vzorkovací kmitočet, šířka pásma, aj.), spolehlivost predikce poruchy, atp.
5. Zhodnoťte implementované metody na základě parametrů bodu č. 4.

### **DOPORUČENÁ LITERATURA:**

[1] RANDALL, Robert B. a Jérôme ANTONI. Rolling element bearing diagnostics - A tutorial. Mechanical Systems and Signal Processing. 2011. 25(2). 485-520. ISSN 08883270.

**Termín zadání:** 7.2.2022

**Termín odevzdání:** 18.5.2022

**Vedoucí práce:** Ing. Stanislav Klusáček, Ph.D.

**Konzultant:** Ing. Martin Doseděl

**doc. Ing. Petr Fiedler, Ph.D.**

předseda rady studijního programu

### **UPOZORNĚNÍ:**

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

## ABSTRAKT

Cílem diplomové práce je identifikovat a klasifikovat poruchy valivých ložisek.

První část práce se zabývá diagnostikou ložisek s využitím vlastních naměřených dat. Byla poškozena dvě ložiska. První bylo poškozeno deformací a korozí. U druhého ložiska bylo uměle vytvořeno poškození vnější dráhy. Vibrace z běhu ložisek byly periodicky sbírány a poté zpracovány v MATLABu. Hodnoty statistických indikátorů naznačovaly poškození u obou ložisek. Obálková analýza odhalila u obou ložisek vývoj závady na vnějším kroužku a také postupný vývoj závady na kleci.

V druhé části práce byly použity metody strojového učení ke klasifikaci závady na datech z databáze CWRU. Data z akcelerometru byla rozdělena dvěma způsoby do bloků. Pro 2D konvoluční síť byly z bloků vytvořeny vibrační obrázky. Nejlepší přesnosti predikce dosahovala 1D konvoluční neuronová síť (1DCNN) (99,2 %), z klasických metod strojového učení poté neuronová síť (94,6 %) a SVM (94,4 %). Na zmenšování trénovacích dat je nejodolnější metoda Random Forest a SVM, z konvolučních sítí poté MATLAB architektura a 1DCNN. Proti přidanému šumu je nejodolnější metoda Random Forest a neuronová síť, z konvolučních sítí poté 1DCNN. Pro silně zašuměná data jsou lepší klasické metody využívající statistické příznaky. Konvoluční sítě nedosahují dobré přesnosti, pravděpodobně kvůli transformaci surových dat z akcelerometru do vibračních obrázků.

## KLÍČOVÁ SLOVA

Valivá ložiska, Metody pro diagnostiku ložisek, Obálková analýza, Statistické indikátory, Strojové učení, Konvoluční neuronové sítě, Klasifikace, MATLAB

## ABSTRACT

The aim of this thesis is to identify and classify rolling bearing failures.

The first part of the thesis deals with the diagnosis of bearings using in-house measured data. Faults were introduced into two bearings. The first one was deformed and corroded. In the second bearing, the outer raceway was damaged. Vibration data was collected at regular intervals and processed in MATLAB. The values of the statistical features indicated faults in both bearings. Envelope analysis showed that both bearings had developed a fault on the outer raceway and a gradual fault on the cage.

In the second part of this thesis, machine learning methods were used to classify defective bearings using the CWRU data set. The accelerometer data were divided into blocks in two ways. Vibration images were created from these blocks for convolutional neural networks (CNNs). The best prediction accuracy was achieved by 1D convolutional neural network (1DCNN) (99.2 %), followed by neural network (94.6 %) and SVM (94.4 %). Random Forest and SVM are the best methods when the training set is reduced, and among CNNs, MATLAB architecture and 1DCNN are the best. The most noise resistant method is Random Forest and neural network, and among CNNs, 1DCNN is the best. Methods using statistical features perform better than CNNs on extremely noisy data. Convolutional networks do not achieve good accuracy, which is probably due to the conversion of raw accelerometer data into vibration images.

## KEYWORDS

Rolling Element Bearings, Bearing Diagnostic Techniques, Envelope Analysis, Statistical Indicators, Machine Learning, Convolutional Neural Network, Classification, MATLAB

BÁR, Martin. *Implementace a praktické ověření metod pro prediktivní identifikaci poruch valivých ložisek*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav automatizace a měřicí techniky, 2022, 100 s. Diplomová práce. Vedoucí práce: Ing. Stanislav Klusáček, Ph.D

## Prohlášení autora o původnosti díla

<b>Jméno a příjmení autora:</b>	Bc. Martin Bár
<b>VUT ID autora:</b>	200459
<b>Typ práce:</b>	Diplomová práce
<b>Akademický rok:</b>	2021/2022
<b>Téma závěrečné práce:</b>	Implementace a praktické ověření metod pro prediktivní identifikaci poruch valivých ložisek

Prohlašuji, že svou závěrečnou práci jsem vypracoval samostatně pod vedením vedoucí/ho 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 neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

Brno 18. 05. 2022

.....

podpis autora\*

---

\*Autor podepisuje pouze v tištěné verzi.

## PODĚKOVÁNÍ

Rád bych poděkoval vedoucímu diplomové práce panu Ing. Stanislavu Klusáčkovi, Ph.D. za odborné vedení, konzultace, trpělivost a podnětné návrhy k práci.

Dále bych chtěl vyjádřit poděkování konzultantovi diplomové práce panu Ing. Martinu Dosedělovi za vstřícnost, ochotu a pomoc při řešení problémů v laboratoři.

Brno 18. 05. 2022

.....

podpis autora

# Obsah

Úvod	13
<b>1 Valivá ložiska</b>	<b>14</b>
1.1 Poškození valivých ložisek	14
1.2 Trvanlivost valivých ložisek	18
1.3 Vibrace ložisek	20
1.3.1 Vliv závad ložiska na spektrum signálu	21
<b>2 Přehled metod pro diagnostiku ložisek</b>	<b>23</b>
2.1 Analýza signálu v časové oblasti	23
2.1.1 Analýza časového průběhu vibrací	23
2.1.2 Hodnocení podle hodnoty RMS	24
2.1.3 Hodnocení s použitím špičatosti (Kurtosis)	25
2.1.4 Hodnocení s použitím crest faktoru	26
2.2 Analýza signálu ve frekvenční oblasti	27
2.2.1 Spektrální analýza	27
2.2.2 Obálková analýza	28
2.2.3 Implementace obálkové analýzy	29
2.3 Analýza signálu v časově-frekvenční oblasti	31
2.3.1 Short Time Fourier Transform (STFT)	31
2.3.2 Vlnková transformace	32
2.3.3 Implementace vlnkové dekompozice	34
<b>3 Algoritmy strojového učení</b>	<b>35</b>
3.1 Standardní algoritmy strojového učení	35
3.1.1 k-NN (k-Nearest Neighbors)	35
3.1.2 SVM (Support Vector Machines)	36
3.1.3 RF (Random Forest)	39
3.1.4 NN (Neuronová síť)	41
3.2 Konvoluční neuronové sítě	43
3.2.1 Konvoluční vrstva	43
3.2.2 Vrstva normalizace dávky (batch normalization layer)	45
3.2.3 Vrstva vypuštění neuronů (dropout layer)	45
<b>4 Ověření metod na vlastních naměřených datech</b>	<b>46</b>
4.1 Popis měření ložisek	46
4.1.1 Přípravek na měření vibrací	46
4.1.2 Sběr dat	47

4.1.3	Poškození ložisek . . . . .	47
4.2	Zpracování naměřených dat . . . . .	48
4.2.1	MATLAB aplikace . . . . .	48
4.3	Metody na vlastních datech . . . . .	50
4.3.1	První poškozené ložisko . . . . .	50
4.3.2	Druhé poškozené ložisko . . . . .	55
<b>5</b>	<b>Klasifikace závady ložiska metodami strojového učení</b>	<b>59</b>
5.1	Vibrační data ložisek . . . . .	59
5.2	Klasifikace pomocí klasických metod STU . . . . .	61
5.2.1	Příprava dat a extrakce příznaků . . . . .	61
5.3	Nastavení parametrů klasických metod STU . . . . .	64
5.3.1	Hledání optimálních parametrů modelu . . . . .	64
5.4	Klasifikace pomocí CNN . . . . .	67
5.4.1	Příprava dat pro CNN . . . . .	67
5.5	Architektury CNN použité v této práci . . . . .	70
5.5.1	MATLAB CNN . . . . .	70
5.5.2	Vylepšená LeNet-5 . . . . .	70
5.5.3	ResNet-18 . . . . .	71
5.5.4	1DCNN . . . . .	72
5.6	Nastavení optimálních parametrů CNN . . . . .	73
5.7	Nalezení optimální hodnoty kroku učení . . . . .	74
5.7.1	MATLAB CNN . . . . .	74
5.7.2	LeNet-5 CNN . . . . .	75
5.7.3	ResNet-18 CNN . . . . .	75
5.7.4	1DCNN . . . . .	76
5.8	Vliv velikosti trénovací množiny na přesnost predikce . . . . .	77
5.8.1	Výsledky pro standardní metody STU . . . . .	77
5.8.2	Výsledky pro CNN . . . . .	79
5.9	Vliv šumu na přesnost predikce . . . . .	81
5.10	Srovnání klasických metod a CNN . . . . .	84
	<b>Závěr</b>	<b>88</b>
	<b>Literatura</b>	<b>90</b>
	<b>Seznam symbolů a zkratk</b>	<b>95</b>
	<b>A Závislost přesnosti predikce na velikosti trénovací množiny</b>	<b>97</b>
	<b>B Obsah přiloženého DVD</b>	<b>100</b>



# Seznam obrázků

1.1	Ukázka dvou typů poškození [3] . . . . .	14
1.2	Poškozený vnitřní kroužek (podpovrchová únava) [3] . . . . .	15
1.3	Ukázka abrasivního a adhezního poškození [3] . . . . .	15
1.4	Fretting a false brinneling [3] . . . . .	16
1.5	Kroužek a valivý element poškozený erozí el. proudem [3] . . . . .	16
1.6	Promáčklina a spalling ložiska [3] . . . . .	17
1.7	Ukázka prasklin na ložisku [3] . . . . .	17
1.8	Faktory ovlivňující trvanlivost ložisek [5], přeloženo a upraveno . . . .	19
1.9	Projev závad jednotlivých komponent [1] . . . . .	21
2.1	Projev závady na ložisku v časové oblasti . . . . .	23
2.2	Časový vývoj hodnoty špičatosti [16], přeloženo . . . . .	25
2.3	Časový vývoj hodnoty crest faktoru [18], přeloženo . . . . .	26
2.4	Porovnání spektra před a po poškození [19], přeloženo a upraveno . .	27
2.5	Obecný postup při obálkové analýze [21], přeloženo . . . . .	28
2.6	Obálková metoda pro ložisko s vadou na vnějším kroužku a zdravé ložisko . . . . .	29
2.7	Srovnání vlastností jednotlivých postupů [22], přeloženo . . . . .	31
2.8	Ukázka algoritmu FWT [25, 26], upraveno a přeloženo . . . . .	32
2.9	Původní signál a jeho rozklad na aproximace a detaily . . . . .	33
2.10	Zobrazení výstupu RMS vlnkové dekompozice . . . . .	34
3.1	Demonstrace k-NN pro $k = 3$ . . . . .	36
3.2	Grafická reprezentace SVM [33], přeloženo a upraveno . . . . .	37
3.3	Přidání druhého příznaku umožní separaci [32], upraveno . . . . .	38
3.4	Ukázka části jednoho rozhodovacího stromu ze souboru . . . . .	39
3.5	Neuronová síť s jednou skrytou vrstvou . . . . .	41
3.6	2D konvoluce [32], upraveno . . . . .	44
4.1	Přípravek fy Schenck a klec z PA66 . . . . .	46
4.2	Software BK Connect a modul LAN-XI 3160 . . . . .	47
4.3	Poškození vnější dráhy druhého ložiska . . . . .	48
4.4	Prostředí aplikace na inspekci změřených dat . . . . .	49
4.5	Amplitudové spektrum v závislosti na otáčkách přípravku . . . . .	50
4.6	Statistické metody pro ložisko 1 . . . . .	51
4.7	Obálkové spektrum pro první ložisko . . . . .	51
4.8	Obálkové spektrum pro první ložisko . . . . .	52
4.9	Vývoj frekvencí závad pro první ložisko . . . . .	52
4.10	Vývoj efektivní hodnoty obálky pro první ložisko . . . . .	53
4.11	Výsledek podílu RMS jednotlivých aproximací pro první ložisko . . .	54

4.12	Statistické metody pro ložisko 2 . . . . .	55
4.13	Obálkové spektrum pro druhé ložisko . . . . .	56
4.14	Obálkové spektrum pro druhé ložisko . . . . .	56
4.15	Vývoj frekvencí závad pro druhé ložisko . . . . .	57
4.16	Vývoj efektivní hodnoty obálky pro druhé ložisko . . . . .	57
4.17	Výsledek podílu RMS jednotlivých aproximací pro druhé ložisko . . . . .	58
5.1	Zařízení na testování ložisek z CWRU [41] . . . . .	59
5.2	Rozdělení dat s a bez opakování vzorků . . . . .	61
5.3	Ukázka křížové validace . . . . .	64
5.4	Pět nejlepších parametrů získaných pomocí 10Fold C-V . . . . .	65
5.5	Nejlepší parametry ostatních metod získané pomocí 10Fold C-V . . . . .	66
5.6	Pět nejlepších parametrů získaných pomocí 10Fold C-V, náhodné bloky dat . . . . .	66
5.7	Srovnání dvou způsobů generování příznaků ze sekce 5.2.1 . . . . .	66
5.8	Ukázka převodu 1D signálu na 2D obrázek . . . . .	68
5.9	Ukázka vibračního obrázku pro různé velikosti a místa závady . . . . .	68
5.10	Struktura a parametry zvolené architektury MATLAB CNN . . . . .	70
5.11	Struktura a parametry zvolené architektury LeNet-5 . . . . .	71
5.12	Residuální jednotky ([32], upraveno); ResNet-18, [46] . . . . .	72
5.13	Struktura a parametry zvolené architektury 1DCNN . . . . .	72
5.14	Prostředí MATLAB aplikace Experiment Manager . . . . .	73
5.15	Nalezení optimálního kroku pro učení MATLAB CNN . . . . .	74
5.16	Nalezení optimálního kroku pro učení LeNet-5 CNN . . . . .	75
5.17	Nalezení optimálního kroku pro učení ResNet-18 CNN . . . . .	75
5.18	Nalezení optimálního kroku pro učení 1DCNN . . . . .	76
5.19	Závislost přesnosti predikce na velikosti trénovací množiny pro standardní ML algoritmy . . . . .	78
5.20	Závislost přesnosti predikce na velikosti trénovací množiny pro standardní ML algoritmy, náhodně zvolené bloky dat . . . . .	78
5.21	Závislost přesnosti predikce na velikosti trénovací množiny pro CNN . . . . .	79
5.22	Závislost přesnosti predikce na velikosti trénovací množiny pro CNN, náhodně zvolené bloky dat . . . . .	80
5.23	Signál z poškození inner14 při různých SNR . . . . .	81
5.24	Odolnost standardních algoritmů STU na přidaný šum . . . . .	82
5.25	Odolnost standardních algoritmů STU na přidaný šum, náhodně zvolené bloky dat . . . . .	82
5.26	Odolnost CNN architektur na přidaný šum . . . . .	83
5.27	Odolnost CNN architektur na přidaný šum, náhodně zvolené bloky dat . . . . .	83

5.28	Porovnání přesnosti predikce jednotlivých algoritmů pomocí 10Fold C-V . . . . .	84
5.29	Srovnání přesnosti predikce pomocí 10Fold C-V . . . . .	84
5.30	Srovnání přesnosti predikce pomocí 10Fold C-V, náhodně vybrané bloky . . . . .	85
5.31	Časové porovnání pro standardní algoritmy strojového učení . . . . .	85
5.32	Celková doba trénování a doba trénování na epochu pro CNN . . . . .	85
5.33	Srovnání matice záměn pro 1DCNN a k-NN algoritmus . . . . .	86
5.34	Doba predikce 1000 pozorování . . . . .	87
A.1	Vliv velikosti trénovací množiny na přesnost predikce, NN1 . . . . .	97
A.2	Vliv velikosti trénovací množiny na přesnost predikce, NN2 . . . . .	97
A.3	Vliv velikosti trénovací množiny na přesnost predikce, k-NN . . . . .	97
A.4	Vliv velikosti trénovací množiny na přesnost predikce, SVM . . . . .	98
A.5	Vliv velikosti trénovací množiny na přesnost predikce, RF . . . . .	98
A.6	Vliv velikosti trénovací množiny na přesnost predikce, MATLAB CNN . . . . .	98
A.7	Vliv velikosti trénovací množiny na přesnost predikce, LeNet-5 CNN . . . . .	99
A.8	Vliv velikosti trénovací množiny na přesnost predikce, ResNet-18 CNN . . . . .	99
A.9	Vliv velikosti trénovací množiny na přesnost predikce, 1DCNN . . . . .	99

# Seznam tabulek

1.1	Vliv závad na frekvenční spektrum signálu [12] . . . . .	22
2.1	Klasifikace pásem mohutnosti vibrací podle ČSN ISO 10816-3 [14] . . . . .	24
5.1	Přehled dat z CWRU a rozdělení do tříd . . . . .	60
5.2	Shrnutí příznaků . . . . .	62
5.3	Seznam testovaných parametrů . . . . .	65

# Úvod

Tato práce se zabývá identifikací poškození valivých ložisek. V první kapitole jsou popsány základní typy závad na ložiscích a mechanismus jejich vzniku. Poté je zde uveden vliv jednotlivých závad na charakter měřených vibrací.

V další kapitole následuje teoretický popis metod, kterými lze provést diagnostiku valivých ložisek – analýza vibračního signálu v časové, frekvenční a časově-frekvenční oblasti. Tato kapitola se také věnuje popisu algoritmů strojového učení, které jsou poté použity ke klasifikaci místa a velikosti závady ložisek. Jedná se o výběr ze standardních metod strojového učení, jako je  $k$  nejbližších sousedů ( $k$ -NN), metoda podpůrných vektorů (SVM) a neuronové sítě. Poté jde o konvoluční neuronové sítě, které mají dvourozměrný vstup.

V první polovině praktické části práce je popsáno vlastní měření vibrací na dvou ložiskách. První ložisko bylo poškozeno deformací a korozí, druhé ložisko má vloženou vadu do vnější dráhy. V MATLABu byla vytvořena aplikace pro inspekci a analýzu naměřených dat. Tato aplikace umožňuje rychlé načtení dat, inspekci dat pro korektní nastavení parametrů metod a poté vykreslení vývoje vybraných parametrů v čase. Jedná se především o demonstraci funkčnosti metod uvedených v teoretické části na vlastních datech změřených v laboratoři.

Ve druhé polovině praktické části práce je popsána aplikace metod strojového učení na data z Case Western Reserve University (CWRU). Nejdříve je zde uvedena metodika rozdělení signálu z akcelerometru na bloky, z kterých jsou poté extrahovány příznaky. Práce pomyslně rozlišuje dvě skupiny metod strojového učení. První jsou standardní metody, které pracují s ručně vybranými příznaky z vibračního signálu, jako jsou  $k$ -NN a Random Forest. Kvalita predikce u těchto metod značně závisí na tom, jestli zvolené příznaky umožní rozlišit jednotlivé třídy poškození. Druhou skupinou jsou konvoluční neuronové sítě, u kterých dochází k extrakci příznaků v jejich první části – v konvolučních vrstvách. U obou těchto skupin metod je nejdříve nalezeno optimální nastavení jejich parametrů, poté jsou algoritmy dále testovány. Je zkoumán vliv velikosti trénovací množiny na přesnost predikce, vliv přidaného šumu na přesnost predikce a poté také výpočetní náročnost těchto metod.

V závěru jsou shrnuty výsledky diagnostických metod na vlastních datech změřených v laboratoři. Také jsou shrnuty výsledky jednotlivých algoritmů strojového učení. Jsou doporučeny vhodné algoritmy pro různé podmínky měření (silně zašuměná data, malá trénovací množina) a také jsou uvedeny případné nedostatky jednotlivých metod.

# 1 Valivá ložiska

Ložiska jsou důležitou komponentou strojů s rotujícími součástmi. Slouží k přenosu sil z rotujících částí zařízení na konstrukci a snižují tření. Z důvodu nevhodného zatížení při provozu, nadměrných vibrací zařízení způsobených nevývahou či vřely, nesprávného uskladnění, znečištění maziva či použití nevhodného maziva, nevhodnou dopravou a skladováním dochází k předčasnému poškození ložiska. Kvůli poškození musí být ložisko vyměněno před uplynutím konstrukční životnosti.[1]

Valivá ložiska se skládají z jednotlivých komponent. Tradičně jde o vnitřní kroužek, valivé elementy, klec na valivé elementy, vnější kroužek a těsnění.

Poškození způsobené výše uvedenými nevhodnými provozními podmínkami nebo i čistě opotřebením se projeví jako vady na jednotlivých komponentech ložiska. [1]

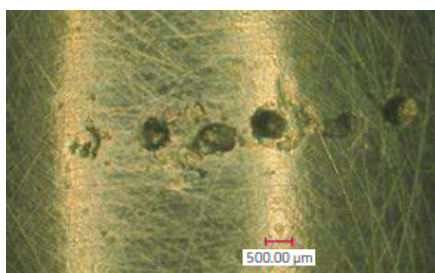
## 1.1 Poškození valivých ložisek

Podle normy ISO 15243:2017, která pojednává o poškození valivých ložisek, se dají poruchy ložisek rozdělit do následujících kategorií.

### Únava v místě valivého styku

Síly, které působí na valivé elementy a jejich dráhy jsou v místě valivého styku velmi velké. Postupem času, i v případě dodržení adekvátní lubrikace ložiska se toto projeví vznikem praskliny pod povrchem. Vzniklá prasklina postupuje objemem materiálu k povrchu dráhy, kde způsobí pitting (lokální poškození – malé dírky na povrchu) nebo spalling (úloмки materiálu způsobené únavou). [2]

Pitting a spalling je zobrazen na obrázku 1.1, samotný spalling je poté přiblížen na obrázku 1.2.



(a) Pitting



(b) Spalling

**Obr. 1.1:** Ukázka dvou typů únavových poškození [3]



**Obr. 1.2:** Poškozený vnitřní kroužek (podpovrchová únava) [3]

## Opotřebení

**Abrasivní opotřebení** – K abrasivnímu opotřebení dojde z důvodu kontaminace maziva částicemi. K poškození dojde v případě, že jsou částice větší, než je tloušťka filmu maziva. Tyto částice mohou následně poškodit styčné plochy. Opotřebení ploch se rozvíjí, protože je z nich postupně odírán materiál. [4]

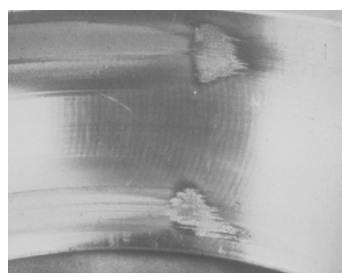
Jedná se o degenerativní proces, jelikož odírané částice zmenšují efektivitu maziva. Tento způsob opotřebení může rychle poškodit valivé elementy a jejich klec. [3]

**Adhezivní opotřebení** – Dochází k němu při kontaktu dvou ploch, které po sobě kloužou. Je charakterizováno přenosem materiálu z jednoho povrchu na druhý. Plochy se stávají postupně hrubější, čímž je poškozen film maziva. Poškození filmu maziva vede ke zvýšenému kontaktu kovu na kov. Působením tření se kontaktní plochy ohřívají, může dojít k popraskání. [3]

Tyto dva typy opotřebení jsou zobrazeny na obrázku 1.3.



(a) Abrasivní poškození -  
vnitřní kroužek



(b) Adhezní poškození -  
vnější kroužek

**Obr. 1.3:** Ukázka abrasivního a adhezního poškození [3]

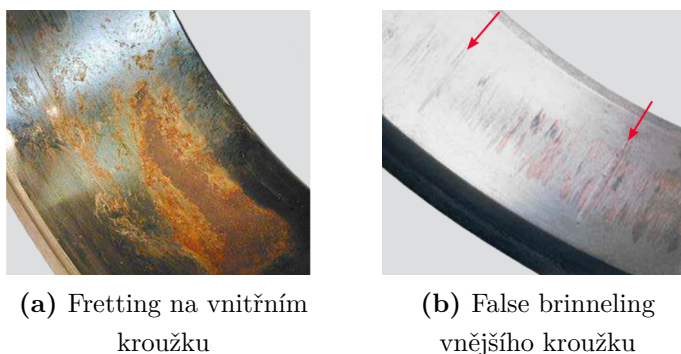
## Koroze

Koroze vzniká působením vlhkosti nebo korozivní kapaliny, která vnikne do ložiska.

**Fretting** – Dalším mechanismem vzniku koroze je tzv. fretting, tento termín popisuje pohyb mezi kroužkem a hřídelí nebo uložením. Následkem tohoto pohybu může dojít k odpadávání malých částeczek ložiska, které na vzduchu rychle zoxidují. [3]

**False brinelling** – Vzniká v důsledku vibrací mezi valivými elementy a jejich drahami, dochází tak k vzniku prohlubní v kroužku. [3]

Tyto dva typy opotřebení jsou zobrazeny na obrázku 1.4.



**Obr. 1.4:** Fretting a false brinelling [3]

## Elektrická eroze

V případě, že mezi vnitřním a vnějším kroužkem dojde k průchodu elektrického proudu přes valivý element o dostatečné proudové hustotě, dojde k roztavení materiálu kroužku/valivého elementu a vzniku kráteru. Pokud je proudová hustota nižší, může dojít k vytvoření mělkých a menších kráterů, které jsou oproti kráterům vytvořeným velkou proudovou hustotou relativně blízko sebe. [3]



**Obr. 1.5:** Kroužek a valivý element poškozený erozí el. proudem [3]



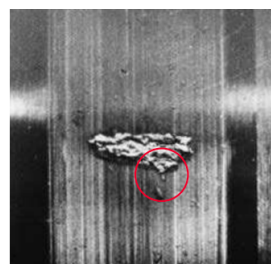
## Plastická deformace

K plastické deformaci dochází při přetížení ložiska, projeví se jako promáčknutí drah nebo valivých elementů. V případě, že se do dráhy ložiska dostane kontaminace, může být tato částice přejeta valivým elementem, což způsobí vznik prohlubně v dráze valivého elementu. Prohlubeň má po obvodu vyvýšený okraj, který je startovacím bodem pro vznik únavového poškození. Postupem času dojde ke vzniku spallingu, který začíná a rozšiřuje se od zadní části (z pohledu směru pohybu valivého elementu) prohlubně. [3]

Příklady plastické deformace ložiska jsou na obrázku 1.6.



(a) Promáčklina v dráze ložiska

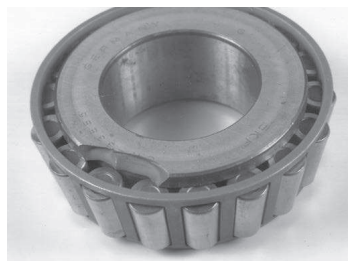


(b) Spalling za prohlubni

**Obr. 1.6:** Promáčklina a spalling ložiska [3]

## Lom/popraskání částí ložiska

K prasknutí může dojít v případě překročení meze pevnosti materiálu v tahu, například při úderu do ložiska. Dalším mechanismem je únavová prasklina při cyklickém ohýbání, která se šíří jako vlásečnicová prasklina skrz klec nebo kroužek. K prasknutí může dojít také následkem velkého nárůstu teploty při tření dvou materiálů při nedostatečné lubrikaci. [3]



(a) Lom ložiska způsobený nevhodným zacházením



(b) Únavová prasklina na vnějším kroužku

**Obr. 1.7:** Ukázka prasklin na ložisku [3]

## 1.2 Trvanlivost valivých ložisek

Za předpokladu naprosto ideálních provozních podmínek ložisek (dokonalá lubrikace, ocel, kontaminace atd.) by byla trvanlivost ložisek omezená vznikem únavy pod povrchem materiálu ložiska, která by vedla ke spallingu. [5] Základní stanovená trvanlivost ložiska  $L_{10}$  udává počet otáček v milionech pro daný model ložiska, při kterém 90 % stejných ložisek při stejných podmínkách dosáhne nebo překročí tuto hodnotu otáček, než dojde k únavě materiálu kroužků/valivých elementů. [6]

Tato hodnota je pro kuličková radiální ložiska dána vztahem 1.1 níže.

$$L_{10} = \left( \frac{C_r}{P_r} \right)^3 \quad (1.1)$$

Ve vztahu 1.1 je  $L_{10}$  základní trvanlivost v milionech otáček,  $C_r$  je základní radiální dynamická únosnost (konstantní radiální zatížení, které může ložisko přenášet při  $L_{10} = 10^6 RPM$ ) [N],  $P_r$  je ekvivalent dynamického radiálního zatížení (konstantní radiální zatížení, při němž dojde u ložiska k příznakům únavy za stejný počet otáček jako při skutečném zatížení) [N]. [6]

Rovnice 1.1 se dá při konstantních otáčkách přepočítat na trvanlivost v hodinách pomocí vztahu 1.2 níže.

$$L_{10h} = \frac{10^6}{60 \cdot RPM} \cdot L_{10} \quad (1.2)$$

Pro vztah 1.2 je  $L_{10h}$  základní trvanlivost v hodinách,  $RPM$  je počet otáček za minutu.

Výpočet  $C_r$  z rovnice 1.1 pro radiální kuličková ložiska s průměrem kuličky  $D_w \leq 25,4 \text{ mm}$  se určí pomocí vztahu 1.3 níže. [6]

$$C_r = b_m \cdot f_c \cdot (i \cdot \cos \alpha)^{0,7} \cdot Z^{2/3} \cdot D_w^{1,8} \quad (1.3)$$

Pro vztah 1.3 je  $C_r$  základní radiální dynamická únosnost [N],  $b_m$  je koeficient pro ocel závislý na typu a konstrukci ložiska (dán tabulkou v normě ČSN ISO 281 [6]),  $f_c$  je koeficient, který je daný geometrií komponent ložiska (dán tabulkou v normě ČSN ISO 281 [6]),  $i$  udává počet řad valivých elementů,  $\alpha$  je úhel styku [°],  $Z$  je počet valivých elementů a  $D_w$  je průměr kuličky [mm]. Podmínky platnosti tohoto vztahu jsou dále uvedeny v normě ČSN ISO 281 [6].

Určení  $P_r$  z rovnice 1.1 je dáno pomocí vztahu 1.4 níže.

$$P_r = X \cdot F_r + Y \cdot F_a \quad (1.4)$$

Pro vztah 1.4 je  $P_r$  ekvivalent dynamického radiálního zatížení [N],  $X/Y$  je koeficient radiálního/axiálního dynamického zatížení uvedený v tabulce v normě [6],  $F_r/F_a$  je radiální/axiální složka skutečného zatížení ložiska [N]. Při přílišném zatížení ložiska může dojít k plastické deformaci, jak je ukázáno na obrázku 1.6, je tedy nutné

vypočítat (norma ČSN ISO 76 [7]) nebo zjistit v katalogu výrobce i hodnotu  $C_{0r}$ , což je základní radiální statická únosnost [N]. [6]

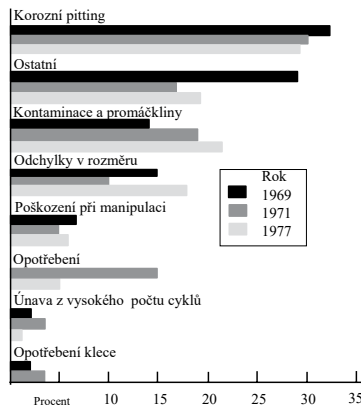
Trvanlivost je také možné určit při vyšší vyžadované spolehlivosti, než je základních 90 %, případně pokud provozujeme ložisko v ideálních nebo naopak nevhodných provozních podmínkách. Mezi ovlivňující faktory patří například míra zatížení, kvalita výroby a materiálu, kvalita mazání a znečištění částicemi. [6]

Pro tyto potřeby slouží vztah 1.5 níže pro výpočet modifikované trvanlivosti.

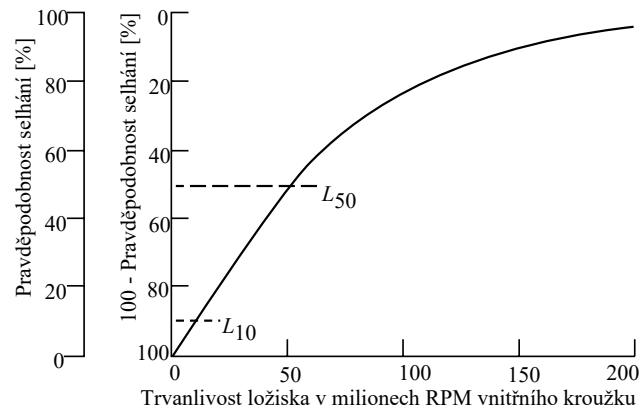
$$L_{nm} = a_1 \cdot a_{ISO} \cdot L_{10} \quad (1.5)$$

Pro vztah 1.5 je  $L_{nm}$  modifikovaná trvanlivost v milionech otáček,  $a_{ISO}$  je koeficient modifikované trvanlivosti a  $a_1$  je požadovaná spolehlivost (například pro spolehlivost 99 % odpovídá  $a_1 = 0,25$ ). Koeficient  $a_{ISO}$  je závislý na znečištění maziva, mazání, dynamickém zatížení ložiska, mezním únavovém zatížením, velikosti ložiska atd. Kompletní metodika je uvedena v [6]. Při reálné aplikaci použijeme hodnoty uvedené v katalogovém listu výrobce ložiska. Také je možné využít online kalkulačku přímo od výrobce ložiska, například SKF Bearing Select [8], která používá koncepty z normy ISO 281. [6]

Hodnota  $L_{10}$  vypočtená vztahem 1.1 počítá s poruchou ložiska způsobenou spalíngem, tedy únavovou poruchou, viz kapitola 1.1. Z obrázku 1.8 je patrné, že porucha ložiska je pouze v malém procentu případů způsobena spalíngem, v reálné aplikaci tedy dojde k poruše (projeví se např. hlučným chodem) většinou dříve. [5]



(a) Důvody selhání ložiska



(b) Spolehlivost vs Trvanlivost ložiska

**Obr. 1.8:** Faktory ovlivňující trvanlivost ložisek [5], přeloženo a upraveno

## 1.3 Vibrace ložisek

Pokud chceme určit stav ložiska, musíme nějakým způsobem monitorovat jeho vibrace. Měření vibrací ložiska se nejčastěji provádí pomocí akcelerometru [9], který je umístěn ideálně co nejblíže ložiska a na vhodném místě.

V momentě, kdy valivý element narazí na poškozené místo v ložisku, dojde k vytvoření krátkého impulsu. Tento impuls se propaguje přes ložisko a domek ložiska a vybudí v něm rezonance. [10]

Místo, kde se nachází poškození, ovlivňuje podobu naměřeného signálu. Z tohoto signálu se poté dá metodami diagnostiky určit, která komponenta ložiska je poškozena.

Z geometrických rozměrů a parametrů ložiska byly odvozeny vztahy pro výpočet tzv. frekvencí ložiskových závad. Z těchto vztahů lze po dosazení parametrů ložiska, které nalezneme v odpovídajícím katalogovém listě a z frekvence otáčení hřídele ložiska vypočítat frekvence, na kterých by se měla objevit konkrétní závada.

Rozlišují se čtyři základní frekvence závad [1]:

- **BPFI** (Ball Pass Frequency Inner Race – závada na vnitřním kroužku)
- **BPFO** (Ball Pass Frequency Outer Race – závada na vnějším kroužku)
- **BSF** (Ball Spin Frequency – závada na valivém elementu)
- **FTF** (Fundamental Train Frequency – závada na kleci)

Následující vztahy platí v případě, kdy je vnitřní kroužek v pohybu a vnější statický.

$$f_{BPFI} = \frac{N}{2} \cdot \left(1 + \frac{B_d}{P_d} \cdot \cos\varphi\right) \cdot f_s \quad (1.6)$$

$$f_{BPFO} = \frac{N}{2} \cdot \left(1 - \frac{B_d}{P_d} \cdot \cos\varphi\right) \cdot f_s = N \cdot FTF \quad (1.7)$$

$$f_{BSF} = \frac{P_d}{2B_d} \cdot \left(1 - \left(\frac{B_d}{P_d} \cdot \cos\varphi\right)^2\right) \cdot f_s \quad (1.8)$$

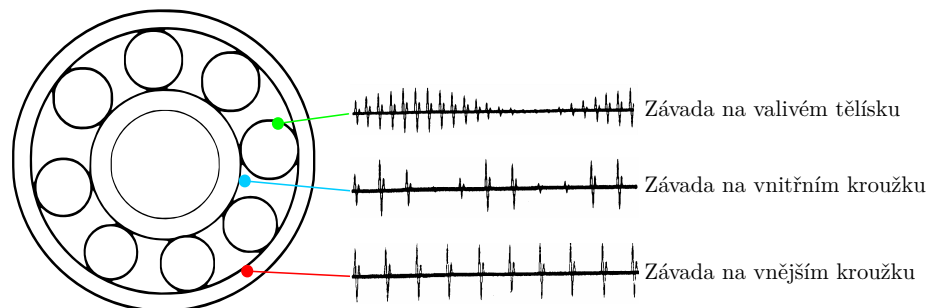
$$f_{FTF} = \frac{1}{2} \cdot \left(1 - \frac{B_d}{P_d} \cdot \cos\varphi\right) \cdot f_s \quad (1.9)$$

Pro vztahy 1.6–1.9 výše je  $f_s$  frekvence hřídele [Hz],  $N$  je počet valivých elementů [-],  $B_d$  je průměr valivého elementu [mm],  $P_d$  je roztečný průměr [mm] a  $\varphi$  je kontaktní úhel [°].

Frekvenční spektrum periodického impulsu s periodou  $T$  je v celém frekvenčním rozsahu složeno z diskretních spektrálních čar, frekvenční interval mezi těmito čarami je roven  $1/T$  Hz. Opakované impulsy způsobené lokálním poškozením na dané komponentě ložiska se ve spektru naměřeného vibračního signálu projeví jako harmonické tohoto opakovaného impulsu. Tento jev se dá využít v metodách identifikace poruchy. Každá mechanická struktura vykazuje na určitých frekvencích rezonance. Přirozeně tak při vzniku impulsu na závadě dojde k vyššímu vybudení vibrací právě

na rezonančních frekvencích. Ve spektru se objeví pásmo se zvýšenou amplitudou kolem strukturálních rezonancí, po demodulaci tohoto pásma získáme frekvence závad a jejich harmonické. [11]

Výše bylo zmíněno, že když se ložisko otáčí konstantní frekvencí, projeví se závada jako pravidelně opakující se impulsy. Pro následující výklad je předpokládáno, že vnější kroužek je statický a vnitřní se otáčí s konstantní frekvencí  $f_s$ . Charakter naměřených impulsů ve vibračním signálu je ovlivněn místem, kde se závada nachází, zatížením ložiska, vzájemnou polohou závady a snímače vibrací, výrobními nedokonalostmi ložiska a nevyváhou hřídele. Výrobní nedokonalosti ložiska (chyba průměru) se projeví při předpětí (valivé elementy jsou pevně umístěny v drahách). [12]



Obr. 1.9: Projev závad jednotlivých komponent [1]

### 1.3.1 Vliv závad ložiska na spektrum signálu

#### Závada na vnějším kroužku

V případě, že se defekt ložiska nachází na vnějším kroužku, je cesta mezi místem vzniku impulsu a snímačem vibrací neměnná (vnější kroužek je upevněn), statické zatížení ložiska se tedy v naměřeném signálu neprojeví. Frekvenční složka s nejvyšší amplitudou bude na frekvenci BPFO, v případě nevyváhy hřídele bude docházet k zatížení ložiska s periodou rovnou frekvenci otáčení hřídele  $f_s$ , ve spektru by tedy měly být kolem frekvence BPFO postranní pásma s rozestupem o frekvenci  $f_s$ . Pokud je ložisko uloženo v předpětí a je u něj výrazná výrobní chyba průměru, bude docházet k tvorbě postranních pásem s rozestupem o frekvenci FTF (Frekvence klece). [12]

#### Závada na vnitřním kroužku

V případě, že se defekt ložiska nachází na vnitřním kroužku, dochází k rotaci defektu s frekvencí  $f_s$ , takže impuls je generován v různých místech od měřícího bodu

vibrací. Vzhledem k tomu, že u ložiska je zatížení rozloženo do určité zóny, bude docházet k vybuzení impulsu různé síly v závislosti na frekvenci  $f_s$ . Frekvenční složka s nejvyšší amplitudou bude na frekvenci BPFI. Pokud dosahuje ložisko výrazné chyby průměru, budou kolem frekvence BPFI postranní pásma s rozestupem daným rozdílem  $f_s$  a frekvence FTF. Vzhledem k výše zmíněné zóně zatížení ložiska budou postranní pásma také s rozestupem o frekvenci  $f_s$ . [12]

### Závada na valivém elementu

V případě defektu na valivém elementu bude frekvenční složka s nejvyšší amplitudou na frekvenci BSF, jelikož dojde k vygenerování impulsu při kontaktu s vnitřním i vnějším kroužkem za jednu otáčku valivého elementu. Při statickém zatížení dojde ke generování postranních pásem s rozestupem o frekvenci FTF, jelikož s touto frekvencí se bude poškozený element dostávat do oblasti zátěže. V případě nevyváhy hřídele dochází k relativnímu pohybu mezi hřídelí a valivým elementem v kleci s frekvencí  $f_s$  a frekvencí FTF, takže postranní pásma kolem frekvence BSF budou s rozestupem  $f_s - \text{FTF}$ . [12]

V případě, že se na ložisku vyskytne více defektů současně, bude vibrační signál tvořen sumou příspěvků jednotlivých defektů v závislosti na jejich fázi. V případě opačné fáze impulsů (např. defekty posunuty na kroužku o  $180^\circ$ ) může tedy dojít i k jejich utlumení. [12]

Očekávané frekvence spektrálních čar v závislosti na defektní komponentě jsou shrnuty v tabulce 1.1 níže.

**Tab. 1.1:** Vliv závad na frekvenční spektrum signálu [12]

Poloha spektrálních čar	Místo defektu		
	Vnitřní dráha	Vnější dráha	Valivý element
Frekvence hlavních spektrálních čar	<b>BPFI</b>	<b>BPFO</b>	<b>BSF</b>
Rozestup bočních pásem	$f_s$ $f_s - \text{FTF}$	$f_s$ FTF	FTF $f_s - \text{FTF}$ BSF/2

## 2 Přehled metod pro diagnostiku ložisek

Pro ověření stavu ložisek existuje mnoho metod. Metody popsané v této práci se zaměřují na zpracování vibračního signálu z akcelerometru. Tyto metody se dají nejobecněji rozdělit následujícím způsobem:

- Analýza signálu v časové oblasti (RMS, Crest faktor, špičatost atd.)
- Analýza signálu ve frekvenční oblasti (FFT, Obálková analýza, Kepstrální analýza atd.)
- Analýza signálu v časově-frekvenční oblasti (STFT, Vlnková transformace atd.)

### 2.1 Analýza signálu v časové oblasti

#### 2.1.1 Analýza časového průběhu vibrací

Nejvíce přímým přístupem pro zjištění přítomnosti závady je kontrola časového průběhu vibrací. V grafu 2.1 jsou vykreslena data naměřená v Case School of Engineering [13] pro ložisko s poškozeným vnějším kroužkem. V časovém průběhu se nachází zřetelný opakovaný impuls. Perioda opakování je téměř shodná s vypočítanou frekvencí závady na vnějším kroužku (BPFO) pomocí vztahu 1.7. Tato metoda je sice teoreticky funkční, ale je nutné manuálně přibližovat časový průběh a počítat časové intervaly, proto se dá jen stěží automatizovat. Hledání impulsů by bylo navíc mnohem komplikovanější v zašuměném signálu.



Obr. 2.1: Projev závady na ložisku v časové oblasti

## 2.1.2 Hodnocení podle hodnoty RMS

Pro zjištění obecného stavu ložiska bez nutnosti klasifikace poruchy slouží techniky statistického zpracování signálu, jako je například výpočet RMS (Root Mean Square – efektivní hodnota signálu). Sledováním efektivní hodnoty vibrací lze hodnotit vývoj poškození zařízení v čase. Úrovně vibrací pro různé velikosti zařízení jsou uvedeny v normě ČSN ISO 10816-3 [14]. Pro stroje s jmenovitým výkonem v rozsahu  $300 \text{ kW} \geq P_{jmen.} > 15 \text{ kW}$  jsou zde uvedena čtyři pásma – A, B, C a D, která slouží ke kvalitativnímu posouzení úrovně vibrací. Hodnoty náležící do těchto pásem slouží k posouzení při měření širokopásmových vibrací na ložiskách při ustálených podmínkách. [14]

- Pásmo A: obvykle vibrace nových přejímaných strojů.
- Pásmo B: stroje provozuschopné po neomezeně dlouhou dobu.
- Pásmo C: stroje neuspokojivé pro dlouhodobý trvalý provoz (mohou být provozovány do té doby, než nastane vhodná příležitost k opravě)
- Pásmo D: nebezpečné vibrace, mohou způsobit poškození stroje.

Pro rozdělení do pásem vibrací podle efektivní hodnoty měřené veličiny slouží tabulka 2.1.

Hodnoty uvedené v tabulce platí pro efektivní hodnoty rychlosti/výchytky ve frekvenčním pásmu od 10 Hz–1000 Hz (otáčky nad 600 RPM), případně 2 Hz–1000 Hz (otáčky pod 600 RPM). [14]

**Tab. 2.1:** Klasifikace pásem mohutnosti vibrací podle ČSN ISO 10816-3 [14]

Klasifikace uložení	Hranice pásma	Ef. hodnota výchytky [ $\mu\text{m}$ ]	Ef. hodnota rychlosti [ $\text{mm/s}$ ]
Tuhé	A/B	22	1,4
	B/C	45	2,8
	C/D	71	4,5
Pružné	A/B	37	2,3
	B/C	71	4,5
	C/D	113	7,1

RMS diskrétního signálu lze vypočítat pomocí vztahu 2.1 následujícího níže.

$$x_{RMS} = \sqrt{\frac{1}{N} \cdot \sum_{i=1}^N x_i^2} \quad (2.1)$$

Pro vztah 2.1 je  $x_{RMS}$  vypočtená efektivní hodnota,  $N$  je počet prvků,  $x_i$  je  $i$ -tý prvek naměřeného signálu.

Na zařízení pro testování ložisek v [15] byl zjištěn rostoucí trend RMS hodnoty pro nepoškozené ložisko se zvyšujícími se otáčkami (rozsah 400–1400 RPM). Pro ložisko s uměle vytvořeným poškozením docházelo také k růstu hodnoty RMS s růstem otáček zařízení.



### 2.1.3 Hodnocení s použitím špičatosti (Kurtosis)

Dalším statistickým parametrem, který lze u vibračního signálu určit je špičatost (Kurtosis).

Matematicky je špičatost definována ve vztahu 2.2 níže.

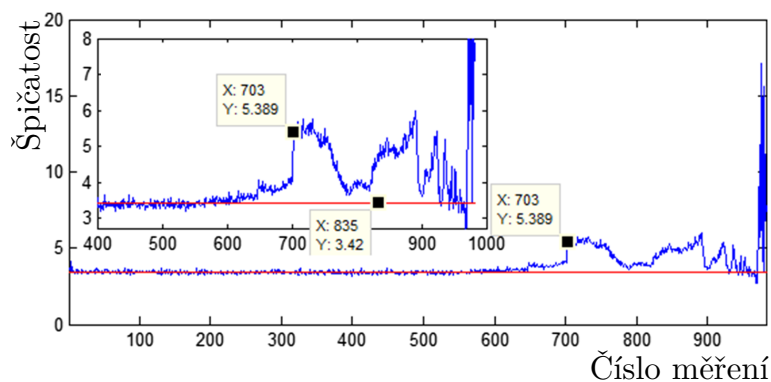
$$K = \frac{1}{N \cdot \sigma^4} \cdot \sum_{i=1}^N (x_i - \bar{x})^4 \quad (2.2)$$

$\sigma$  je směrodatná odchylka definována ve vztahu 2.3 níže.

$$\sigma = \sqrt{\frac{1}{N} \cdot \sum_{i=1}^N (x_i - \bar{x})^2} \quad (2.3)$$

Pro vztahy 2.2 a 2.3 je  $N$  počet prvků a  $\bar{x}$  je průměrná hodnota.

Statistická metoda špičatosti využívá toho, že zdravé ložisko by mělo mít hustotu pravděpodobnosti naměřeného vibračního signálu velmi podobnou normálnímu rozložení. Normální rozložení má špičatost 3. Když se u ložiska rozvine závada, dojde k zvýšení hodnoty špičatosti. Na obrázku 2.2 je vidět vývoj tohoto indikátoru po celý životní cyklus ložiska (až do zničení), na konci životního cyklu dojde dokonce k poklesu pod zdravou hodnotu, je tedy nutné sledovat trend tohoto indikátoru pro korektní vyhodnocení stavu.



**Obr. 2.2:** Časový vývoj hodnoty špičatosti [16], přeloženo

V [15] bylo na zařízení pro testování ložisek dokázáno, že parametr špičatost má pro nepoškozené ložisko v rozsahu 3000–6000 RPM se zvyšujícími otáčkami mírně klesající charakter (pokles o cca 10 %). V [17] bylo v rozsahu 400–1400 RPM ukázáno, že parametr špičatosti dokáže dobře odhalit bodové promáčknutí na vnějším kroužku ložiska, ale nedokáže odhalit korozi kroužku, stejně jako poškrábání po celé ploše vnějšího kroužku. Pro hodnocení stavu zařízení se v [17] více osvědčilo použití hodnoty zrychlení než rychlosti vibrací.

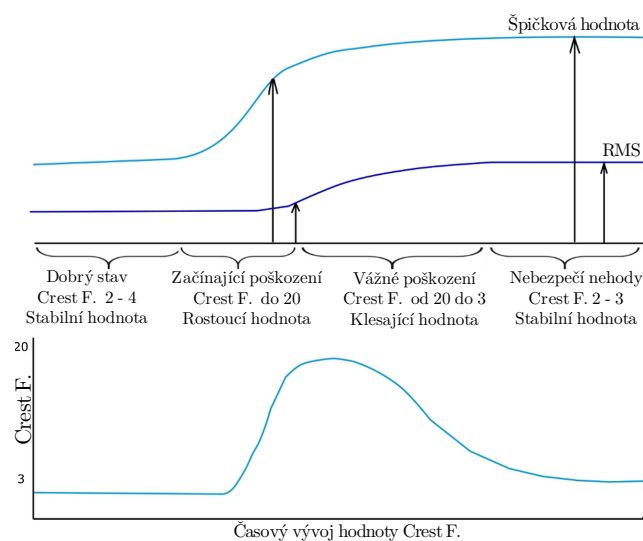
## 2.1.4 Hodnocení s použitím crest faktoru

Crest faktor je vhodný pro monitorování vývoje stavu zařízení v čase. Jde o poměr špičkové hodnoty ku efektivní hodnotě měřených vibrací. U nepoškozeného ložiska je hodnota crest faktoru kolem hodnoty 3. Když se začne rozvíjet poškození ložiska, dojde ke generování krátkých pulzů, které ovšem nemají značný vliv na hodnotu RMS, tudíž dojde k vzrůstu hodnoty Crest faktoru. Při dalším rozvoji poškození ložiska se začne počet pulsů v měřeném signálu zvětšovat dostatečně na to, aby došlo k zvýšení efektivní hodnoty, dojde tak k poklesu hodnoty crest faktoru. Proto je pro správné vyhodnocení crest faktoru nutno sledovat jeho vývoj v čase, případně používat současně i ostatní metody diagnostiky. Teoretický vývoj hodnoty crest faktoru v čase je zobrazen na obrázku 2.3.

Vztah 2.4 pro výpočet crest faktoru je níže.

$$c_f = \frac{|x_{peak}|}{x_{RMS}} \quad (2.4)$$

Pro 2.4 je  $c_f$  crest faktor,  $x_{peak}$  je špičková hodnota měřeného signálu,  $x_{RMS}$  je efektivní hodnota, jak bylo popsáno vztahem 2.1.



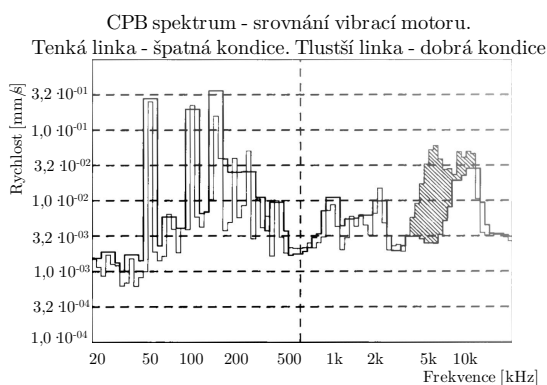
**Obr. 2.3:** Časový vývoj hodnoty crest faktoru [18], přeloženo

V [15] bylo na zařízení pro testování ložisek dokázáno, že parametr špičkatosti má pro nepoškozené ložisko v rozsahu 3000–6000 RPM s rostoucími otáčkami mírně klesající charakter pro osu  $x$  akcelerometru, pro osy  $y$  a  $z$  docházelo v rozsahu 3000–4000 RPM k růstu, poté už tento parametr měl podobný průběh pro všechny tři osy.

## 2.2 Analýza signálu ve frekvenční oblasti

### 2.2.1 Spektrální analýza

V kapitole 1.3 byly vypsány frekvence, na kterých se u ložisek projevují závady na jednotlivých komponentech. Z tohoto výčtu by se mohlo zdát, že stačí ve spektru zkoumat frekvence příslušící závadám. Tyto závady se ale pro standardní rychlosti otáčení hřídele pohybují v nízkých frekvencích, typicky do 1000 Hz. V tomto frekvenčním pásmu se ale můžou u reálného zařízení objevovat vibrace jiných komponent, frekvence ložiskových závad se tak ve spektru reálného zařízení lehce ztratí, jelikož mají malou amplitudu. Díky monitorování trendu spektra ale lze sledovat změny ve frekvenčním složení vibrací, při zhoršení stavu dojde k nárůstu pásem na vysokých frekvencích. [19]



**Obr. 2.4:** Porovnání spektra před a po poškození [19], přeloženo a upraveno

Tento nárůst na vyšších frekvencích je vidět na obrázku 2.4. Jsou zde překryta dvě spektra pro vibrace na ložisku. Tenkou čarou je zobrazeno spektrum 6 týdnů před vyhořením ložiska, tlustou čarou je referenční spektrum dobré kondice. Je pozorovatelný nárůst energie na vysokých frekvencích (vyšrafovaná plocha). [19]

## 2.2.2 Obálková analýza

Jak bylo zmíněno v kapitole 1.3.1, závady na jednotlivých komponentech ložiska se díky vybuzení strukturálních rezonancí projeví také na vyšších frekvencích, než jsou samotné frekvence ložiskových závad. Tohoto jevu využívá metoda obálkové analýzy. Ve frekvenčním spektru vibračního signálu vybereme pomocí pásmové propusti oblast, kde se dostatečně silně projeví rezonance, ale je vzdálená od nižšího frekvenčního pásma, kde se projeví složky od ostatních komponent zařízení. Z tohoto úzkého frekvenčního pásma můžeme poté extrahovat informace o možné přítomnosti závady na ložiskové komponentě, případně závadu klasifikovat. [20]

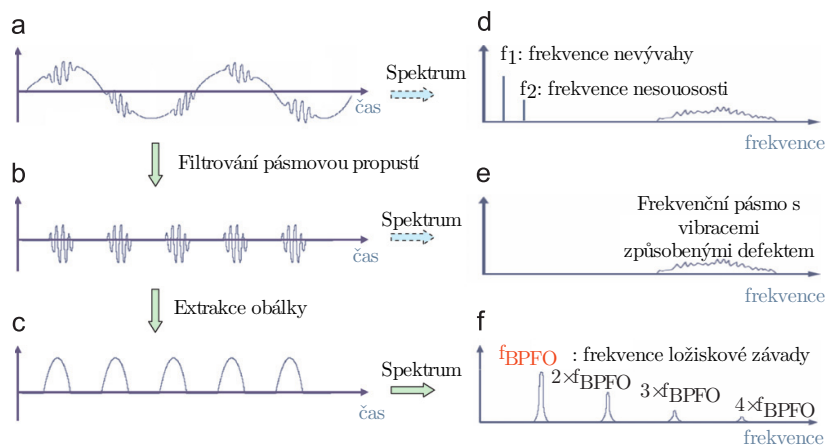
Z vyfiltrovaného signálu získáme obálku například vypočtením modulu analytického signálu.

Analytický signál je získán aplikací následující rovnice

$$a_s = x(t) + j \cdot \mathcal{H}_t\{x(t)\} \quad (2.5)$$

kde  $a_s$  označuje analytický signál,  $x(t)$  označuje zkoumaný časový průběh a  $\mathcal{H}_t\{x(t)\}$  označuje aplikaci Hilbertovy transformace na časový průběh. Komplexní část analytického signálu získáme aplikací Hilbertovy transformace. Získáním obálky jsme demodulovali signál, spektrum obálky poté může obsahovat frekvence potenciálních ložiskových závad, postranní pásma a jejich harmonické.

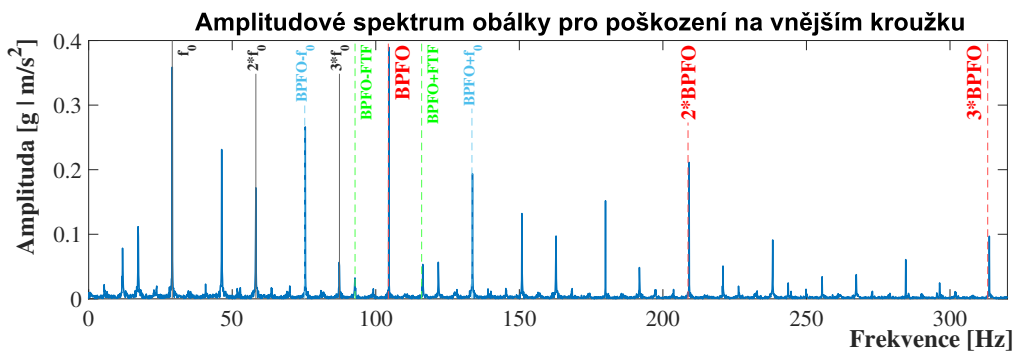
Základní postup je zobrazen na obrázku 2.5.



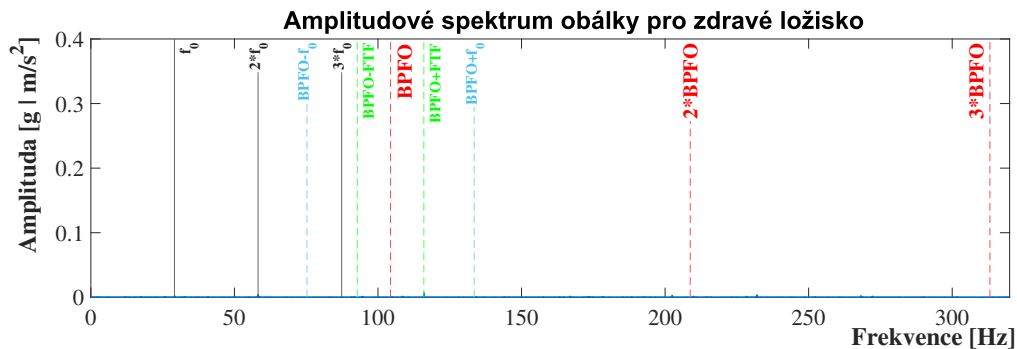
**Obr. 2.5:** Obecný postup při obálkové analýze [21], přeloženo

### 2.2.3 Implementace obávkové analýzy

V sekci 2.2.2 byl popsán stručný postup při obávkové analýze. Tato metoda byla implementována v programu MATLAB, který disponuje přímo funkcí *envspectrum* pro výpočet obávkové analýzy. V této funkci lze nastavit metodu výpočtu obálky jako výpočet z analytického signálu pomocí Hilbertovy transformace (viz. vztah 2.5) a také parametry FIR pásmové propusti – řád, frekvenci počátku a konce pásma propusti. Na následujícím grafu 2.6 je obávkové spektrum pro ložisko s poškozením na vnějším kroužku (průměr 0,53 mm) a zátěž 2HP u dat z Case Western Reserve University [13] a pro stejné provozní podmínky také pro ložisko zdravé. Frekvenční pásmo pro pásmovou propust bylo zvoleno v rozsahu 2,2 kHz–3,1 kHz. V tomto pásmu dosahuje vibrační signál dostatečné amplitudy a zároveň už by se na těchto vyšších frekvencích už neměly vyskytovat ostatní rušivé signály, jak bylo uvedeno v kapitole 2.2.2.



(a) Poškozené ložisko



(b) Zdravé ložisko

**Obr. 2.6:** Obávková metoda pro ložisko s vadou na vnějším kroužku a zdravé ložisko

Barevnými svislými čarami jsou označeny vypočtené očekávané frekvence ložiskových závad získaných z rovnic 1.6-1.9 a ostatní očekávané frekvence, které jsou shrnuty v tabulce 1.1. K výpočtu těchto frekvencí je nutné znát otáčkovou frekvenci hřídele  $f_s$ , jejíž přibližná hodnota lze zjistit na frekvenčním měniči motoru a přesná

poté ověřit na spektru vibračního signálu, případně můžeme do měřicího systému přidat snímač otáček.

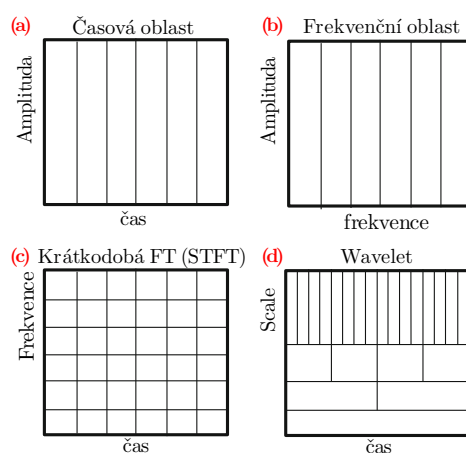
Z grafu 2.6 (a) je patrné, že obálkovému spektru dominuje frekvence  $f_{BPFO}$  a její harmonické, jde o frekvenci naznačující existující závadu na vnějším kroužku. Jsou zde také patrné harmonické otáčkové frekvence hřídele  $f_s$ . Pro stejné zatížení totožného ložiska, které nemělo zanesenou vadu na žádné komponentě, je amplitudové spektrum obálky pro stejné nastavení pásmové propusti v grafu 2.6 (b).

Ke grafu spektra zdravého ložiska 2.6 (b) je důležitá poznámka, že osa  $y$  má nastavené stejné měřítko jako u grafu 2.6 (a) výše. Při tomto měřítku nejsou ve spektru patrné téměř žádné složky. Tento fakt potvrzuje dobrou schopnost zjištění typu závady pro metodu obálkové analýzy.

## 2.3 Analýza signálu v časově-frekvenční oblasti

Aplikováním Fourierovy transformace na časový vibrační signál získáme přesný přehled o tom, jaké frekvence se v daném časovém úseku nacházejí. Bohužel dojde také k tomu, že nemáme žádnou informaci o tom, v jakém konkrétním čase se daná frekvenční složka nachází, tedy kromě toho, že se nachází v analyzovaném bloku dat. Toto není problém pro stacionární periodické signály nebo signály jim blízké.

Při reálném měření vibrací ale dochází i při ustálených podmínkách k drobným fluktuacím otáčkové frekvence hřídele  $f_s$ , stejně jako impulsy v signálu se mohou vyskytovat v různém čase. Při přechodných jevech, jako je spuštění a vypnutí zařízení či změna otáček, je spektrum získané aplikací Fourierovy transformace pro diagnostiku nepoužitelné.



Obr. 2.7: Srovnání vlastností jednotlivých postupů [22], přeloženo

Tento problém řeší metody časově-frekvenční analýzy (na obrázku 2.7 jde o možnosti c) a d)).

### 2.3.1 Short Time Fourier Transform (STFT)

Krátkodobá Fourierova transformace nám k frekvenční informaci signálu dodává i informaci o čase. Jedná se pouze o úpravu klasické Fourierovy transformace, kdy je analyzovaný signál rozdělen okénkovou funkcí na jednotlivé úseky. Délka okna poté rozhoduje o rozlišení v čase/frekvenci. Po délku okna považujeme zkoumaný signál za stacionární. Zvolením širokého okna získáme větší rozlišení ve frekvenci, ale horší časovou lokalizaci. Při krátkém okně získáme dobrou časovou lokalizaci, ale hrubé rozlišení ve frekvenci. Při rozhodování, zda preferujeme vyšší rozlišení v časové nebo frekvenční oblasti, jsme limitováni principem neurčitosti. Pro optimální rozlišení v čase i frekvenci slouží speciální případ STFT – Gaborova transformace.

V tomto případě se jako okno používá Gaussova funkce. Tento případ je zobrazen na obrázku 2.7c). [23]

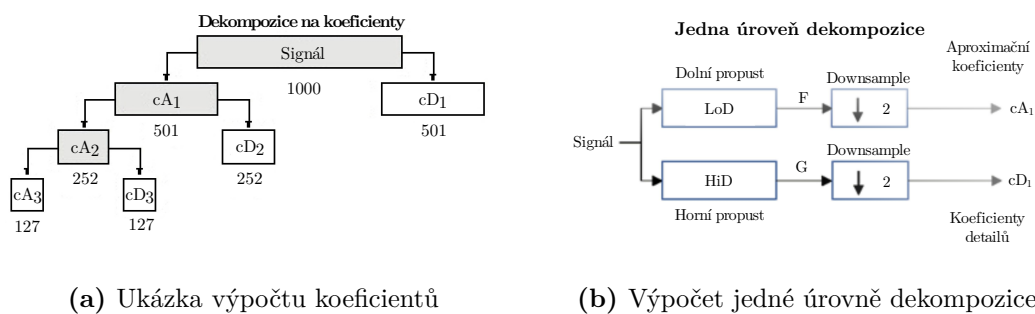
### 2.3.2 Vlnková transformace

Vlnková transformace (wavelet transform) do určité míry řeší problém s konstantním časově-frekvenčním rozlišením. Složky o malé frekvenci považujeme za v čase příliš neměnné, naopak složky o vysokých frekvencích se v čase mění rychle. Tohoto poznatku využívá vlnková transformace, pro nízké frekvence má nízké rozlišení v čase a pro vysoké frekvence má časovou lokalizaci přesnou. Tato transformace má tedy variabilní časově-frekvenční rozlišení. Pro vlnkovou transformaci používáme vlnku, což je funkce, která se dá v čase posouvat (translace) a také smršťovat a natahovat (scale). Časový průběh zkoumaného signálu porovnáváme s různě dilatovanou a posunutou vlnkou. Při zkoumání jednorozměrného signálu tak získáme dvourozměrnou reprezentaci signálu s parametry posunutí a natažení, kdy natažení je reprezentací frekvence. [23]

#### Rychlá vlnková transformace (FWT)

Při aplikaci spojité vlnkové transformace tak, jak byla popsána výše je možné volit parametry časového posunutí a rozměrové dilatace vlnky libovolně. Při spojitě volbě těchto parametrů by došlo k redundantní reprezentaci signálu.

Pro dekompozici signálu v programu MATLAB se používá Mallatův algoritmus pro rychlou vlnkovou transformaci, viz [24]. Tento algoritmus využívá v jednotlivých úrovních dvojici dolních a horní propustí (filtrů) pro dekompozici, která se nazývá diskrétní vlnková transformace (DWT) a zpětnou rekonstrukci, která se nazývá jako inverzní vlnková transformace (IDWT) signálu. Dekompozice je zobrazena na ob-



**Obr. 2.8:** Ukázka algoritmu FWT [25, 26], upraveno a přeloženo

rázku 2.8. Dekompozice může být provedena v několika úrovních, které vzniknou zařazení schématu z obrázku 2.8 (b) za sebe ( $cA_j$  je vstupem do další banky filtrů).

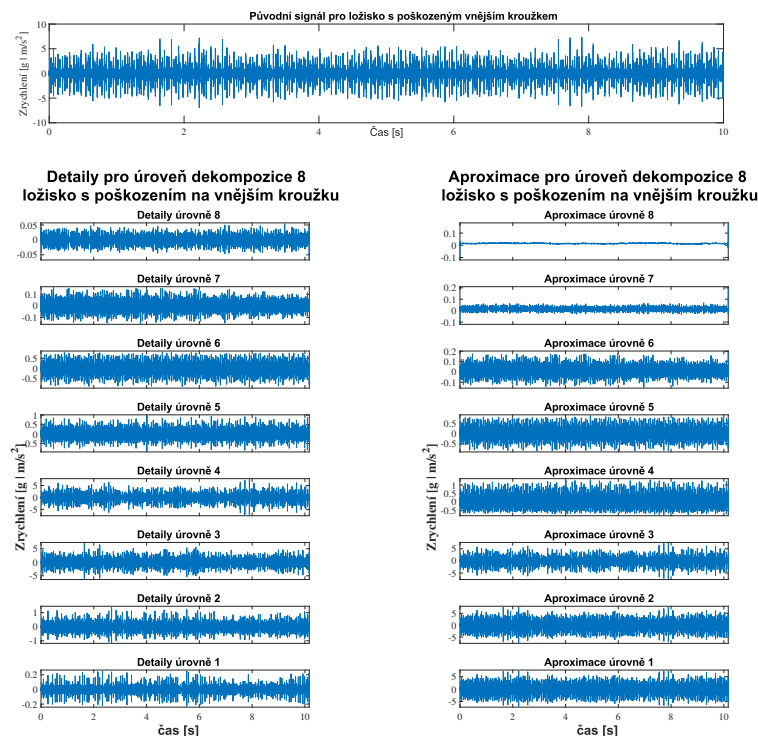


Následující popis vlnkové transformace je podle implementace MATLABu, viz [28]. V jednotlivých úrovních dekompozice je provedena konvoluce signálu s koeficienty filtru. Z výsledku jsou poté vypuštěny liché vzorky, na obrázku 2.9 (b) je tento krok označen jako *downsample*. Výsledkem jsou koeficienty aproximace a detailů. Tyto koeficienty mají pro jednotlivé úrovně dekompozice díky vypuštění lichých prvků zhruba poloviční délku. Z koeficientů detailů a koeficientů aproximace lze provést rekonstrukci pomocí IDWT. Ta se provede jako filtrace s odpovídajícími rekonstrukčními filtry a vložením nul mezi prvky (tento krok se anglicky označuje jako *upsampling*) získat detaily a aproximace signálu pro zvolené úrovně dekompozice. Aproximace a detaily jsou úplnou reprezentací signálu, mají stejnou délku jako původní dekomponovaný signál a jejich součtem dojde k rekonstrukci původního signálu. Pro případ dekompozice z obrázku 2.8 (a) je po provedení IDWT pro tři úrovně koeficientů detailů a koeficientů aproximace signál reprezentován součtem detailů a aproximací následovně:

$$S = A_1 + D_1 = A_2 + D_2 + D_1 = A_3 + D_3 + D_2 + D_1 \quad (2.6)$$

Ve vztahu 2.6 označuje  $S$  původní signál,  $A_i$  označuje aproximace pro  $i$ -tou úroveň dekompozice a  $D_i$  označuje detaily pro  $i$ -tou úroveň dekompozice. [27, 28]

V grafu 2.9 je zobrazena dekompozice s hloubkou 8 pro data z CWRU [13].



Obr. 2.9: Původní signál a jeho rozklad na aproximace a detaily

### 2.3.3 Implementace vlnkové dekompozice

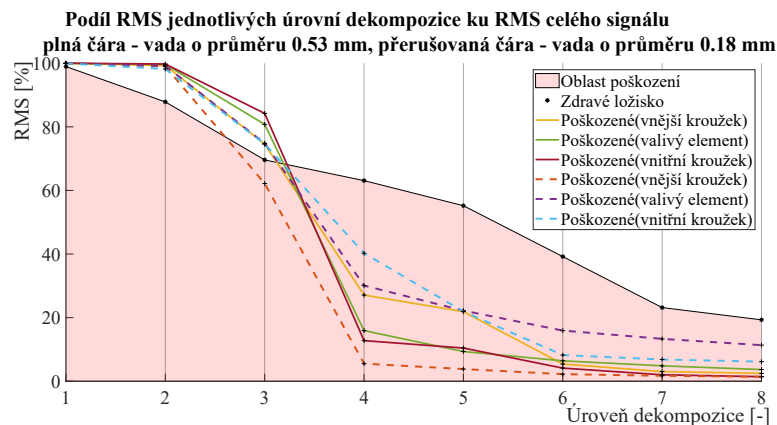
Vlnkovou dekompozici, která byla blíže popsána v sekci 2.3.2, využívá i článek [29]. V tomto článku používají vlnku typu Daubechies 4 s hloubkou dekompozice rovnou osmi. Z osmi získaných aproximací je pro každou z nich určena efektivní hodnota  $RMS(A_i)$ .

$$RMS\% = \frac{100 \cdot RMS(A_i)}{RMS(S)} \quad (2.7)$$

Pro vztah 2.7 je  $RMS\%$  podíl energie aproximace  $i$ -té úrovně k celkové energii signálu,  $RMS(A_i)$  efektivní hodnotu  $i$ -té úrovně aproximace a  $RMS(S)$  označuje efektivní hodnotu zkoumaného signálu  $S$ .

Poté je určen procentuální podíl efektivní hodnoty jednotlivých úrovní k efektivní hodnotě celkového signálu. Při aplikaci této metody je nutné mít naměřená data zdravého ložiska. Pokud leží křivka procentuálního podílu efektivní hodnoty aktuálně měřeného ložiska pod křivkou zdravého ložiska, můžeme dle autorů studie považovat zkoumané ložisko za poškozené. Účinnost této metody je závislá na otáčkách, nejlépe funguje pro vyšší otáčky. Je také nutné mít naměřené vibrace na zdravých ložiskách pro všechny očekávané stavy zařízení pro možné porovnání (tedy pro všechny uvažované otáčky a zatížení).

Grafický výstup z této metody pro data z Case Western Reserve University [13] je v grafu 2.10 níže.



**Obr. 2.10:** Zobrazení výstupu RMS vlnkové dekompozice

Poškození ložisek zkoumaných v grafu 2.10 jsou o průměru 0,53 mm a 0,18 mm při zatížení 2 HP a cca 1750 otáčkách za minutu. Pokud zkoumaný průběh spadá pod referenční hodnotu na zdravém ložisku (v grafu vybarvena růžovou barvou), může být dané ložisko považováno podle [29] za poškozené. Z grafu je vidět, že tato metoda odhalila všechny typy poškození jak pro velkou vadu 0,53 mm, tak pro malou vadu 0,18 mm už u čtvrté úrovně dekompozice.

## 3 Algoritmy strojového učení

Pro klasifikaci místa a velikosti závady je možné použít algoritmy strojového učení, které po vytrénování na označených datech provádí klasifikaci místa a závady automaticky. V praktické části této práce byly použity algoritmy popsané níže.

### 3.1 Standardní algoritmy strojového učení

#### 3.1.1 k-NN (k-Nearest Neighbors)

k-NN je zástupce neparаметrických metod, tzv. učení založeného na instancích/založené na paměti. Hlavní ideou této metody je, že pokud jsou si pozorování podobná, pravděpodobně patří do stejné třídy. Aby bylo možné provést klasifikaci nového pozorování, dojde nejdříve k uložení předchozích pozorování/trénovací množiny. Podobnost nového pozorování s trénovací množinou je hodnocena pomocí vzdálenosti. Model k-NN se tedy neučí při definování trénovací množiny, ale výpočet probíhá až při klasifikaci nového pozorování. [30]

Vzdálenost lze definovat různými metrikami, například Euklidovská vzdálenost nebo Manhattan vzdálenost.

Vztahy 3.1 a 3.2 popisují vybrané metriky vzdálenosti jednoho páru pozorování.

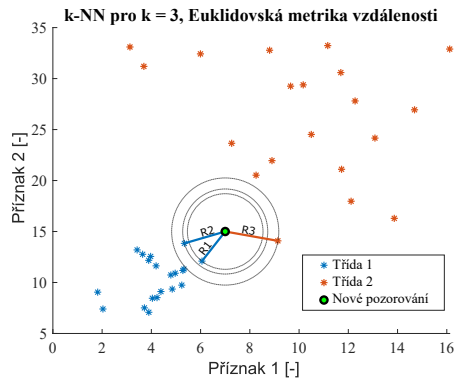
$$d_{Euklid} = \sqrt{\sum_{i=1}^n (p_{i(nové)} - p_{i(trénovací)})^2} \quad (3.1)$$

$$d_{Manhattan} = \sum_{i=1}^n |p_{i(nové)} - p_{i(trénovací)}| \quad (3.2)$$

Pro vztahy 3.1 a 3.2 je  $n$  počet příznaků,  $p_{i(nové)}$  značí  $i$ -tý příznak nového pozorování,  $p_{i(trénovací)}$  značí  $i$ -tý příznak z trénovací množiny.

Po výpočtu vzdáleností všech pozorování z trénovací množiny od nového pozorování seřadíme výsledky od nejmenšího. Z  $k$  nejmenších vybereme nejčastěji zastoupenou třídu. Poté považujeme nové pozorování za pozorování z této třídy. [31]

V grafu 3.1 je zobrazena situace pro prostor dvou příznaků,  $k = 3$  a Euklidovskou metriku vzdálenosti. Tři nejmenší vzdálenosti nového pozorování z neznámé třídy od pozorování ze známých tříd seřazené podle velikosti jsou R1, R2, R3. Pro Euklidovskou metriku a dva příznaky se dá vzdálenost vyjádřit jako kružnice se středem v novém pozorování. Je vidět, že nejčastější třídou je třída 1, nové pozorování tedy náleží do této třídy.



Obr. 3.1: Demonstrace k-NN pro  $k = 3$

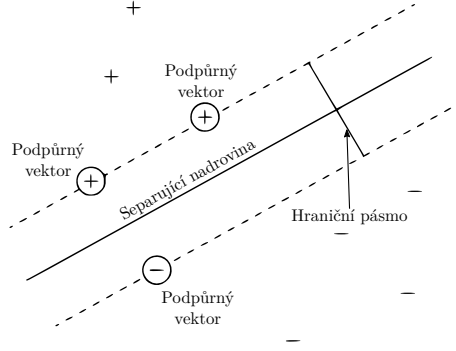
### 3.1.2 SVM (Support Vector Machines)

V základní podobě slouží lineární SVM k separování dvou tříd, označených jako  $-1$  a  $1$ . Na obrázku 3.2 jsou tyto třídy reprezentovány v prostoru dvou příznaků jako znaménka  $+$ ,  $-$ . Pozorování v grafu jsou lineárně separovatelná nadrovinou, úkolem SVM je nalezení ideální nadroviny, v tomto případě přímky (v případě  $N$ -rozměrného prostoru daného  $N$  příznaky se jedná o  $N - 1$  rozměrnou nadrovinu). Přímka je definovaná jako ideální v případě, že je mezi pozorováními odlišných tříd co největší vzdálenost. Tato vzdálenost lze česky označit jako hraniční pásmo (margin). Přerušované přímky jsou dané tzv. podpurnými vektory (support vector), což jsou pozorování, které plně definují SVM model. Na obrázku je pouze nejjednodušší případ. Pokud by se nějaké pozorování ze třídy  $+$  nacházelo mezi pozorováními ze třídy  $-$  (outlier) tak, že by už nebylo možné rozdělit pozorování na dvě třídy přímkou bez chybné klasifikace, bylo by nutné použít hraniční pásmo s možností chyb klasifikace (soft margin). Po použití soft margin je možná určitá míra chybné klasifikace, aby bylo vůbec možné matematicky SVM aplikovat, případně aby došlo ke zlepšení generalizace modelu. [32]

Matematicky lze vyjádřit trénovací data jako matici, kde počet řádků  $n$  odpovídá počtu pozorování a počet sloupců  $d$  odpovídá počtu příznaků jednotlivých pozorování. Pro jedno pozorování se jedná o vektor příznaků  $\mathbf{x}^{(1)}$  s délkou  $d$  a příslušící třídu  $y^{(1)}$ , která nabývá hodnot  $+1, -1$ . Separující přímka, na obrázku plnou čarou, je popsána vztahem 3.3 níže,

$$h = \mathbf{w}^T \mathbf{x} + b = 0 \quad (3.3)$$

kde pro obecně rozměrný prostor je  $\mathbf{w}$  vektor s parametry nadroviny a  $b$  je posunutí (bias). Pro 2D případ z obrázku 3.2 jde o dvě váhy  $w_1, w_2$  a posunutí  $b$ . Vektor  $\mathbf{w}$  je normálovým vektorem nadroviny. Nadroviny označené přerušovanou čarou jsou dané



**Obr. 3.2:** Grafická reprezentace SVM [33], přeloženo a upraveno

podpůrnými vektory. Pro pozorování ze tříd  $+$ ,  $-$  jsou splněny nerovnice 3.4. [32]

$$\begin{aligned} \text{třída } - : \quad w^T x + b &\leq -1 \\ \text{třída } + : \quad w^T x + b &\geq 1 \end{aligned} \quad (3.4)$$

S využitím hodnoty pro  $i$ -tou třídu  $y^{(i)}$ , která nabývá pouze hodnot  $-1, +1$  lze podmínku z 3.4 přepsat do vztahu 3.5, tato podmínka

$$y^{(i)}(\mathbf{w}^T \mathbf{x} + b) \geq 1 \quad (3.5)$$

musí platit pro všechny pozorování mimo hraniční pásmo. Pro co nejširší hraniční pásmo je třeba minimalizovat velikost vektoru  $\mathbf{w}$ , tedy  $\|\mathbf{w}\|$ , což se dá pro optimalizování výpočtu (derivací) zapsat jako vztah 3.6. [32]

$$\begin{aligned} \min_{\mathbf{w}, b} \quad & \frac{1}{2} \cdot \|\mathbf{w}\|^2 \equiv \left( \frac{1}{2} \cdot \mathbf{w}^T \mathbf{w} \right) \\ \text{za podmínky} \quad & y^{(i)}(\mathbf{w}^T \mathbf{x} + b) \geq 1 \\ & i = 1, \dots, n \end{aligned} \quad (3.6)$$

Jedná se tedy o primární problém kvadratického programování, minimalizace vektoru  $\mathbf{w}$  a posunutí  $b$  za omezujících podmínek popsanych v 3.6. V případě soft margin SVM je přidána proměnná  $\xi^{(i)}$ , která pro každé pozorování udává míru, na kterou lze porušit hraniční pásmo. Proměnnou  $C$  poté nastavujeme, do jaké míry je povolena chybná klasifikace při řešení optimalizačního problému. Pro soft margin SVM lze tedy přepsat rovnici pro minimalizování do tvaru uvedeného v 3.7. [32]

$$\begin{aligned} \min_{\mathbf{w}, b, \xi} \quad & \frac{1}{2} \cdot \mathbf{w}^T \mathbf{w} + C \sum_{i=1}^m \xi^{(i)} \\ \text{za podmínek} \quad & y^{(i)}(\mathbf{w}^T \mathbf{x} + b) \geq 1 - \xi^{(i)} \\ & \xi^{(i)} \geq 0 \text{ pro } i = 1, \dots, n \end{aligned} \quad (3.7)$$

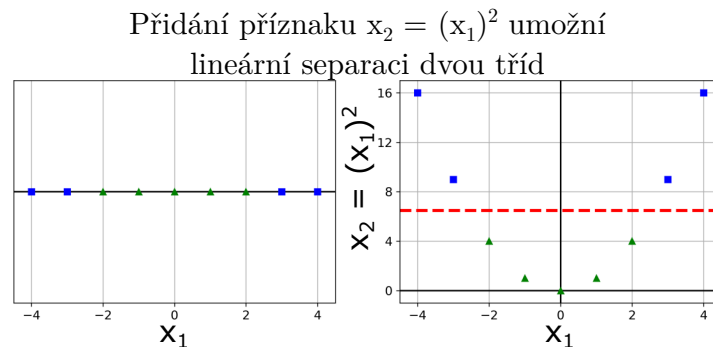
Primární úloha lze v případě SVM optimalizace řešit pomocí duální úlohy, díky tomu je možné použít tzv. trik s jádrovou funkcí (kernel trick). Využitím triku s jádrovou funkcí lze obejít výpočetně náročnou transformaci příznaků do vyšší dimenze, ve které už můžou být třídy lineárně separovatelné. Duální úloha se řeší metodou Lagrangeových multiplikátorů. [32]

V případě, že chceme provést transformaci vektoru  $\mathbf{x}$  dvou příznaků polynomem druhého řádu, můžeme použít mapovací funkci  $\phi(\mathbf{x})$ , která je definována v první části vztahu 3.8. [32]

$$\phi(\mathbf{x}) = \phi \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} x_1^2 \\ \sqrt{2}x_1x_2 \\ x_2^2 \end{pmatrix} \quad (3.8)$$

$$\phi(\mathbf{a})^T \phi(\mathbf{b}) = \begin{pmatrix} a_1^2 \\ \sqrt{2}a_1a_2 \\ a_2^2 \end{pmatrix}^T \cdot \begin{pmatrix} b_1^2 \\ \sqrt{2}b_1b_2 \\ b_2^2 \end{pmatrix} = \dots = (\mathbf{a}^T \mathbf{b})^2$$

Pokud poté tuto definovanou mapovací funkci použijeme na dvourozměrné vektory  $\mathbf{a}$ ,  $\mathbf{b}$  a s transformovanými vektory provedeme skalární součin, dostaneme výstup z druhé části vztahu 3.8. [32]



**Obr. 3.3:** Přidání druhého příznaku umožní separaci [32], upraveno

Z tohoto vyplývá, že aplikaci mapování a poté výpočtu skalárního součinu dvou vektorů odpovídá pouhé umocnění skalárního součinu na druhou. Funkce, která tuto operaci zastává, se nazývá polynomiální jádrová funkce druhého řádu, je definovaná jako  $K(\mathbf{a}, \mathbf{b}) = (\mathbf{a}^T \mathbf{b})^2$ . Není tedy nutné počítat transformaci  $\phi(\mathbf{x})$ . [32]

Tohoto triku se dá využít při řešení duální úlohy. V duální úloze se vyskytuje výraz skalárního součinu, viz vztah 3.9.

$$\mathbf{x}^{(i)T} \mathbf{x}^{(j)}; \text{ pol. transformace: } \phi(\mathbf{x}^{(i)T})\phi(\mathbf{x}^{(j)}) \equiv (\mathbf{x}^{(i)T} \mathbf{x}^{(j)})^2 \quad (3.9)$$

V případě, že chceme využít polynomiální transformace vektorů příznaků, lze tuto operaci nahradit aplikováním jádrové funkce. [32]

Řešením duální úlohy řešení pomocí Lagrangeových multiplikátorů je uvedeno ve vztahu 3.10.

$$\mathbf{w} = \sum_{i=1}^m \alpha^{(i)} y^{(i)} \mathbf{x}^{(i)} \quad (3.10)$$

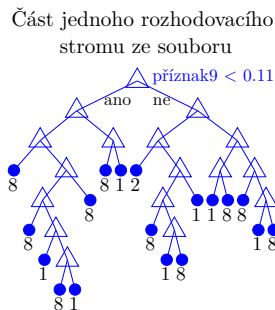
kde  $\alpha^{(i)}$  jsou multiplikátory, která nabývají nenulovou hodnotu pouze pro podpůrné vektory. Když dosadíme tento výsledek do vztahu 3.3 s využitím transformace  $\phi(\mathbf{x})$  pro predikci nové instance  $\mathbf{x}^{(n)}$ , vyjde nám vztah, do kterého lze dosadit jádrovou funkci za  $\phi(\mathbf{x}^{(i)})^T \phi(\mathbf{x}^{(i)})$ . [32]

### Klasifikace tří a více tříd

V případě klasifikace závady na ložiscích je vhodné rozlišovat více stavů ložiska, než jenom *poškozené* a *zdravé*, čili více než dvě třídy. V MATLABu je tento problém klasifikace více tříd převeden na binární klasifikaci více modelů podle kódovacího a dekódovacího schématu *one-vs-one*. Pro každý binární SVM model je jedna třída označena jako pozitivní, jedna jako negativní a ostatní třídy jsou ignorovány. První model tedy trénuje na pozorováních ze třídy 1 a třídy 2, kdy třída 1 je označena jako +, třída 2 jako -. Nové pozorování je přiřazeno té třídě, pro kterou je minimální souhrnná velikost ztrátové funkce přes všechny SVM modely. Pro kódování *one-vs-one* je počet modelů určen jako  $K(K - 1)/2$ , kde  $K$  je počet tříd. [34]

### 3.1.3 RF (Random Forest)

Algoritmus Random Forest je zástupcem ensemble učení, tedy model, který klasifikuje s použitím většího množství submodelů. Každý ze submodelů provede predikci na základě nového pozorování, výsledná predikce souboru modelů je dána hlasováním submodelů. V případě hard voting klasifikace se jedná o výběr nejčastěji predikované třídy napříč submodely, v případě soft voting se jedná o výběr třídy s největší průměrnou predikovanou pravděpodobností napříč submodely. [32]



Obr. 3.4: Ukázka části jednoho rozhodovacího stromu ze souboru

U algoritmu Random Forest jsou submodely v souboru rozhodovací stromy. Rozhodovací stromy jsou uživatelsky jednoduše interpretovatelné, viz obrázek 3.4. Trojúhelníky na obrázku jsou interpretací uzlu stromu. V uzlu je zjištěno splnění podmínky, například  $Kurtosis < 3$ . Podle výsledku podmínky se v uzlu strom buďto dále větví, nebo končí listem, který je v obrázku 3.4 reprezentován číslem. Toto číslo je označením třídy, která náleží pozorování splňující podmínky v rozhodovacích uzlech vedoucích k tomuto výslednému listu. [32]

Jeden rozhodovací strom může být obecně budován rozdílnými algoritmy, příkladem je CART algoritmus. Algoritmus CART vytvoří uzel stromu tak, že vhodným výběrem příznaku v uzlu a prahové hodnoty tohoto příznaku rozdělí trénovací množinu na dvě podmnožiny, levou a pravou. Levá podmnožina obsahuje pozorování splňující prahovou hodnotu příznaku uzlu, pravá nespĺňuje. Vhodná volba příznaku a prahové hodnoty se zjistí minimalizací chybové funkce. Chybová funkce je váženým průměrem hodnot Gini impurity získaných ze vztahu 3.12 pro levou a pravou podmnožinu. [32]

Výpočet chybové funkce je uveden ve vztahu 3.11 níže.

$$J(\text{příznak,práh}) = \frac{m_{\text{levá}} \cdot G_{\text{levá}} + m_{\text{pravá}} \cdot G_{\text{pravá}}}{m_{\text{celk}}} \quad (3.11)$$

Pro vztah 3.11 je  $J(\text{příznak,práh})$  hodnota chybové funkce,  $m_{\text{pravá/levá}}$  je počet instancí v pravé/levé podmnožině pro daný uzel,  $m_{\text{celk}}$  je celkový počet pozorování pro daný uzel. Takto probíhá větvení stromu, dokud nedojde k dosažení ukončovací podmínky, například maximální hloubky stromu. [32]

Tento algoritmus může být s úpravami, které budou popsány níže použit k vybudování jednoho stromu v algoritmu Random Forest.

$$G_i = 1 - \sum_{k=1}^n p_{i,k}^2 \quad (3.12)$$

Pro vztah 3.12 je  $G_i$  hodnota Gini impurity,  $n$  je počet tříd pro uzel,  $p_{i,k}$  je poměr pro daný uzel, kdy jmenovatelem v poměru je součet počtu pozorování všech tříd daného uzlu, čitatelem je počet pozorování  $i$ -té třídy náležící danému uzlu.  $G_i = 0$  odpovídá takovému uzlu, který má všechny trénovací pozorování od stejné třídy (vždy tedy jde o list stromu).

Problémem stromů je, že pokud nedojde k regularizaci – například k omezení hloubky, minimálního počtu instancí na list či prořezávání, tak se stromy kompletně naučí na trénovací data a pravděpodobně dojde k přeučení. [32]

Řešení nabízí algoritmus Random Forest, který využívá soubor rozhodovacích stromů. Každý ze stromů není učen na všech  $m$  trénovacích pozorováních, nýbrž je z  $m$  trénovacích pozorování vybrána náhodně podmnožina o stejné velikosti  $m$ ,

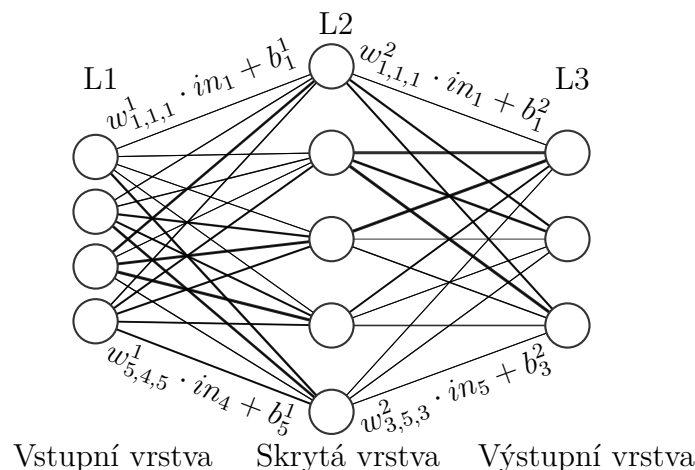


a to s možným opakováním pozorování. Volba podmnožin s opakováním pozorování z trénovací množiny se nazývá bootstrap aggregating (bagging). [32]

Pro zvýšení diversity jednotlivých stromů v souboru je navíc při tvorbě rozvětvovacího uzlu vybírán optimální příznak ne ze všech příznaků trénovací množiny, ale z malé, náhodně vybrané podmnožiny těchto příznaků. Použitím algoritmu Random Forest dojde ke zvýšení přesnosti predikce převážně pro klasifikační problémy. [35]

### 3.1.4 NN (Neuronová síť)

Neuronové sítě mohou být použity na klasifikaci i regresi. Skládají se minimálně ze tří vrstev – vstupní vrstva, alespoň jedna skrytá vrstva a výstupní vrstva, viz obrázek 3.5. Jedná se o model, který je inspirován základní výpočetní jednotkou v mozku – neuronem. Neurony jsou mezi sebou v mozku propojeny velkým množstvím axonů, stejně tak mezi umělými neurony jednotlivých vrstev neuronové sítě existují propojení. [32]



**Obr. 3.5:** Neuronová síť s jednou skrytou vrstvou

Vstupní vrstva obsahuje stejný počet neuronů jako počet příznaků dat, výstupní vrstva obsahuje počet neuronů rovný počtu výstupů – například počtu klasifikovaných tříd. Neuronová síť může obsahovat libovolný počet skrytých vrstev. Jednotlivé skryté vrstvy mohou obsahovat libovolný počet neuronů. Vhodné kombinace počtu skrytých vrstev a neuronů v nich jsou otázkou problému, který se s neuronovou sítí snažíme vyřešit a je nutné kombinace otestovat.

Každý jeden neuron je propojen s každým neuronem v předchozí a následující vrstvě (s výjimkou vstupních a výstupních vrstev). Spojení mezi neurony ve dvou sousedních vrstvách je realizováno příslušící vahou  $w_{i,n-1,n}^L$ . Každý neuron má také

příslušící posunutí (bias)  $b_n^L$ . Váha a posunutí jsou parametry, které je nutné optimalizovat učením. Vstup do neuronu z předchozí vrstvy je škálován vahou jako  $n(x) = x \cdot w + b$ , tato hodnota je poté použita jako vstup do aktivační funkce  $f(n(x))$ , jejíž výstup je výstupem neuronu. Mezi aktivační funkce patří například ReLU nebo hyperbolický tangens. Použitá aktivační funkce může mít i negativní vliv na kvalitu vytrénované sítě. [32]

### Zpětné šíření chyby (backpropagation)

Jako metrika hodnocení přesnosti predikce se používá chybová funkce  $L$ . Nižší hodnota obvykle znamená lepší model. Chybovou funkcí může být například průměr z kvadrátu odchylek (MSE) nebo křížová entropie (Cross Entropy).

$$L_{MSE} = \frac{1}{m} \cdot \sum_{i=1}^m (\text{pred}(\mathbf{x}^i) - y^i)^2 \quad (3.13)$$

Pro 3.13 je  $L_{MSE}$  chybová funkce,  $m$  je počet pozorování,  $\text{pred}(\mathbf{x}^i)$  je predikovaný výstup modelu z  $i$ -tého vektoru příznaků  $\mathbf{x}^i$ ,  $y^i$  je cílový výstup modelu pro  $i$ -té pozorování.

$$L_{C-E} = -\frac{1}{m} \cdot \sum_{i=1}^m \sum_{k=1}^K y_k^i \cdot \log(p_k^i) \quad (3.14)$$

Pro 3.14 je  $L_{C-E}$  chybová funkce,  $m$  je počet pozorování,  $K$  je počet tříd,  $y_k^i$  je cílová pravděpodobnost náležitosti daného pozorování k dané třídě (většinou 1 – náleží, 0 – nenáleží),  $p_k^i$  je predikovaná pravděpodobnost, že  $i$ -té pozorování náleží ke  $k$ -té třídě.

Cílem trénování je tedy minimalizovat hodnotu chybové funkce. K minimalizování chybové funkce je třeba vhodně upravit hodnoty vah a posunutí. K nalezení takových hodnot vah a posunutí, při kterém dosáhne chybová funkce minima, je použit algoritmus zpětného šíření chyby v kombinaci s algoritmem gradientního sestupu (gradient descent). [32]

Metoda zpětného šíření chyby byla představena v [36]. Pro každý parametr sítě, v pořadí od výstupu neuronové sítě směrem ke vstupu (proto zpětné šíření chyby) je vypočtena parciální derivace chybové funkce v závislosti na optimalizovaném parametru ( $w/b$ ). Toho je možné docílit aplikací řetězového pravidla (chain rule) pro derivaci složené funkce. Po tomto kroku se aplikuje algoritmus gradientního sestupu. [36]

### Gradientní sestup (gradient descent)

Gradientní sestup je iterativní výpočet k nalezení minima funkce. Postup metody je následující. V prvním kroku jsou váhy mezi neurony inicializovány na počáteční

hodnotu, váhy náhodně, posunutí většinou na nulu. Poté je vypočten výstup neuronové sítě pro trénovací data. Hodnota váhy v novém kroku iterace je dána vztahem 3.15. Vztah 3.15 využívá vztahů pro parciální derivace chybové funkce vypočtených pomocí zpětného šíření chyby. Při algoritmu gradientního sestupu dochází k posunu směrem k záporné hodnotě parciální derivace chybové funkce podle dané váhy. Parametr  $\epsilon$  je krok učení. [32]

$$w_i^{new} = w_i^{old} - \epsilon \cdot \nabla L(w_i) = w_i^{old} - \epsilon \cdot \frac{\partial L(w_i)}{\partial w_i} \quad (3.15)$$

Takto jsou vypočteny nové hodnoty všech vah a posunutí pro danou iteraci. Při reálné aplikaci jsou váhy vypočteny maticově. Ve výpočtu nových hodnot parametrů neuronové sítě se pokračuje, dokud není splněna nějaká z ukončovacích podmínek. Tou může být dosažení konečné nízké hodnoty gradientu, dosažení konečného počtu iterací přes trénovací data nebo zvýšení chyby na validační množině.

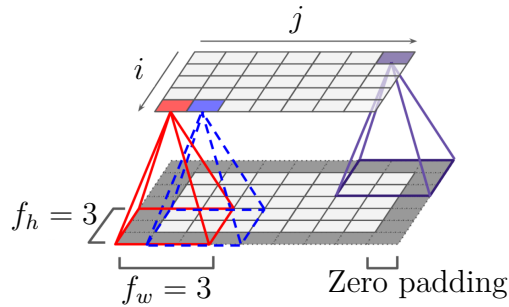
Klasický algoritmus gradientního sestupu (GD) je pomalý, jelikož se pro každou iteraci počítá přes celou trénovací množinu. Může být použit stochastický gradientní sestup (SGD), případně gradientní sestup s iteracemi přes malé množství pozorování (mini-batch) (MBGD). SGD vybere pro každou iteraci z trénovací množiny náhodné pozorování a upraví parametry sítě podle něj. To je výhodné z hlediska menšího počtu výpočtů, a také díky tomu lze lépe najít globální optimum parametrů. Vzhledem k náhodné povaze ale nemusí dojít k ustálení metody, používá se tedy MBGD, který provádí iterace přes malý počet (např. 32) trénovacích pozorování. Díky tomu lze využít rychlé maticové výpočty. [32]

## 3.2 Konvoluční neuronové sítě

V případě, že bychom potřebovali klasifikovat 2D data (obrázky) například použitím klasické neuronové sítě, bylo by potřeba nejdříve konvertovat 2D signál na 1D signál a jednotlivé pixely poté použít jako vstupní příznaky. Pro obrázek 64x64 by se jednalo o 4096 vstupních příznaků, v případě dvou skrytých vrstev o 300 a 500 neuronech a 10 klasifikačních tříd se tedy jedná o  $(4096 \cdot 300 + 300) + (300 \cdot 500 + 500) + (500 \cdot 10 + 10) = 1384610$  parametrů, které je nutné při každé trénovací iteraci optimalizovat. Výpočetní náročnost těchto operací řeší konvoluční neuronová síť (CNN), která má před klasifikační neuronovou sítí konvoluční část, která má za úkol extrakci příznaků ze vstupních obrázků. [32]

### 3.2.1 Konvoluční vrstva

V konvolučních vrstvách probíhá 2D konvoluce vah jádra filtru s obrázkem, k výsledné mapě je poté přičtena hodnota posunutí, viz obrázek 3.6.



**Obr. 3.6:** 2D konvoluce [32], upraveno

Konvoluce probíhá pro různé velikosti jádra filtru, na obrázku 3.6 velikost  $3 \times 3$ , které se pohybují s určitým krokem, na obrázku 3.6 jde o krok 1. Výstupem 2D konvoluce jsou tzv. mapy, jejich velikost a počet je závislý na počtu konvolučních filtrů, velikosti filtrů, kroku postupu přes obrázek. Také závisí na tom, jestli vyžadujeme u obrázku výplň (padding), viz obrázek 3.6. Výplň zvětší rozměry vstupu (obrázku), díky čemuž je poté výsledná mapa větší. Výstupní mapy z nižších konvolučních vrstev extrahují příznaky nižší úrovně, kterými jsou například hrany. Tyto mapy jsou vstupem do následujících konvolučních vrstev, které z příznaků v mapách předchozí úrovně konstruuji příznaky vyšší úrovně. [32]

Výstupní mapy z konvoluční vrstvy mohou být zmenšeny použitím slučující vrstvy (pooling layer), která přes plochu slučující vrstvy  $p_h \times p_w$  vybere buďto největší hodnotu z mapy (tzv. Max Pool) nebo průměrnou hodnotu z mapy (tzv. Average Pool). Slučující vrstvy zmenšují výpočetní náročnost a také zvyšují invariančnost sítě na drobné posuvy objektu na obrázku, jelikož operují přes plochu  $p_h \times p_w$ . [32]

Bloky konvolučních vrstev a slučujících vrstev jsou v závislosti na architektuře CNN různými způsoby zapojeny za sebou. [32]

Váhy a posunutí jsou v případě konvolučních sítí optimalizovány ve standardní implementaci s využitím gradientního sestupu, viz 3.1.4 a zpětného šíření chyby. V případě, že se trénuje síť s větším počtem vrstev, kterou často CNN je, může docházet k problému s mizejícím gradientem (vanishing gradient). Při zpětné propagaci chyby od výstupních ke vstupním vrstvám dochází k postupnému zmenšování gradientu, pro vrstvy u vstupu tak už dojde k minimální úpravě vah a model nekonverguje k dobrému řešení. Tento problém je způsoben mimo jiné použitím aktivačních funkcí, které saturují (například hyperbolický tangens saturuje v  $-1$  a  $1$ ), derivace tedy při velkých vstupních hodnotách jde k nule. Při algoritmu zpětné propagace chyby tedy dochází k aktualizaci vah velmi malými hodnotami. [32]

Řešením je použití nesaturované aktivační funkce, například ReLU a její varianty,

případně speciální metody inicializace vah sítě. Dalším řešením problému s gradientem je zavedení speciální vrstvy přímo před aktivační funkce s názvem normalizace dávky (batch normalization). [32]

### 3.2.2 Vrstva normalizace dávky (batch normalization layer)

Vrstva normalizace dávky upravuje trénovací dávku. Spočívá v tom, že pro danou dávku je pro každý vstup vypočítán průměr a směrodatná odchylka, vstupy jsou následně pomocí těchto parametrů posunuty a normalizovány. Jako učící parametry této vrstvy jsou škálovací koeficienty a posunutí pro každý ze vstupů. Ty jsou optimalizovány při procesu učení standardně použitím algoritmu zpětného šíření chyby. Výstupy z vrstvy normalizace dávky jsou vynásobeny škálou a posunuty o hodnotu posunutí. Proces testování/predikce nemusí nutně probíhat v dávkách, parametry průměru a směrodatné odchylky, které jsou při režimu trénování vypočítány z dávky jsou nahrazeny exponenciálním plovoucím průměrem přes všechny pozorování použité při učení. [32]

Vrstva normalizace dávky přináší vylepšení přesnosti predikce a také omezuje problém s mizícím gradientem. Díky tomu lze s modelem vylepšeným o vrstvy normalizace dávky dosáhnout lepší přesnosti při menším počtu trénovacích iterací. [32]

### 3.2.3 Vrstva vypuštění neuronů (dropout layer)

Vrstva vypuštění neuronu nemá žádný parametr optimalizovaný při procesu učení. Jejím účelem je zabránit přetrénování sítě v případě sítí, které mají velké množství parametrů a mohou být díky tomu k přetrénování náchylné. Hlavním parametrem vrstvy je parametr  $p$  – pravděpodobnost vypuštění. Ten specifikuje, s jakou pravděpodobností bude **při trénování** neuron v dané vrstvě a dané iteraci ignorován. Přetrénování je omezeno díky tomu, že při každé iteraci je síť mírně odlišná. Po dokončení trénování už tato vrstva není aktivní a všechny neurony jsou funkční. V případě, že model vykazuje známky přetrénování je možné zvýšit hodnotu parametru  $p$ . Použití této vrstvy zvýší přesnost predikce o 1–2 %. [32]

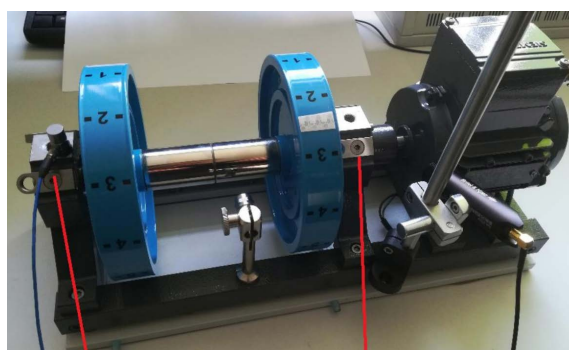
## 4 Ověření metod na vlastních naměřených datech

V následující kapitole je popsán postup měření, měřící přípravek a MATLAB aplikace pro inspekci naměřených dat a vykreslení časových trendů pro metody popsané v kapitole 2. Jedná se o metody, které sice dokáží indikovat poškození, případně i poškozenou komponentu, ale jejich výsledky je nutné ručně interpretovat a podle zkušeností rozhodnout míru poškození a komponentu poškození. Funkčnost těchto metod je v této kapitole ověřena na vlastních datech periodicky získávaných z poškozených ložisek.

### 4.1 Popis měření ložisek

#### 4.1.1 Přípravek na měření vibrací

Vibrace ložiska byly měřeny na přípravku fy Schenck na obrázku 4.1(a) níže. Tento přípravek má dvě ložiska na hřídeli, vibrace byly měřeny v radiálním směru na ložisku L1. Jako pohon hřídele slouží motor poháněný frekvenčním měničem, maximální otáčky motoru činí 1350 RPM.



L1 L2  
(a) Přípravek fy Schenck [37]



(b) Klec z PA66 [38]

Obr. 4.1: Přípravek fy Schenck a klec z PA66

Jako ložisko bylo zvoleno jednořadé kuličkové ložisko SKF 6202 TN9/C3, které má rozměry odpovídající uložení ložiska na přípravku fy Schenck na obrázku 4.1. Toto ložisko má klec z polymeru PA66. Tento typ klece je na kuličkách nacvaknutý – nejedná se o standardní kovovou klec snýťovanou dohromady, je tedy možné ji demontovat. Odstranění klece umožňuje lepší přístup k vnější a vnitřní dráze

ložiska, jelikož je možné kuličky seskupit k sobě. Díky tomu pak lze lépe uměle poškodit dráhy kroužků.

### 4.1.2 Sběr dat

Data byla měřena měřícím systémem od firmy Brüel & Kjær. K sběru dat byla použita vstupně výstupní karta LAN-XI 3160, která má 4 vstupy a 2 výstupy generátoru. Jako obslužný software ke sběru dat byl použit BK Connect 2021. Tento software při měření poskytuje i základní údaje o měřeném signálu, pro důkladnější analýzu dat ale není příliš vhodný, jelikož neposkytuje dostatečnou volnost, kterou disponuje například software MATLAB.



**Obr. 4.2:** Software BK Connect a modul LAN-XI 3160

Jako akcelerometr byl použit PCB PIEZOTRONICS 352C03 připevněný k přípravku z obrázku 4.1(a) pomocí šroubu. Sběr vibračních dat z ložiska byl ovládán periodickým triggerem z výstupu generátoru LAN-XI modulu. Každých 10 minut bylo spuštěno na 10 sekund měření. Změřená data jsou ukládána v binárním .bkc souboru, v BK Connect je možné vyexportovat data do .csv pro zpracování v ostatních nástrojích.

### 4.1.3 Poškození ložisek

První ložisko SKF 6202 TN9/C3 bylo poškozeno opakovaným deformováním a ponecháním na vzduchu, čímž došlo k částečné korozi všech komponent tohoto ložiska. Ložisko bylo zbaveno maziva a upevněno do přípravku fy Schenck z obrázku 4.1(a). Toto ložisko bylo provozováno celkem 2350 minut ( $\approx 39$  hodin) při 1350 RPM, aby bylo možné sledovat případnou další degradaci s časem.

Druhé ložisko SKF 6202 TN9/C3 bylo poškozeno diamantovým brusným tělískem – došlo k poškrábání vnější dráhy, viz červeně označená oblast na obrázku 4.3. Pro rychlejší rozvoj poškození bylo také zbaveno maziva. Druhé ložisko bylo v pří-



**Obr. 4.3:** Poškození vnější dráhy druhého ložiska

pravku fy Schenck z obrázku 4.1(a) provozováno celkem 560 minut ( $\approx 9,5$  hodin) při 1350 RPM, aby bylo možné sledovat případnou další degradaci s časem.

## 4.2 Zpracování naměřených dat

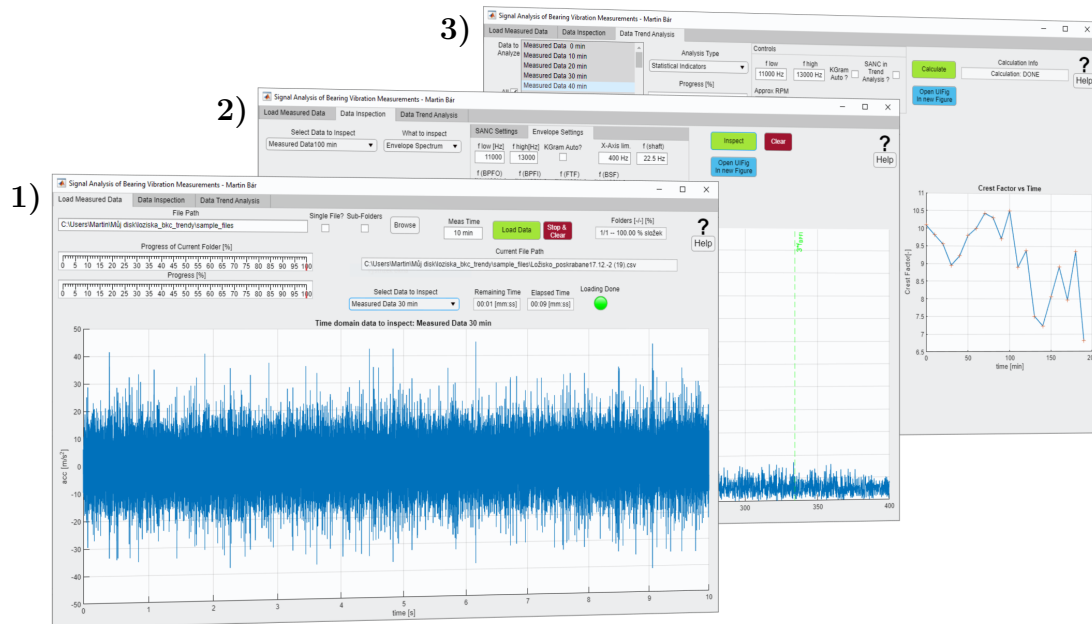
Software BK Connect umožňuje online sledování základních parametrů měřeného signálu jako je FFT spektrum, úroveň vibrací, časově-frekvenční zobrazení atd. Aby bylo možné použít metody z této práce popsané v kapitole 2, je nutné naměřená data exportovat z BK Connect a zpracovat je jiným nástrojem. Data v této práci jsou zpracována v programu MATLAB 2021b. Nástrojem *App Designer* byla v MATLABU vytvořena aplikace, která slouží k načtení, inspekci dat a k následnému vykreslení trendových grafů. Aplikace v MATLAB lze nástrojem *Application Compiler* připravit i pro použití na počítačích bez nainstalovaného MATLABu.

### 4.2.1 MATLAB aplikace

Aplikace na inspekci ložiskových vibrací je rozdělena na tři části, každé jedné části náleží jedna záložka, viz obrázek 4.4 níže.

V první části je nutné načíst .csv soubory s měřenými daty. Aplikace počítá s formátem hlavičky a sloupců, který vyexportuje BK Connect. Data vibrací z ložiska jsou očekávána ve druhém sloupci .csv souboru (sloupec 1 je časový vektor). Aplikace umožňuje načtení dávky souborů v podsložkách – takto mohou být uspořádána data například z jednotlivých dnů měření. Další možností je načtení celé jedné složky nebo i načtení jediného souboru. Aplikace indikuje zbývající čas do dokončení načítání. V případě chyby dojde k zobrazení výjimky, případně k výzvě k uživatelské změně nastavení načítání. Po načtení všech souborů je možné v této záložce provést rychlou kontrolu časového průběhu naměřených vibrací.





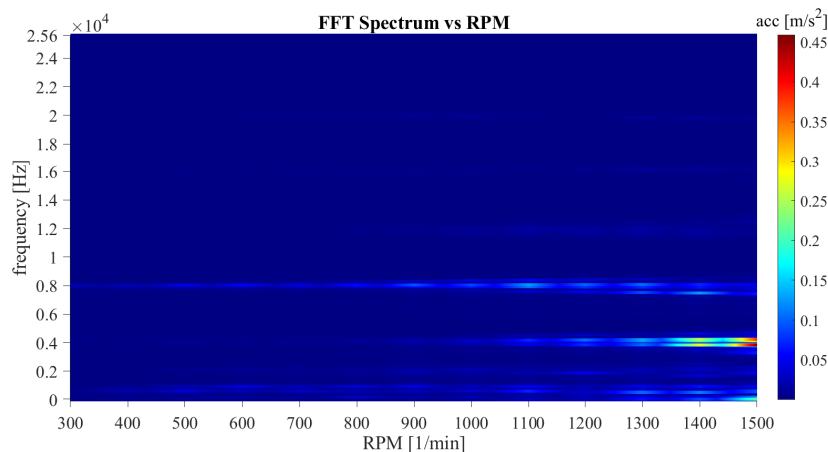
**Obř. 4.4:** Prostředí aplikace na inspekci změřených dat

Následující záložka slouží k podrobnější inspekci naměřených dat. Pro jednotlivé měření je zde možné zobrazit časový průběh vibrací, amplitudové spektrum, kurtoqram, časový průběh obálky signálu, amplitudové spektrum obálky signálu a také procentuální podíl RMS signálu pro různé úrovně FWT vs. referenční signál z kapitoly 2.3.3. Tato záložka je primárně určena pro rychlou inspekci různých parametrů naměřeného signálu a pro zjištění optimálního nastavení metod pro vykreslování časových trendů na následující záložce.

Poslední záložka slouží k vykreslení časových trendů vybraných metod. Jde o vývoj spektra obálky signálu, RMS obálky signálu, součet průměrného výkonu prvních třech harmonických frekvencí poruch ložisek, statistické indikátory a vývoj amplitudového spektra v čase. V této záložce je možné vybrat k vykreslení trendů všechna data nebo jenom výběr určitých měření. Protože MATLAB v *App Designeru* používá odlišný (a značně pomalejší) způsob vykreslování grafů než standardně, nejsou 3D grafy vykresleny v samotném okně programu, ale každý zvlášť v samostatných oknech.

## 4.3 Metody na vlastních datech

Pro určení frekvencí, na kterých dochází u přípravku fy Schenck z obrázku 4.1(a) k silným vibracím, bylo provedeno měření v rozsahu 300–1500 RPM. V grafu 4.5 níže je zobrazena závislost amplitudového spektra na otáčkách přípravku pro nepoškozené ložisko. Relativně nejsilnější vibrace jsou pro všechny rychlosti přípravku na



**Obr. 4.5:** Amplitudové spektrum v závislosti na otáčkách přípravku

frekvencích 4kHz, 8kHz a 12kHz. V grafech trendů je pro metody, které využívají obálkové analýzy, použito pásmo pro pásmovou propust v rozsahu 11–13 kHz a poté také pásmo automaticky vypočtené kurtogramem. V ideálním pásmu zvoleném kurtogramem je zkoumaný signál nejvíce impulsivní, mělo by se tedy jednat o pásmo, ve kterém jsou zesilovány impulsy od ložisek.

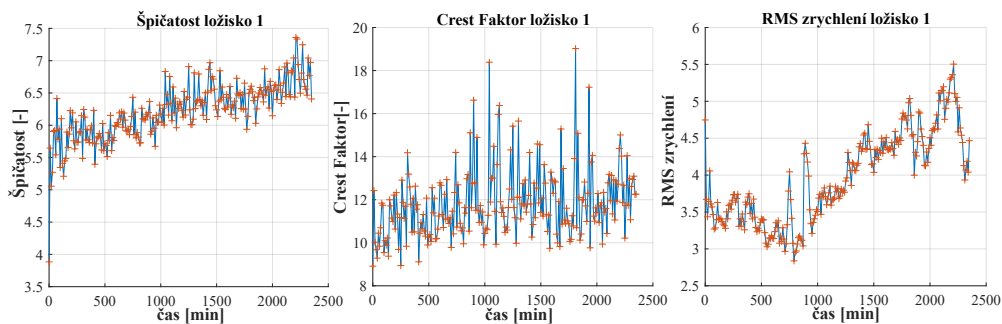
### 4.3.1 První poškozené ložisko

První ložisko poškozené deformací a korozí popsané v kapitole 4.1.3 bylo měřeno celkem 2350 minut. Pro toto ložisko jsou výsledky metod popsány v následující kapitole.

#### Výsledky statistických indikátorů

Statistické parametry signálu byly použity na neupravený signál přes jeho celý frekvenční rozsah. Výsledky jednotlivých metod jsou v grafu 4.6 níže.

Pro špičatost a efektivní hodnotu zrychlení je vidět očekávaný rostoucí trend s časem provozu, pro Crest Faktor je také vidět růst, ovšem ne tak silný, jako u předchozích metod.

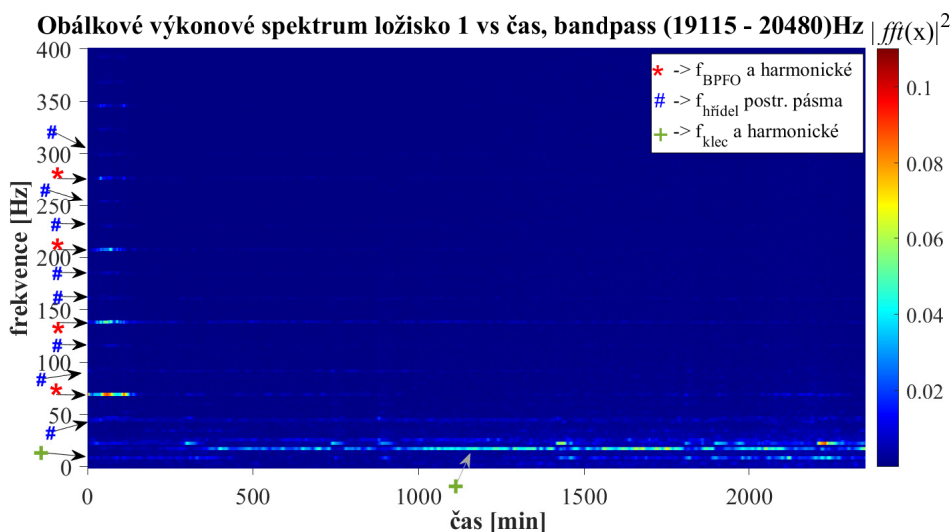


Obr. 4.6: Statistické metody pro ložisko 1

### Metoda obáلكové analýzy

Před provedením obáلكové analýzy bylo pomocí kurtogramu zjištěno frekvenční pásmo s největší impulzivitou dat pro pásmovou propust. Na všechna data z jednoho trendového grafu bylo aplikováno stejné pásmo, aby se dal objektivně posuzovat vývoj vibrací ve zvoleném pásmu.

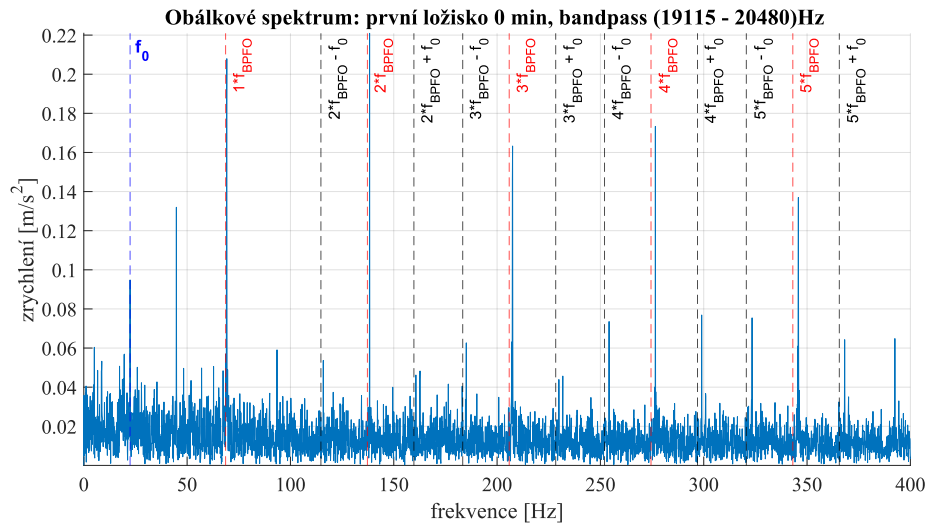
Vývoj výkonového spektra obáلكy je pro pásmo 19115–20480 Hz určené kurtogramem v grafu 4.7 níže. Pro toto frekvenční pásmo je nejlépe vidět vývoj vady.



Obr. 4.7: Obáلكové spektrum pro první ložisko

V grafu 4.7 jsou šipkami se symboly # , + , \* označeny frekvence ložiskových závad. Z grafu vyplývá, že na začátku měření je zde silná složka pro závadu na vnějším kroužku (označeno \*). U této složky jsou přítomny i první 4 harmonické, kolem kterých jsou postranní pásma s rozstupem rovným otáčkové frekvenci hřídele (označeno #). Od cca 1000 minut měření se v grafu objevuje i závada na kleci

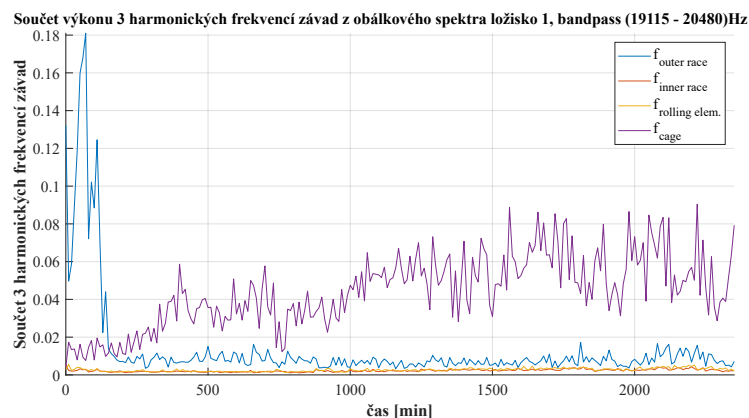
(označeno +) a její druhá harmonická. Pro přehlednost byly z tohoto grafu vyextrahovány první 3 harmonické pro 4 různé frekvence ložiskových závad (které jsou popsány v kapitole 1.3), tyto 3 harmonické byly pro každý typ závady sečteny a jsou zobrazeny v grafu 4.9.



Obr. 4.8: Obálkové spektrum pro první ložisko

Ukázka jednoho měření z grafu 4.7 je v grafu 4.8. Jedná se spektrum obálky pro začátek měření.

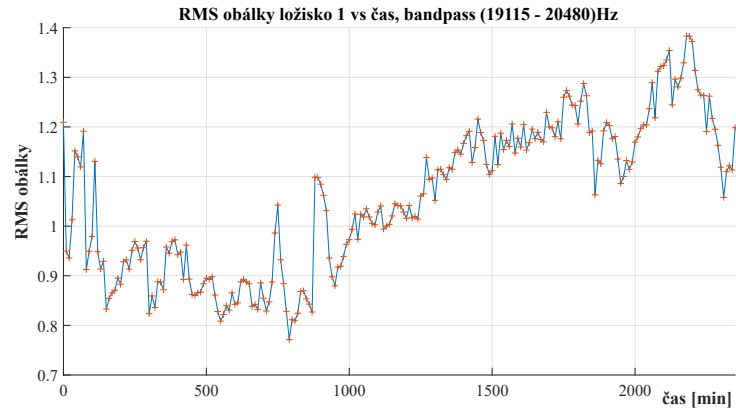
V grafu 4.8 je dobře rozlišitelných 5 harmonických závadové frekvence na vnější dráze ( $f_{BPFO}$ ). Také je vidět, že se kolem této závadové frekvence tvoří postranní pásy s rozestupem otáčkové frekvence hřídele ( $f_0$ ). Tyto postranní pásma jsou pravděpodobně způsobena nevývahou hřídele. Tento jev je popsán v kapitole 1.3.1.



Obr. 4.9: Vývoj frekvencí závad pro první ložisko

V grafu 4.9 je lépe viditelný vývoj závady na kleci a zároveň pokles závady na vnější dráze.

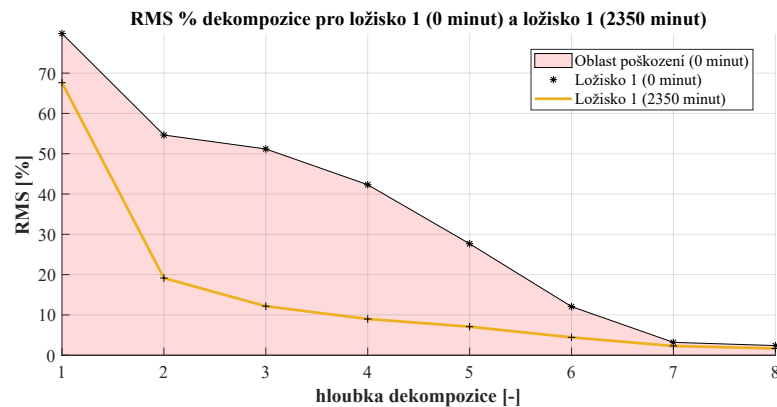
Dále byla vypočtena efektivní hodnota obálky, pro pásmo 19115–20480 Hz je v grafu 4.10 níže.



**Obr. 4.10:** Vývoj efektivní hodnoty obálky pro první ložisko

## FWT dekompozice a podíl na celkovém RMS

V kapitole 2.3.3 byla popsána metoda na určení stavu ložisek rozkladem měřeného signálu na jednotlivé aproximace a srovnání míry poklesu RMS těchto aproximací vůči RMS celkového signálu a referenčnímu měření. Pro první ložisko bylo jako referenční měření vybráno měření z 0 minut chodu a jako zkoumané bylo vybráno měření z 2350 minut. Z grafu 4.11 je vidět, že pro pozdější měření je výsledek v červené oblasti a klesá s hloubkou dekompozice rychleji než pro referenční měření v 0 minutě chodu. Z těchto výsledků lze usuzovat, že ložisko je v 2350 minutě chodu v horším stavu.



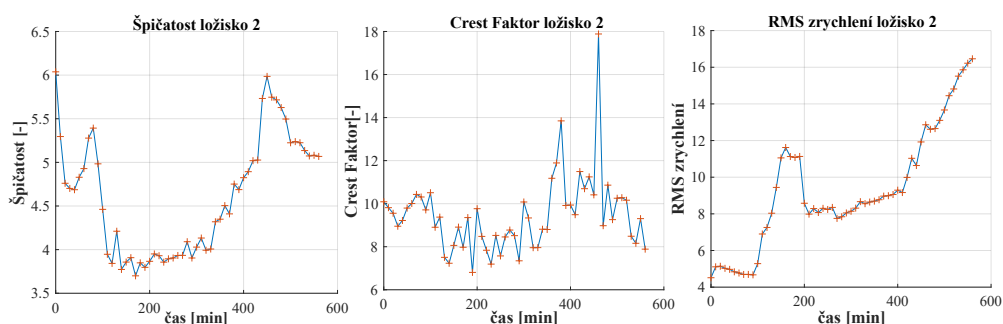
Obr. 4.11: Výsledek podílu RMS jednotlivých aproximací pro první ložisko

### 4.3.2 Druhé poškozené ložisko

Druhé ložisko s poškozenou drahou vnějšího kroužku popsané v kapitole 4.1.3 bylo měřeno celkem 560 minut. Pro toto ložisko jsou výsledky metod popsány v následující kapitole.

#### Výsledky statistických indikátorů

Statistické parametry signálu byly použity na neupravený signál přes jeho celý frekvenční rozsah. Výsledky jednotlivých metod jsou v grafu 4.12 níže.



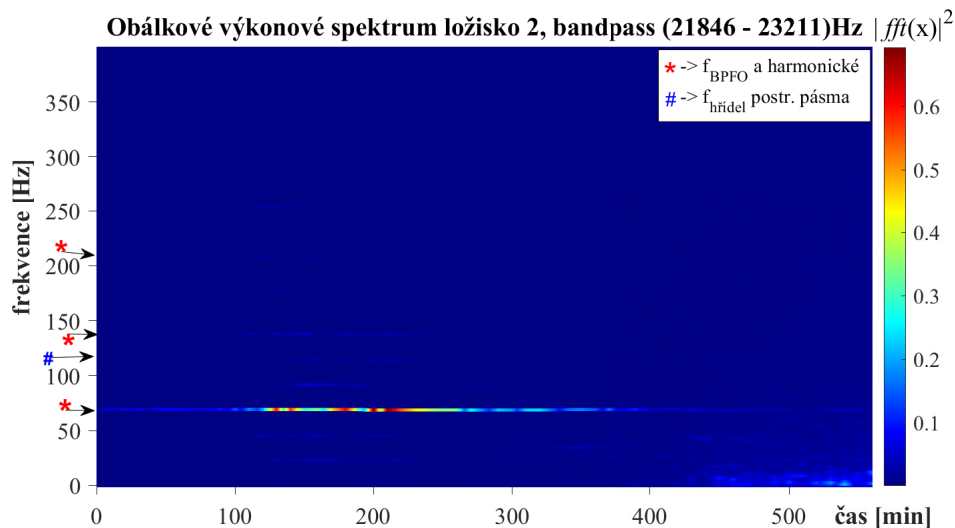
Obr. 4.12: Statistické metody pro ložisko 2

Efektivní hodnota zrychlení má pro ložisko 2 rostoucí trend, pro Crest faktor a špičatost by bylo nutné ještě pokračovat v měření, aby byl výsledek průkazný. U špičatosti i crest faktoru je nutné sledovat jejich vývoj po celou dobu života ložiska, protože i těsně před zničením může dojít k poklesu do hodnot, které lze interpretovat jako zdravé.

#### Metoda obálkové analýzy

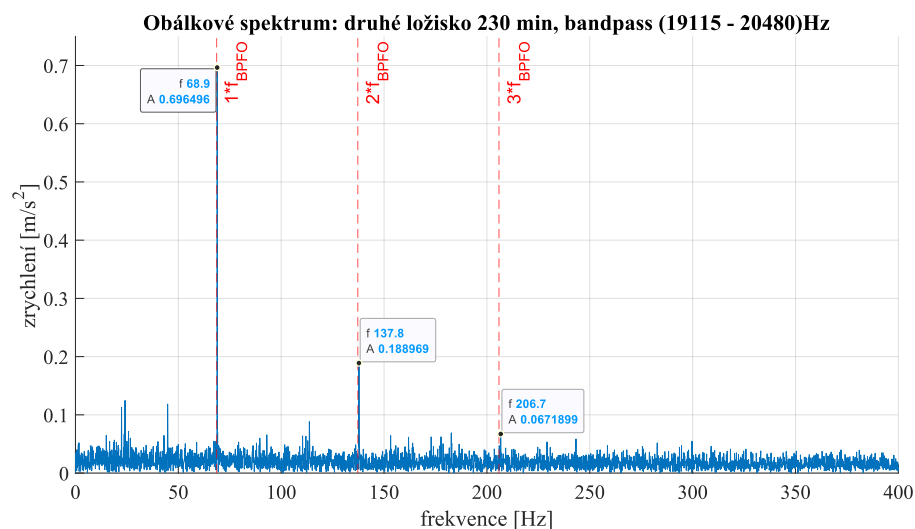
Pro druhé ložisko bylo frekvenční pásmo s největší impulzivitou v rozsahu 21,8–23,2 kHz, pro toto pásmo výsledky zobrazeny v grafu 4.13.

V grafu 4.13 lze vidět zpočátku zvyšování vibrací na frekvenci odpovídající vadě na vnějším kroužku. Zhruba u 400 minuty dochází k utlumení této diskrétní frekvenční čáry a harmonických a začíná vystupovat pásmo frekvencí do 50 Hz. Možným důvodem toho, že se po určitém čase přestane projevovat složka odpovídající frekvenci poruchy na vnějším kroužku je zahlazení poškození. Když je poškození čerstvé, má ostré hrany a při setkání dvou povrchů tak může ložisko více zvonit. Delší dobou chodu se může poškození zahladit, a tak se už ve spektru neprojeví tak ostrou složkou. Toto ovšem neznamená, že by se stav ložiska zlepšil. Z grafu 4.12 pro efektivní hodnotu zrychlení a z grafu 4.16, který zobrazuje efektivní hodnotu obálky je zřejmé, že se vibrace ložiska s postupem času zhoršují.



Obr. 4.13: Obálkové spektrum pro druhé ložisko

Ukázka jednoho měření z grafu 4.13 je v grafu 4.14. Jedná se spektrum obálky pro měření v 230 minutě chodu. Ve spektru jsou patrné první tři harmonické závadové frekvence na vnější dráze ( $f_{BPFO}$ ).



Obr. 4.14: Obálkové spektrum pro druhé ložisko

V grafu 4.15 je vidět vývoj amplitud pro frekvence ložiskových závad. Je vidět, že zhruba po 400 minutě vystupuje frekvence závady na kleci. Při dané otáčkové frekvenci ale první tři harmonické této závady sahají pouze do 30 Hz. V grafu 4.13 je patrné, že v tomto pásmu už se nevyskytují jednoduše rozlišitelné diskrétní frekvenní složky, takže se může jednat pouze o falešnou indikaci závady následkem

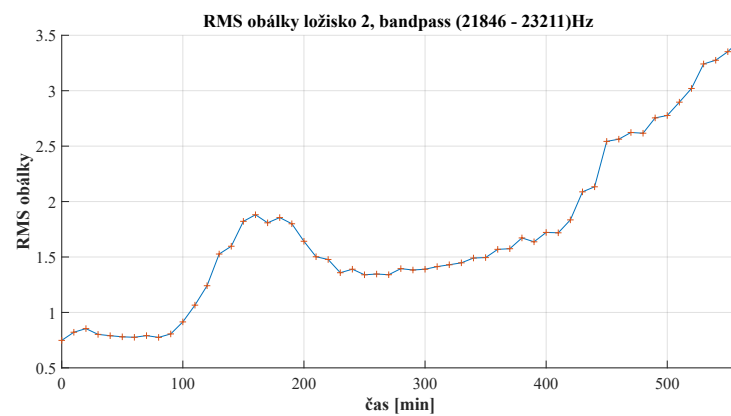


nárůstu výkonu v celém tomto frekvenčním pásmu.



Obr. 4.15: Vývoj frekvencí závad pro druhé ložisko

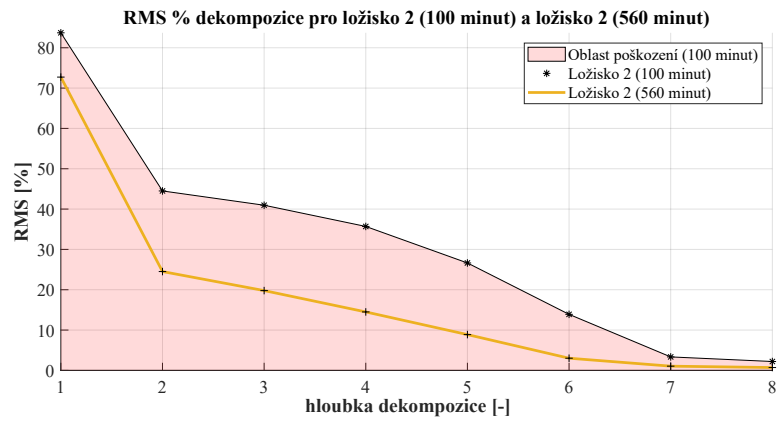
Vývoj efektivní hodnoty obálky je zobrazen grafem 4.16 níže.



Obr. 4.16: Vývoj efektivní hodnoty obálky pro druhé ložisko

## FWT dekompozice a podíl na celkovém RMS

V grafu 4.17 je porovnáno RMS % pro druhé ložisko po 100 minutách chodu a po 560 minutách chodu. Hodnota pro pozdější měření leží v červené oblasti, která signalizuje, že ložisko je v horším stavu, než po 100 minutách chodu.



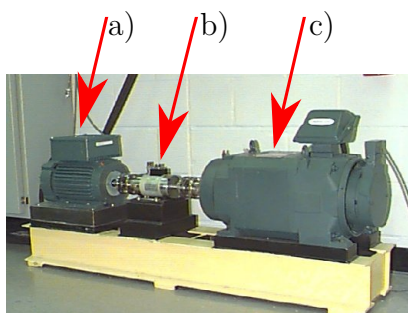
Obr. 4.17: Výsledek podílu RMS jednotlivých aproximací pro druhé ložisko

## 5 Klasifikace závady ložiska metodami strojového učení

Tato kapitola diplomové práce se zabývá automatizovanou klasifikací místa a velikosti závady ložisek metodami strojového učení. Algoritmy uvedené v této kapitole se vytrénují na datech, která jsou označená – u každého ze zaznamenaného průběhu vibrací máme informaci o tom, jestli se jedná o ložisko zdravé nebo s poškozením. V případě dat naměřených na poškozeném ložisku máme také informaci o míře poškození. Díky tomu lze poté automatizovaně klasifikovat nově změřená data v případě, že bylo zařízení provozováno ve stejných podmínkách, v jakých jsou zaznamenána i trénovací data.

### 5.1 Vibrační data ložisek

Jako zdrojový dataset vibračních dat ložisek byla vybrána často používaná volně přístupná databáze od Case Western Reserve University [13]. Tato databáze vznikla měřeními vibrací dvěma akcelerometry, jeden je umístěn na ložisku označeném jako *drive end* (SKF 6205-2RS JEM), druhý na ložisku označeném jako *fan end* (SKF6203-2RS JEM). Zařízení na testování ložisek je zobrazeno na obrázku 5.1 níže.



**Obr. 5.1:** Zařízení na testování ložisek z CWRU [41]

Tato ložiska jsou podporou pro hřídel motoru (5.1 a)). Tento motor má výkon 2 HP, je připojen přes snímač momentu (5.1 b)) k dynamometru (5.1 c)). Zátěž je volitelná v rozsahu 0–3 HP s krokem 1 HP (1 HP  $\doteq$  0,75 kW). Výkon motoru v metrických jednotkách je tedy 1,49 kW, zátěž je volitelná v rozsahu 0–2,24 kW s krokem 0,75 kW. Pro přehlednost budou dále v práci použity jednotky HP, jelikož se jedná o celá čísla. Motor pracuje v závislosti na aplikované zátěži v rozsahu 1730–1797 RPM. Do měřeného ložiska jsou vloženy závady na vnitřní kroužek, vnější kroužek a valivý element s přesně daným průměrem a hloubkou. Přesného poškození je dosaženo metodou elektroerozivního obrábění. Data z akcelerometrů jsou

zaznamenána vzorkovací frekvencí 12 kHz a 48 kHz. Pro každou variantu poškození a zatížení je zaznamenáno zhruba 10 sekund signálu.

V této práci jsou použity pouze data vzorkovaná na 48 kHz na ložisku *drive end* (SKF 6205-2RS JEM). Pro tuto vzorkovací frekvenci jsou dostupná data pro zdravé ložisko a pro závady o průměru 7 mils, 14 mils a 21 mils a hloubkou 11 mils (1 mils = 0,001 palce). V metrických jednotkách jde tedy o průměr 0,18 mm, 0,36 mm a 0,53 mm a hloubku 0,28 mm. Pro zvýšení přehlednosti je v této práci dále uváděn rozměr v jednotkách mils, jelikož se jedná o celá čísla. Vzhledem k tomu, že vnější dráha je při dané aplikaci ložiska stacionární, je důležité specifikovat lokaci poškození vzhledem k zóně zatížení. V této práci jsou použita data pro umístění závady přímo v zóně zatížení (tedy na šesti hodinách). Vibrační data použitá pro klasifikaci v této práci jsou shrnuta v tabulce 5.1(a), jedná se celkově o 40 souborů.

**Tab. 5.1:** Přehled dat z CWRU a rozdělení do tříd

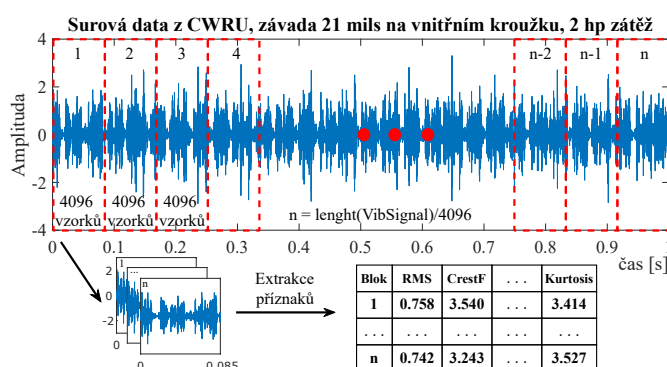
(a)						(b)		
Data z CWRU databáze vzorkovaná na 48 kHz						Označení tříd dat z CWRU		
Průměr [mils]	Zátěž [HP]	RPM	Vnitřní	Kulička	Vnější	Index třídy [-]	Místo/velikost [-][mils]	Zkrácený název
7	0	1797	I7_0	B7_0	O7_0	0	Zdravé/0	Healthy
	1	1772	I7_1	B7_1	O7_1	1	Kulička/7	Ball7
	2	1750	I7_2	B7_2	O7_2	2	Kulička/14	Ball14
	3	1730	I7_3	B7_3	O7_3	3	Kulička/21	Ball21
14	0	1797	I14_0	B14_0	O14_0	4	Vnitřní/7	Inner7
	1	1772	I14_1	B14_1	O14_1	5	Vnitřní/14	Inner14
	2	1750	I14_2	B14_2	O14_2	6	Vnitřní/21	inner21
	3	1730	I14_3	B14_3	O14_3	7	Vnější/7	Outer7
21	0	1797	I21_0	B21_0	O21_0	8	Vnější/14	Outer14
	1	1772	I21_1	B21_1	O21_1	9	Vnější/21	Outer21
	2	1750	I21_2	B21_2	O21_2			
	3	1730	I21_3	B21_3	O21_3			
0 (bez závady)	0	1797		H0_0				
	1	1772		H0_1				
	2	1750		H0_2				
	3	1730		H0_3				

Z tabulky 5.1(a) vyplývá, že data lze klasifikovat více způsoby. Nejpřirozeněji se jeví dva přístupy, prvním je rozlišovat 4 třídy poškození čistě podle poškozené komponenty, tedy {zdravé ložisko, vnější dráha, vnitřní dráha, kulička}. Druhý způsob je rozlišovat jak místo poškození, tak velikost poškození, zkoumaný signál tak lze rozdělit do 10 tříd. Druhý způsob, tedy rozlišení jak poškozené komponenty, tak průměru závady, je použitý v této práci. V tabulce 5.1(b) je rozlišen číselný index třídy, český ekvivalent a poté anglická zkratka používaná k označení třídy v grafech.

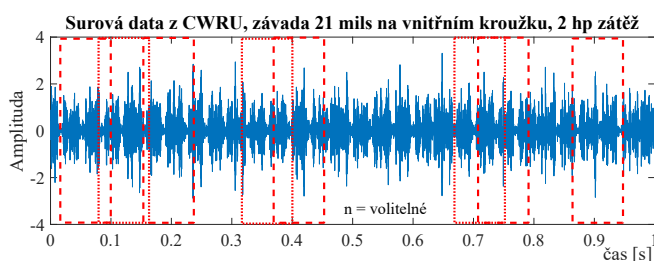
## 5.2 Klasifikace pomocí klasických metod STU

### 5.2.1 Příprava dat a extrakce příznaků

Pro klasické klasifikační algoritmy strojového učení používané v této práci je nejprve nutné extrahovat ze surových dat příznaky, které ideálně nesou informaci o stavu ložiska. Pro spolehlivou predikci je také nutné mít co největší množství naměřených dat. Z tabulky 5.1(a) výše lze vyčíst, že pro dané místo závady a daný průměr závady jsou k dispozici vždy jenom 4 měření o cca deseti sekundách, což není dostačující počet. Jedním z postupů, jak zvýšit množství dat pro učení a testování algoritmů, je rozdělit naměřená data do kratších bloků. Tento postup je zobrazen v grafu 5.2(a) níže.



(a) výběr bloků bez opakování vzorků



(b) výběr bloků s možným opakováním vzorků

**Obr. 5.2:** Rozdělení dat s a bez opakování vzorků

Jako kompromis mezi počtem pozorování a dostatečnou délkou bloku (a tedy i množstvím informací v něm obsažených) byla zvolena délka 4096 vzorků, což při dané vzorkovací frekvenci odpovídá zhruba 85 ms signálu. Touto metodou je možné z deseti sekund signálu vyextrahovat cca 117 pozorování (v grafu 5.2(a) označeno jako  $n$  bloků), pro jednu třídu jde tedy o cca 470 pozorování pro zátěž v rozsahu 0–3 HP.

V případě, že ani toto číslo není dostačující je možné zvolit z vibračního signálu

náhodné úseky o délce 4096 vzorků. Při tomto postupu lze vyextrahovat ze signálu libovolné množství pozorování, ale se zvyšujícím se počtem roste také riziko toho, že se některé vzorky budou v úsecích opakovat. Tato skutečnost může snížit věrohodnost získaných přesností predikce, protože v testovací množině se mohou vyskytovat některé vzorky z množiny trénovací. Přiblížení tohoto postupu je na obrázku 5.2(b), je zde naznačeno možné překrývání vybraných bloků. V této práci jsou použity obě metody rozdělení dat, které jsou popsány v textu výše. Pro možnost z obrázku 5.2(b), tedy pro výběr náhodných bloků dat byl zvolen počet bloků jako 400 na jeden soubor s vibracemi. Při tomto nastavení tedy ze 40 souborů, které jsou shrnuty v tabulce 5.1 dostaneme dohromady 16000 pozorování.

**Tab. 5.2:** Shrnutí příznaků

Souhrn příznaků z časové oblasti [42]

index	Název	Vztah
1	Směrodatná odchylka	$s = \sqrt{\frac{1}{N-1} \cdot \sum_{i=1}^N (x_i - \mu)^2}$
2	Šikmost	$skw = \frac{\frac{1}{N} \cdot \sum_{i=1}^N (x_i - \mu)^3}{\left(\sqrt{\frac{1}{N} \cdot \sum_{i=1}^N (x_i - \mu)^2}\right)^3}$
3	Špičatost	$k = \frac{\frac{1}{N} \cdot \sum_{i=1}^N (x_i - \mu)^4}{\left(\frac{1}{N} \cdot \sum_{i=1}^N (x_i - \mu)^2\right)^2}$
4	Peak2Peak	$\max(x) +  \min(x) $
5	RMS	$\sqrt{\frac{1}{N} \cdot \sum_{i=1}^N x_i^2}$
6	Crest Faktor	$c_f = \frac{ x_{peak} }{x_{RMS}}$
7	Shape Faktor	$\frac{x_{RMS}}{\frac{1}{N} \sum_{i=1}^N  x_i }$
8	Impulse Faktor	$\frac{ x_{peak} }{\frac{1}{N} \sum_{i=1}^N  x_i }$
9	Clearance Faktor	$\frac{ x_{peak} }{\left(\frac{1}{N} \cdot \sum_{i=1}^N \sqrt{ x_i }\right)^2}$
10	Energie	$\sum_{i=1}^N x_i^2$

Z takto získaných pozorování (bloků dat) o délce 4096 vzorků jsou poté extrahovány příznaky. Příznaky lze získat jak z časového průběhu, tak i ze spektra a z ostatních transformací signálu. V této práci jsou z důvodu snížení doby výpočtu vypočteny pouze příznaky z časového průběhu vibrací. Celkem bylo zvoleno 10 příznaků, které jsou popsány včetně vztahu v tabulce 5.2. Jedná se jak o základní statistické parametry signálu, tak o příznaky používané speciálně pro diagnostiku, jako je crest faktor a clearance faktor. [42]

Příznaky jednotlivých pozorování byly rozděleny do testovacích a trénovacích skupin a standardizovány metodou z-skóre. Vztah 5.1 pro výpočet z-skóre je uveden níže.

$$z_i^j = \frac{x_i^j - \text{mean}(x^j)}{s(x^j)} \quad (5.1)$$

Pro 5.1 je  $z_i^j$  standardizovaný j-tý příznak i-tého pozorování,  $x_i^j$  je původní hodnota j-tého příznaku pro i-té pozorování,  $\text{mean}(x^j)$  je výběrový průměr j-tého příznaku a  $s(x^j)$  je výběrová směrodatná odchylka j-tého příznaku.

Aby nebyla přenesena informace o testovací množině na množinu trénovacích, je testovací množina standardizována zvláště s pomocí parametrů  $\text{mean}(x^j)$  a  $s(x^j)$  vypočtených z trénovací množiny daného příznaku.

## 5.3 Nastavení parametrů klasických metod STU

Pro dosažení maximální možné přesnosti predikce je nutné nalézt optimální parametry metod strojového učení uvedených výše. Pro k-NN se v případě této práce jedná o nalezení optimálního počtu sousedů a také metriky vzdálenosti. Pro SVM se jedná o řád polynomiální jádrové funkce (kernelu). Pro Random Forest (RF) se jedná o počet stromů v lese. Pro neuronovou síť se jedná o počet skrytých vrstev a počet neuronů v těchto vrstvách.

### 5.3.1 Hledání optimálních parametrů modelu

K posouzení přesnosti predikce pro dané nastavení parametrů modelu lze použít například  $K$ -Fold křížovou validaci. Principem této metody je rozdělení datasetu na  $K$  bloků, které obsahují náhodný výběr pozorování z datasetu. Tyto bloky by ideálně měly mít stejný počet pozorování (pokud je celkový počet pozorování dělitelný beze zbytku číslem  $K$ ). Jeden z těchto  $K$  bloků je poté použit jako testovací, zbylý počet  $K-1$  bloků se spojí dohromady a tvoří trénovací množinu. Toto se opakuje  $K$ -krát, aby byl každý z  $K$  bloků použit jako testovací právě jednou. Rozdělení na bloky a postup křížové validace je zobrazen na obrázku 5.3.

Běh [-]	K-Fold Cross-Validation			
1	B1	B2	...	BK
2	B1	B2	...	BK
...	...	...	...	...
K	B1	B2	...	BK

Testovací      Trénovací

Obr. 5.3: Ukázka křížové validace

Číslo  $K$  se volí typicky jako 5, 10, 30. Když zvyšujeme číslo  $K$ , dojde ke zvýšení doby výpočtu, jelikož se model musí počítat  $K$ -krát. Nevýhodou této metody je překrývání jednotlivých trénovacích množin, každé dvě množiny spolu sdílí  $K-2$  bloky. [30]

Přesnost modelu s danými parametry se po provedení křížové validace určí jako průměr z  $K$  dosažených přesností na testovacích blocích.



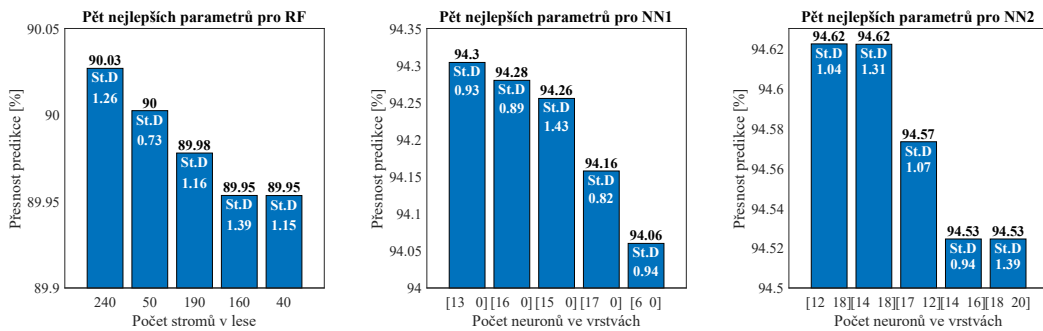
Pro jednotlivé klasické metody strojového učení ze sekce 3.1 byly zvoleny parametry a jejich rozsahy. S každou kombinací hodnot parametrů byla provedena 10-Fold křížová validace. Jako finální kombinace parametrů byla zvolena ta, která dosahovala při křížové validaci největší přesnosti predikce. Testované parametry jednotlivých metod jsou shrnuty v tabulce 5.3.

**Tab. 5.3:** Seznam testovaných parametrů

Testované parametry pro C-V		
Metoda	Název parametru	Hodnoty parametru
NN1	[Počet neuronů v 1. skryté vrstvě]	[5, 6, 7, .. 20]
NN2	[Počet neuronů v 1.   Počet neuronů v 2.]	[5, 6, 7, .. 20   5, 6, 7, .. 20]
k-NN	[Počet sousedů   Metrika vzdálenosti]	[1, 3, 5, .. 19   'Euclidean', 'Cosine', ... ... 'Chebychev', 'Mahalanobis', 'Hamming']
SVM	[Polyn. řád kernelu]	[1, 2, 3]
RF	[Počet stromů v lese]	[30, 40, 50, ... 250]

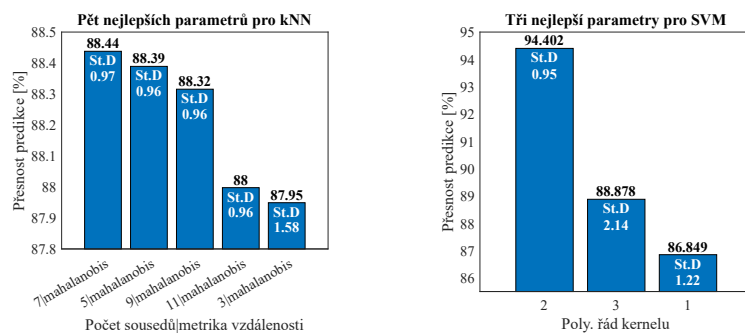
  

Detailní nastavení metod						
NN1 a NN2		k-NN	SVM	RF		
MLAB fce	<i>patternnet</i>	MLAB fce	<i>fitcknn</i>	MLAB fce	<i>fitcsemble</i>	
train. fce	<i>'trainlm'</i>	Dist. Weight	<i>'Equal'</i>	Kern. Scale	<i>'auto'</i>	
perf. fce	<i>'mse'</i>		Box Constr.	<i>1</i>	Method	<i>'bag'</i>
epochs	<i>1500</i>		Coding	<i>'onesone'</i>		
max_fail	<i>20</i>					
min_grad	<i>1e-10</i>					

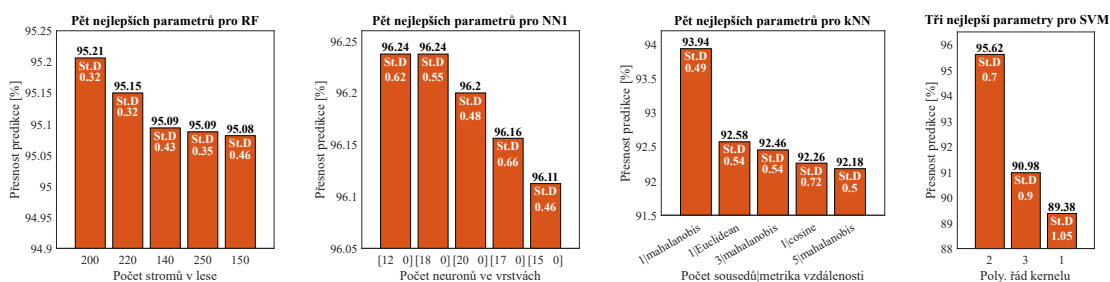


**Obr. 5.4:** Pět nejlepších parametrů získaných pomocí 10Fold C-V

V grafech 5.4 a 5.5 jsou zobrazeny výsledky pro několik nejlepších nastavení parametrů. Hodnota přesnosti predikce na ose  $y$  je průměrem z deseti testování během křížové validace,  $St.D$  u jednotlivých sloupců grafu označuje směrodatnou odchylku přesnosti pro deset testování. Výsledky jsou seřazeny zleva doprava od největšího. Pro neuronovou síť s jednou skrytou vrstvou je optimální počet neuronů 13. Pro neuronovou síť s dvěma skrytými vrstvami je nejlepší kombinace 12 a 18 neuronů ve skrytých vrstvách. Pro algoritmus k-NN je nejlepším nastavením 7 sousedů a metrika vzdálenosti 'Mahalanobis'. Pro SVM je neoptimálnější polynomiální jádrová funkce druhého řádu, optimální počet stromů v lese pro Random Forest je 240. V grafu



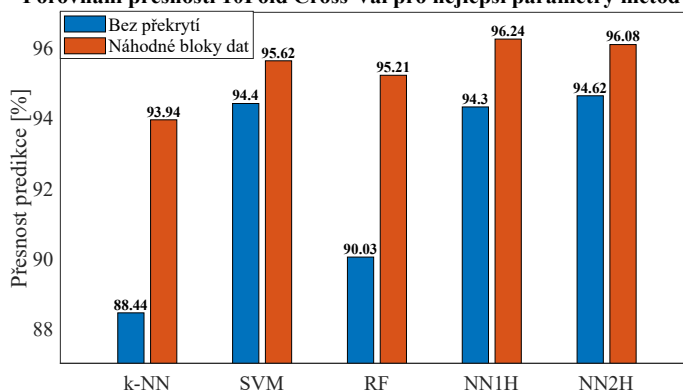
Obr. 5.5: Nejlepší parametry ostatních metod získané pomocí 10Fold C-V



Obr. 5.6: Pět nejlepších parametrů získaných pomocí 10Fold C-V, náhodné bloky dat

5.6 je srovnání pro větší počet pozorování popsány v sekci 5.2.1 pro výběr náhodných bloků dat. Pro neuronovou síť s dvěma vrstvami z důvodu časové náročnosti validace nebyl výběr optimálního počtu neuronů proveden, jsou použity hodnoty nalezené v grafu 5.4.

Porovnání přesností 10Fold Cross-Val pro nejlepší parametry metod



Obr. 5.7: Srovnání dvou způsobů generování příznaků ze sekce 5.2.1

V grafu 5.7 je ukázán rozdíl v přesnostech predikce v závislosti na způsobu výběru bloku vibračního signálu. Tyto dva způsoby jsou popsány v sekci 5.2.1. Větší množství příznaků zvýší přesnost především pro algoritmus k-NN a RF. U ostatních nedochází k tak významnému zvýšení přesnosti predikce.

## 5.4 Klasifikace pomocí CNN

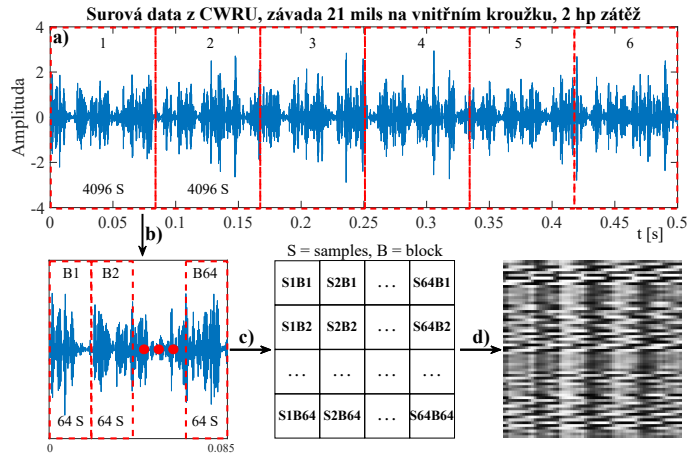
### 5.4.1 Příprava dat pro CNN

Standardní metody strojového učení zmíněné v kapitole 3.1 vyžadují pro funkční klasifikaci extrakci příznaků. Tento krok přináší do automatizované predikce poruchy komplikace. Hlavním problémem je, že musíme z průběhu extrahovat takové příznaky, které nesou informaci o místě poškození ložiska a velikosti poškození ložiska. Obecně je možné vyextrahovat libovolné množství příznaků jak z časové, tak frekvenční oblasti a případně i z jiných transformací signálu. Z tohoto velkého množství příznaků lze potom vybrat podmnožinu, s kterou je přesnost predikce nejvyšší, například metodou dopředné/zpětné selekce příznaků. V této práci bylo v kapitole 5.2.1 vybráno 10 příznaků z časové oblasti. Tyto příznaky, seřazené v tabulce 5.2 nejsou zvoleny náhodně, jedná se například o crest faktor a špičatost, které se běžně používají pro diagnostiku stavu ložisek. Popis některých z těchto příznaků je v kapitole 2.1.3 a 2.1.4. Výpočet příznaků může také být v závislosti na počtu dat časově náročný.

Výhodou konvolučních neuronových sítí je to, že ruční výběr a extrakce příznaků odpadá. Je tomu tak proto, že konvoluční neuronové sítě provádí extrakci příznaků automaticky ve své první části – v konvolučních vrstvách. Poté jsou tyto vyextrahované příznaky klasifikovány druhou částí konvoluční sítě – tzv. plně propojenou vrstvou.

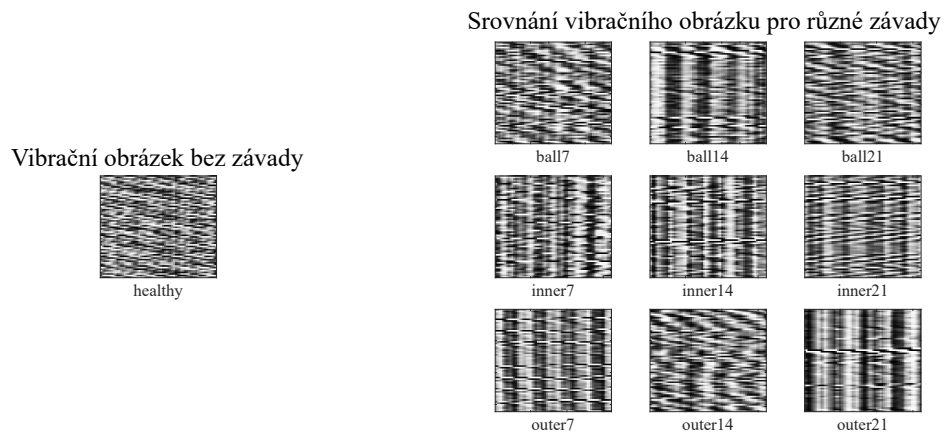
Konvoluční neuronové sítě jsou často používány a existují mnohé ověřené architektury, problémem ovšem je, že jejich využití je v oblastech 2D signálů – klasifikace obrázků. Tento problém se dá u 1D signálu získaného z akcelerometru připevněného na ložisku obejít tak, že provedeme transformaci 1D signálu na 2D signál. Obecně je možné více postupů, například využitím krátkodobé Fourierovy transformace (STFT). Výsledný spektrogram je poté použit jako 2D vstup do konvoluční neuronové sítě.

V této práci je použita metoda vytvoření vibračního obrázku. Jedná se o nejjednodušší způsob transformace 1D signálu na 2D signál. Princip metody je zobrazen na obrázku 5.8. Nejprve je nutné určit si výslednou velikost výstupního obrázku. Příliš malý obrázek neobsahuje dostatek informací o stavu ložiska a v případě příliš velkého obrázku by bylo málo pozorování pro jednotlivé třídy poškození. V rámci této práce je použita velikost 64x64 (4096) pixelů. Jedná se o stejnou velikost, jako je délka bloků, ze kterých se vypočítávají příznaky u standardních metod strojového učení popsanych v podkapitole 5.2.1. Přesnost predikce se tedy dá mezi standardními metodami strojového učení a CNN objektivně porovnávat.



**Obr. 5.8:** Ukázka převodu 1D signálu na 2D obrázek

Postup pro výrobu vibračního obrázku podle 5.8 je následující. 1D signál z vibračního snímače se rozdělí na bloky po 64x64 (4096) vzorcích (obrázek 5.8 a)). Tyto bloky o 4096 vzorcích se poté ještě rozdělí na 64 bloků o 64 vzorcích (obrázek 5.8 b)). Poté se vytvoří matice o 64 řádcích a 64 sloupcích (obrázek 5.8 c)). Každý řádek matice je tvořen jedním blokem o 64 vzorcích, každá jedna buňka matice tedy odpovídá právě jednomu vzorku z tohoto bloku. Výstupem je matice hodnot. Hodnoty v matici mohou být obecně v jakémkoliv rozsahu. Pro vytvoření obrázku se tedy ještě normalizují na uint8 hodnoty (0–255). Tuto matici už poté lze převést na obrázek v odstínech šedé (grayscale image) (obrázek 5.8 d)). Tento krok je proveden pro všechny ostatní bloky o 4096 vzorcích, vznikne počet obrázků odpovídající počtu vzorků vibračního signálu a velikosti obrázku, tedy  $length(VibSignal) / 4096$ . Vibračních obrázků pro klasifikaci závad na ložiscích používá například článek [44].



**Obr. 5.9:** Ukázka vibračního obrázku pro různé velikosti a místa závady

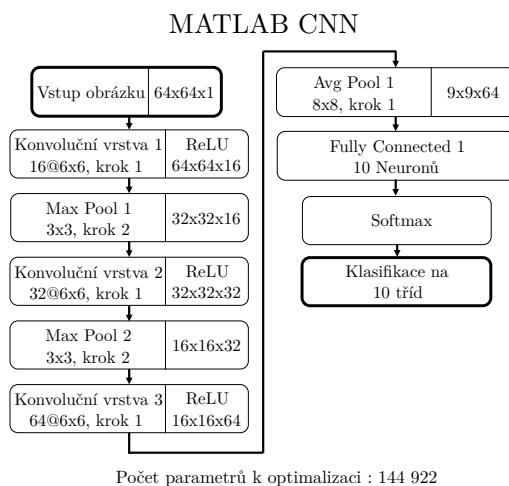
Podle zjištění v článku [44] je u vibračních obrázků v této práci provedena také ekvalizace histogramu. Díky tomuto kroku je vylepšen kontrast obrázku, můžou tak lépe vyniknout vzory na obrázcích náležící jednotlivým zavadám.

Když je tento krok proveden pro všechny typy a velikosti poškození ložiska, dostáváme obrázky, které v sobě nesou informaci o závadě. Pro deset tříd uvedených v tabulce 5.1 b) jsou příklady vibračních obrázků zobrazeny v 5.9. Pro člověka jsou tyto obrázky jen těžko interpretovatelné, ale je vidět, že obrázky různých typů a velikostí poškození se liší. V následujícím textu bude ukázáno, že vzory na obrázcích jsou pro konvoluční vrstvy CNN rozpoznatelné.

## 5.5 Architektury CNN použité v této práci

### 5.5.1 MATLAB CNN

Jako první konvoluční síť byla zvolena architektura z příkladu na mathworks stránkách [43]. V příkladu uvedeném na zmíněných stránkách je vytvořena ukázková CNN pro klasifikaci obrázků o velikosti 32x32x3 (RGB). Síť má programově ovládaný počet konvolučních vrstev ve třech konvolučních blocích. Na obrázku 5.10 je zobrazena síť s jednou konvoluční vrstvou v konvolučním bloku, jelikož je to pro úlohu klasifikace závady z vibračního obrázku nejlepší možnost, což bude zjištěno v sekci 5.7.1. Pojmy na obrázku 5.10 jsou vysvětleny na konvoluční vrstvě 1. V levé části buňky je název dané vrstvy a parametry dané vrstvy – v tomto případě jde o 16 konvolučních filtrů s rozměry 6x6 a krokem 1. V pravé části buňky je uveden rozměr výstupu z dané vrstvy, v tomto případě je výstupem 16 map o rozměru 64x64. Aby šla síť použít na vstupní data o rozměru 64x64x1 použitá v této práci, je u sítě upravena vstupní vrstva a výstupní plně propojená vrstva. Dále je zvětšena velikost konvolučního filtru na 6x6 pro všechny konvoluční vrstvy. U konvolučních vrstev je nastavena možnost výplně (padding) na 'same' s hodnotami '0', MATLAB tedy automaticky určí počet řádků a sloupců pro výplň (padding) v závislosti na velikosti konvolučního filtru a kroku posuvu tohoto filtru. Za každou konvoluční vrstvou také následuje vrstva normalizace dávky (batch normalization).

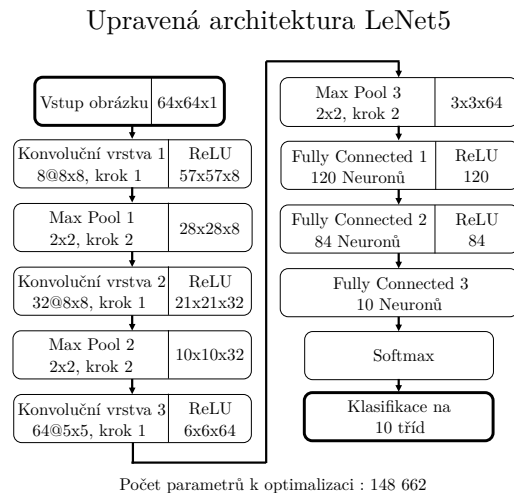


Obr. 5.10: Struktura a parametry zvolené architektury MATLAB CNN

### 5.5.2 Vylepšená LeNet-5

Tato architektura je popsána a otestována na datech z CWRU v článku [44]. Autor článku modernizuje jednu z nejstarších architektur CNN, takzvanou LeNet-5

architekturu [45]. Tato síť byla původně navržena pro rozpoznávání ručně psaných číslic. Oproti původní architektuře LeNet-5 byla v [44] přidána navíc jedna konvoluční vrstva se slučující vrstvou (max pool) a byly změněny rozměry a počet filtrů v konvolučních vrstvách. Také byly přidány vrstvy normalizace dávky (batch normalization) za konvoluční vrstvy a u dvou plně propojených vrstev byla přidána vrstva vypuštění neuronů (dropout layer) s parametrem  $p = 0,2$  pro zlepšení generalizace sítě. Tato síť má zhruba stejný počet parametrů k učení jako síť MATLAB CNN ze sekce 5.5.1 výše.

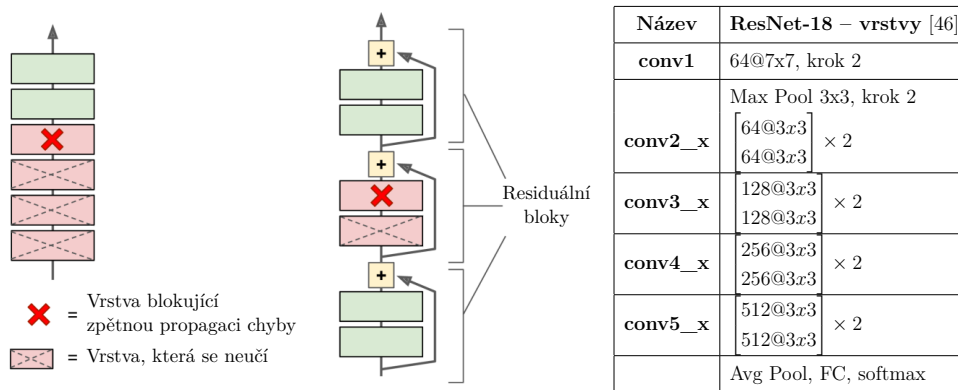


**Obr. 5.11:** Struktura a parametry zvolené architektury LeNet-5

### 5.5.3 ResNet-18

Nejhlubší CNN v této práci je architektura ResNet-18, která má hloubku 18 vrstev (když se počítají pouze konvoluční a plně propojené vrstvy). Tato architektura je navržena v [46]. Síť byla upravena tak, aby měla vstup na obrázek  $64 \times 64 \times 1$  a výstup klasifikoval 10 tříd. Byla tedy upravena vstupní vrstva a výstupní plně propojená vrstva. Tato síť má  $11 \cdot 2 \cdot 10^6$  parametrů, jedná se o dva řády více než má architektura LeNet-5 popsaná v sekci 5.5.2.

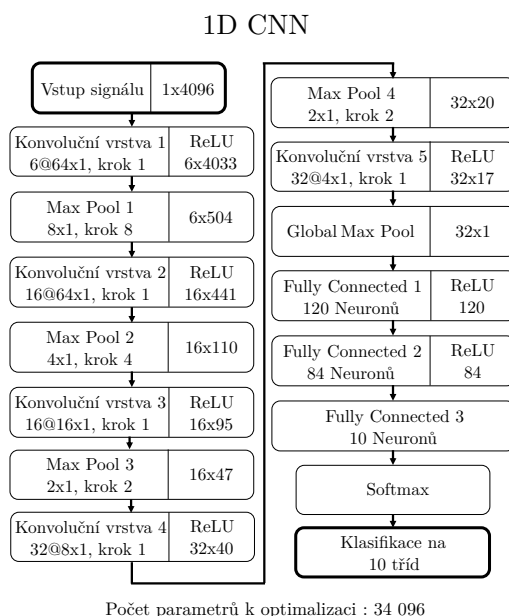
Hlavním znakem ResNet sítí je přeskokování konvolučních bloků, viz obrázek 5.12. Proces učení je u této architektury vylepšen tím, že je za sebou poskládáno několik bloků s konvolučními vrstvami. Mezi těmito bloky jsou navíc přidány přímé odbočky směrem k výstupu sítě. Při zpětné propagaci chyby díky této úpravě nedochází k blokaci vrstvami, které se ještě nezačaly učit. Za konvolučními vrstvami následuje vrstva normalizace dávky (viz sekce 3.2.2), aktivační funkce je ReLU. [32]



Obr. 5.12: Residuální jednotky ([32], upraveno); ResNet-18, [46]

### 5.5.4 1DCNN

Jelikož jsou výše uvedené konvoluční sítě určené pro klasifikaci obrázků, je kromě procesu učení a predikce dat nutné provést krok navíc – transformaci z 1D vibračních dat na 2D vibrační obrázek. Tento krok odpadá u 1D konvolučních sítí, které mají jako vstup přímo surová vibrační data. Architektura použitá v této práci na obrázku 5.13 je navržena v článku [44]. Rozdílem je, že po konvoluční vrstvě 5 následuje slučující vrstva (global max pool), která sníží velikost dimenze na 1D (maximum ze 17 hodnot pro všech 32 map). Za každou konvoluční vrstvou následuje normalizace dávky (batch normalization), u dvou plně propojených vrstev také vrstva vypuštění neuronů (dropout layer) s parametrem  $p = 0,2$  pro zlepšení generalizace sítě.



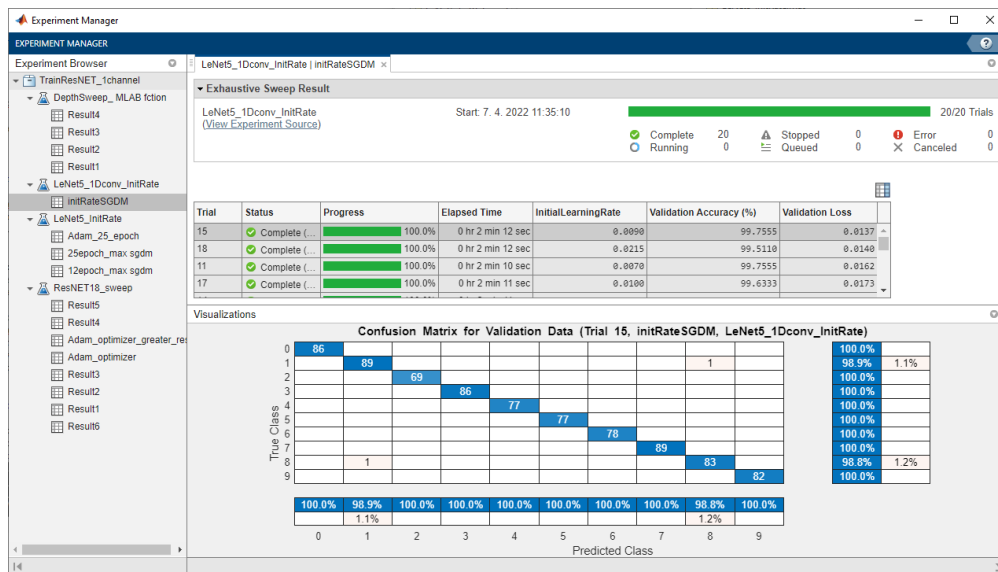
Obr. 5.13: Struktura a parametry zvolené architektury 1DCNN



## 5.6 Nastavení optimálních parametrů CNN

V podkapitole 5.3.1 bylo uvedeno, že pro maximální přesnost predikce je nutné najít pro jednotlivé metody strojového učení vhodné nastavení parametrů. U konvolučních neuronových sítí je především [32] nutné nalézt optimální velikost kroku pro algoritmus SGDM (Stochastic Gradient Descent with Momentum). Špatná volba kroku může vést i k tomu, že algoritmus optimalizace parametrů sítě pro daný problém nenajde řešení a predikce vůbec nefunguje.

Pro hledání optimálního kroku učení architektur CNN použitých v této práci byla použita MATLAB aplikace *Experiment Manager*. Prostředí této aplikace je na obrázku 5.14.



Obr. 5.14: Prostředí MATLAB aplikace Experiment Manager

Cílem této aplikace je otestovat kombinace libovolných parametrů a nalézt optimální kombinaci hodnot. Aplikaci je třeba definovat strukturu testované sítě a také učící funkci. Pro všechny parametry, které chceme otestovat, je nutné zadat vektor hodnot, například velikost kroku nebo velikost parametru momentum. Pro příklad, cílem je zjistit optimální hodnotu kroku učení, jako proměnný parametr pro test je tedy využít *InitialLearningRate* v rozsahu  $10^{-4}$ – $10^{-1}$  s logaritmickým rozložením. Sít se poté automaticky vytrénuje s každým z deseti parametrů v tomto vektoru. Hodnocením kvality aktuálního nastavení parametrů může být jakákoliv uživatelsky zvolená metrika, v tomto případě byla použita hodnota chybové funkce. Po dokončení testování lze seřadit parametry od nejlepšího a finální síť trénovat s optimálními parametry.

Aplikaci *Experiment Manager* při použití možnosti *Exhaustive Sweep* lze v podstatě nahradit i kódem v MATLABu s několika vnořenými *for* cykly, kdy počet

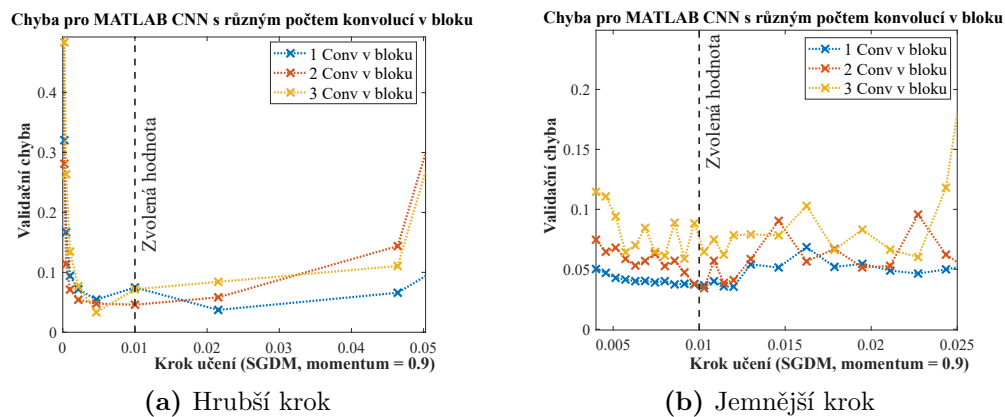
for cyklů odpovídá počtu testovaných parametrů. Aplikace ale umožňuje testovat přehledněji, výsledky lze jednoduše filtrovat, seřazovat a vizualizovat.

## 5.7 Nalezení optimální hodnoty kroku učení

### 5.7.1 MATLAB CNN

MATLAB CNN má parametrizován počet za sebou následujících konvolučních vrstev ve třech konvolučních blocích. Byly testovány možnosti s jednou konvoluční vrstvou v bloku, se dvěma a se třemi konvolučními vrstvami v bloku. V bloku konvolučních vrstev mezi těmito vrstvami není žádná slučující vrstva (pooling layer), jde čistě o za sebe zapojené konvoluční vrstvy s normalizací dávky (BN), podobně jako je tomu u architektury ResNet. Zároveň byl pro každou z těchto tří možností testován krok učení v rozsahu  $10^{-4}$ – $10^{-1}$  s logaritmickým rozložením, tedy vektor deseti hodnot  $[0.0001, 0.0002, 0.001, \dots, 0.0464, 0.1000]$ . Tímto hrubým krokem byl zjištěn užší interval, který byl dále testován přesněji. Výstupem byla síť s parametry z poslední trénovací iterace, trénovalo se na 25 epoch.

Pro grafické nalezení ideální hodnoty je možné zobrazit závislost hodnoty ztrátové funkce na velikosti kroku učení a současně na počtu konvolucí v bloku. Závislost má vanový tvar – pro moc malé a moc velké hodnoty kroku učení je ztráta velká, pro optimální hodnotu dosahuje minima.



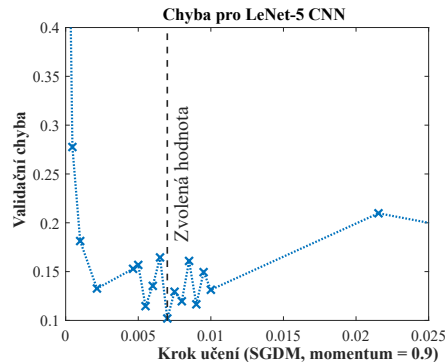
**Obr. 5.15:** Nalezení optimálního kroku pro učení MATLAB CNN

Pro MATLAB CNN je tato závislost v grafu 5.15. V levém grafu je nejdříve hrubé nalezení minima, v pravém grafu poté bližší zjištění ideální hodnoty. Z 5.15a) se zdá, že lepší jsou 2 konvoluce v bloku, jelikož ale trénování s popořadě jednou, dvěma a třemi konvolucemi v bloku trvá přibližně  $\{6, 10, 20\}$  minut, je z důvodu úspory času u následujících výpočtů zvolena možnost s jednou konvolucí. V grafu 5.15b) lze

také pozorovat, že pro jemnější krok je možnost s jednou konvolucí v bloku lepší pro širší rozsah hodnot kroku učení. Zvolenými parametry jsou tedy jedna konvoluce v bloku a krok učení 0,01.

### 5.7.2 LeNet-5 CNN

Pro architekturu LeNet-5 byl v aplikaci *Experiment Manager* testován pouze krok učení. Výstupem byla síť s parametry z poslední iterace, trénovalo se na 25 epoch.

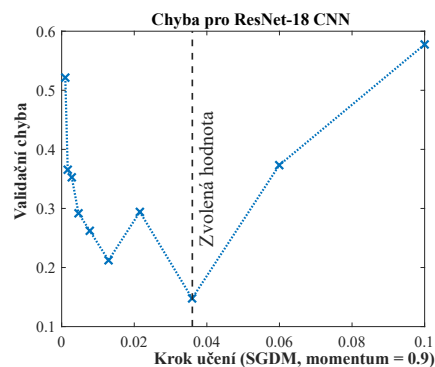


Obr. 5.16: Nalezení optimálního kroku pro učení LeNet-5 CNN

Zvolený krok učení 0,007 je vyznačen v grafu 5.16.

### 5.7.3 ResNet-18 CNN

Pro architekturu ResNet-18 byl v aplikaci *Experiment Manager* testován pouze krok učení. Výstupem byla síť s parametry z nejlepší iterace, trénovalo se na 10 epoch.

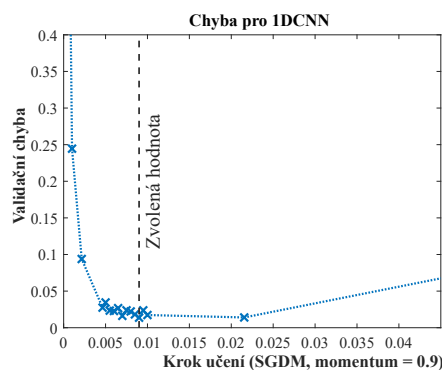


Obr. 5.17: Nalezení optimálního kroku pro učení ResNet-18 CNN

Zvolený krok učení 0,036 je vyznačen v grafu 5.17.

## 5.7.4 1DCNN

Pro architekturu 1DCNN byl v aplikaci *Experiment Manager* testován pouze krok učení. Výstupem byla síť s parametry z poslední iterace, trénovalo se na 15 epoch.



**Obr. 5.18:** Nalezení optimálního kroku pro učení 1DCNN

Zvolený krok učení 0,009 je vyznačen v grafu 5.18.

Jednotlivé architektury CNN jsou v následující části práce použity s parametry, které byly zjištěny nástrojem *Experiment Manager* v sekcích výše. Takto nalezené nastavení parametrů metod nedá nazývat přímo jako optimální, k tomu by bylo nutné testovat kombinace všech parametrů. Kromě kroku učení jde také o nastavení parametru *momentum*, velikost *mini-batch* a dalších. Z důvodu velké časové náročnosti testování kombinací byl zkoumán pouze vliv učícího kroku, i přesto se s metodami dá dosáhnout dostatečných přesností predikce.

## 5.8 Vliv velikosti trénovací množiny na přesnost predikce

V reálné aplikaci nemusí být pro učení dostupné dostatečné množství označených dat. Aby se otestovala závislost přesnosti predikce jednotlivých algoritmů strojového učení na velikosti trénovací množiny, byla provedena série testů s postupně zmenšující se trénovací množinou. Výstupní model byl poté vždy otestován na zbytku dat. Pro každou velikost trénovací množiny a každý algoritmus bylo provedeno trénování pětkrát, průměr přesností přes těchto pět testování byl vyneseno do grafů 5.19–5.22. Velikost trénovací množiny byla z celkového počtu pozorování testována v rozsahu 90 %–40 % s krokem 10 % a poté pro 20 %. Výsledky jsou prezentovány pro dva různé způsoby extrakce pozorování z dat, které jsou popsány na obrázku 5.2. Prvním je vybrání bloků postupně, tedy bez opakování vzorků, druhým je vybrání bloků náhodně, tedy s možným opakováním vzorků. Druhým způsobem je možné značně zvýšit počet pozorování, přesnosti predikce jsou tedy vyšší.

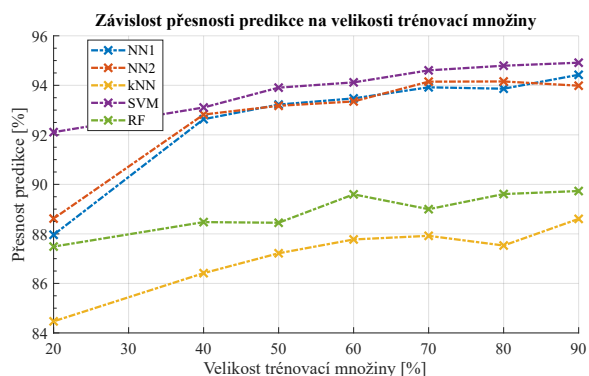
Prezentace výsledků je vždy na dvou souvisejících grafech. V levém grafu je zobrazena závislost absolutních přesností predikce na velikosti trénovací množiny. Pro objektivní porovnání odolnosti algoritmů na zmenšování datasetu je ale vhodnější normalizování přesnosti predikce. Na pravém grafu jsou tyto hodnoty normalizovány vzhledem k přesnostem predikce, kterých algoritmy dosahují při velikosti trénovací množiny 90 %. Díky tomu lze objektivně srovnat odolnost přesnosti predikce na zmenšování trénovací množiny.

Pro různé velikosti trénovací množiny probíhá učení vždy pětkrát. Zobrazení těchto pěti průběhů se dá dobře shrnout v krabicovém grafu (boxplot), ovšem velký počet grafů v textu zhoršuje přehlednost. V případě zájmu lze tedy výsledky v krabicových grafech najít v příloze A.

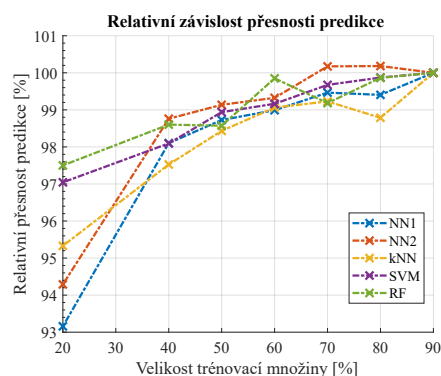
### 5.8.1 Výsledky pro standardní metody STU

Přesnost predikce neuronové sítě s jednou skrytou vrstvou (NN1) při postupném zmenšování trénovací množiny mírně klesá, pohybuje se ale zhruba kolem hodnoty 93 %. K poklesu pod 90 % dochází až při omezení trénovací množiny na 20 %. Podobná je situace také pro neuronovou síť s dvěma skrytými vrstvami. Nejlepšího výsledku ze standardních metod strojového učení dosáhl algoritmus SVM, který má pro všechny velikosti trénovací množiny přesnost nad 92 %. Toto je očekávaný výsledek, jelikož SVM dokáže dobře pracovat s relativně malými datasey [32].

Z grafu 5.19 b) pro relativní přesnosti predikce vyplývá, že relativně nejodolnějšími algoritmy jsou RF a SVM. Algoritmus RF nedosahuje dostatečně dobré absolutní přesnosti predikce (max. 90 %). Algoritmus SVM dosahuje jak nejvyšší



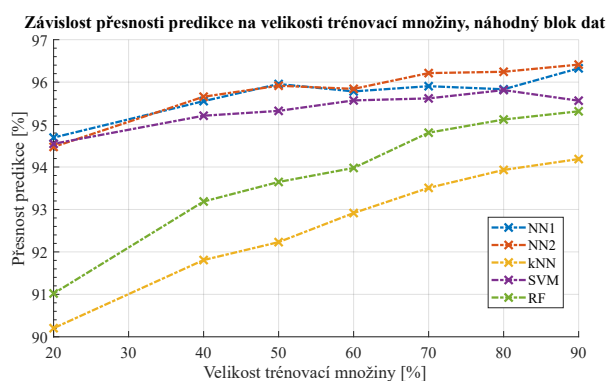
(a) Absolutní přesnosti predikce



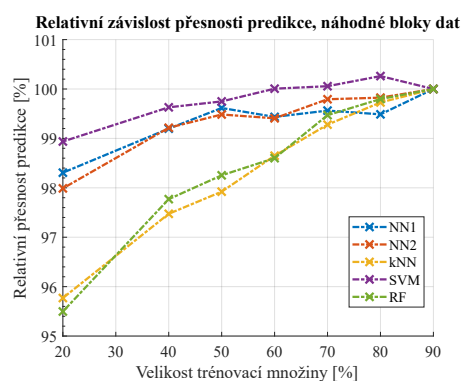
(b) Relativní přesnosti predikce

**Obr. 5.19:** Závislost přesnosti predikce na velikosti trénovací množiny pro standardní ML algoritmy

přesnosti predikce mezi standardními metodami strojového učení (max. 95 %), tak také druhé nejlepší relativní odolnosti na zmenšování trénovací množiny. SVM tak lze jak z pohledu absolutní přesnosti predikce, tak z pohledu odolnosti vůči zmenšení trénovací množiny prohlásit za nejlepší volbu.



(a) Absolutní přesnosti predikce



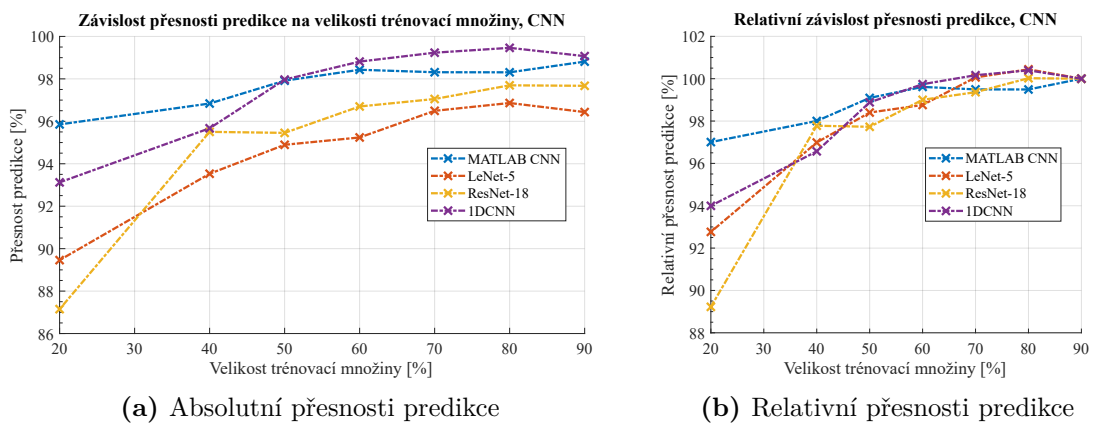
(b) Relativní přesnosti predikce

**Obr. 5.20:** Závislost přesnosti predikce na velikosti trénovací množiny pro standardní ML algoritmy, náhodně zvolené bloky dat

V grafu 5.20 jsou zobrazeny přesnosti predikce v závislosti na velikosti trénovací množiny pro bloky pozorování vybrané náhodně. Při srovnání grafu 5.19 a), který pracuje s celkovým počtem cca 4100 pozorování a grafu 5.20 a), který pracuje s 16000 pozorováními je vidět, u druhé možnosti dosahují všechny algoritmy vyšší absolutní přesnosti. Také se zmenšováním velikosti trénovací množiny nedochází k tak velkému relativnímu poklesu přesnosti. Toto je způsobeno tím, že i 20 % trénovací množiny u druhé možnosti odpovídá stále 3200 pozorování, kdežto u první možnosti odpovídá 20 % trénovací množině jen cca 800 pozorování.

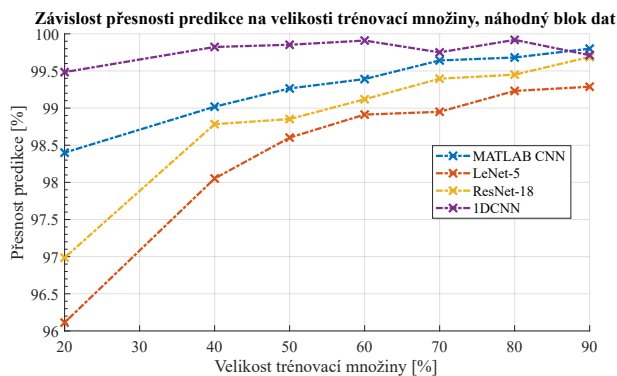
## 5.8.2 Výsledky pro CNN

Pro konvoluční neuronové sítě je situace obdobná jako pro standardní algoritmy strojového učení. Závislosti přesností predikce na velikosti trénovací množiny jsou v grafu 5.21. Všechny architektury si udržují dobrou přesnost do velikosti trénovací množiny 40 %, poté přesnost predikce klesá strměji. Výjimkou je architektura MATLAB CNN, která i pro velikost trénovací množiny 20 % dosahuje přesnosti predikce 95,85 %. Toto zjištění může být výhodné, protože pro větší trénovací množiny dosahuje nejlepších výsledků 1DCNN (pro trénovací množinu od 90 % do 70 % je přesnost predikce  $> 99$  %), ale pro velikost trénovací množiny 20 % už dosahuje jenom přesnosti 93,12 %. Z tohoto porovnání se tedy MATLAB CNN architektura jeví jako univerzálnější volba, protože i pro velké množství trénovacích dat dosahuje uspokojivé přesnosti nad 98 %.

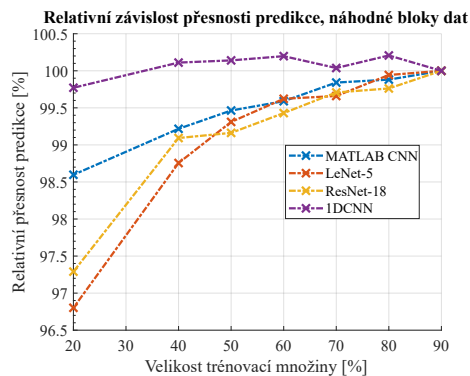


**Obr. 5.21:** Závislost přesnosti predikce na velikosti trénovací množiny pro CNN

Z grafu 5.21 b) vyplývá, že relativně nejodolnější architekturou CNN je s přehledem MATLAB CNN. Tato architektura také dosahuje druhé nejlepší absolutní přesnosti predikce (max. 98,8 %) za 1DCNN architekturou. 1DCNN architektura má absolutní přesnost predikce max. 99,5 % a v rámci relativní odolnosti vůči zmenšení trénovací množiny je druhá právě za MATLAB CNN architekturou. Vzhledem k malému poklesu relativní přesnosti predikce pro MATLAB CNN architekturu (ze 100 % na 95,9 %) lze architekturu MATLAB CNN prohlásit za vhodný kompromis mezi absolutní přesností a odolností vůči zmenšení trénovací množiny.



(a) Absolutní přesnosti predikce



(b) Relativní přesnosti predikce

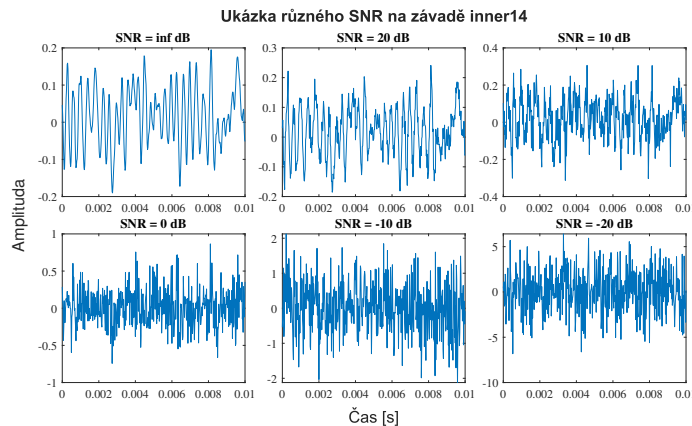
**Obr. 5.22:** Závislost přesnosti predikce na velikosti trénovací množiny pro CNN, náhodně zvolené bloky dat

V grafu 5.22 je zobrazena závislost přesnosti predikce na velikosti trénovací množiny pro náhodně zvolené bloky dat. Situace je podobná jako v grafu 5.21. Rozdílem je, že pro náhodně vybrané bloky dat je jak v relativní, tak v absolutní přesnosti predikce nejlepší 1DCNN. Tato architektura dosahuje pro všechny velikosti trénovací množiny přesnosti nad 99,5 %. Druhou nejlepší architekturou je poté MATLAB CNN.



## 5.9 Vliv šumu na přesnost predikce

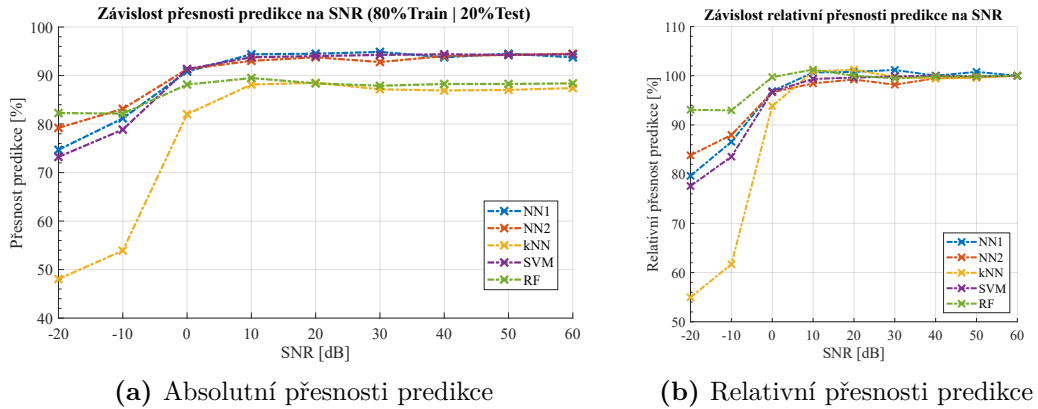
Pro otestování odolnosti klasifikačních algoritmů na šum byl do originálních vibračních dat přidán bílý šum s definovaným SNR. K přidání šumu do vibračních signálů byl použit příkaz `awgn(VibSignal,SNR,'measured','dB')`. SNR byl otestován v rozsahu od 60 dB do -20 dB s krokem 10 dB. Při zpětné kontrole výsledků by bylo lepší zvolit rozsah -20–20 dB s drobnějším krokem, jelikož pro  $\text{SNR} > 30$  dB je procentuální podíl šumu na výsledném signálu téměř zanedbatelný, přesnost predikce tedy není příliš ovlivněna. Ukázka průběhu vibrací pro 10 ms a závadu na vnitřním kroužku je na obrázku 5.23.



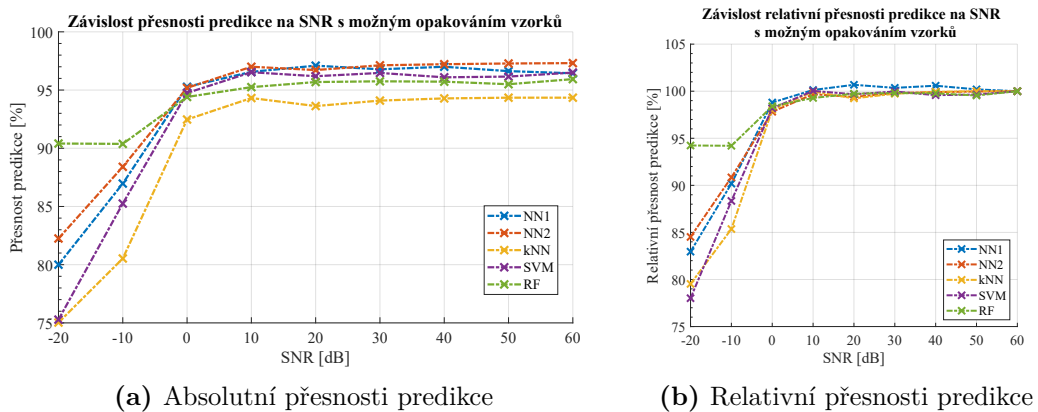
Obr. 5.23: Signál z poškození inner14 při různých SNR

Jednotlivé algoritmy byly testovány pro danou hodnotu SNR, přesnost predikce zobrazená v grafech níže je jenom pro jeden běh trénování. Z průměrování většího počtu trénování, tak jak tomu bylo v předchozí sekci, bylo ustoupeno. Hlavním důvodem je fakt, že opakované trénování všech modelů trvá značně dlouho. Pro standardní algoritmy strojového učení je výsledná přesnost predikce v závislosti na velikosti přidaného šumu v grafu 5.24. Je zřejmé, že pro všechny metody je průběh podobný. Výjimkou je algoritmus k-NN, kdy je pro hodnotu  $\text{SNR} = -20$  dB přesnost predikce pod 50 %. Nejlepší predikce pro  $\text{SNR} = -20$  dB dosahuje algoritmus RF (82 %) a poté neuronová síť se dvěma skrytými vrstvami (79 %).

Algoritmus RF při menším množství přidaného šumu nedosahuje tak velké absolutní přesnosti jako neuronová síť (88,3 % vs 94,5 %). Neuronová síť s dvěma skrytými vrstvami je tak lepší volbou pro méně zašuměná data.



**Obr. 5.24:** Odolnost standardních algoritmů STU na přidaný šum



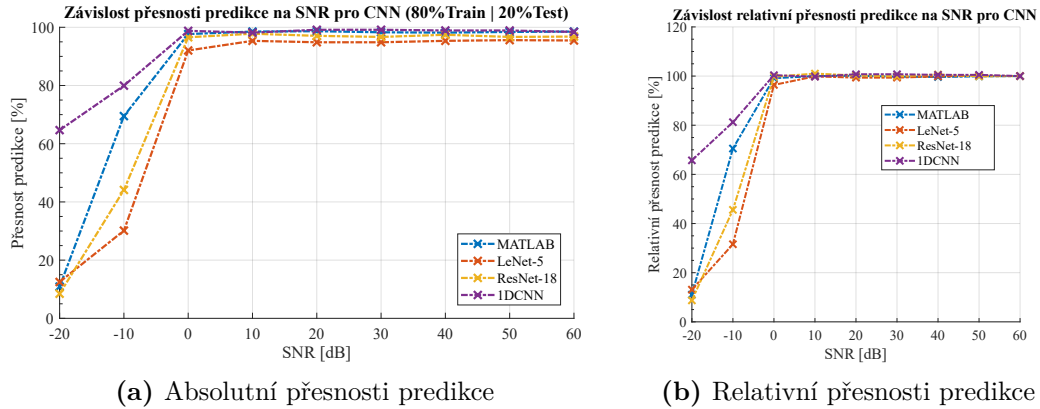
**Obr. 5.25:** Odolnost standardních algoritmů STU na přidaný šum, náhodně zvolené bloky dat

Výsledek pro různé architektury konvolučních neuronových sítí je v grafu 5.26. Situace je jiná než u klasických metod strojového učení. U architektur, které mají 2D vstup je patrné, že pro SNR = -20 dB a -10 dB je predikce velmi špatná, pro -20 dB je dokonce úplně nefunkční (10% přesnost predikce pro 10 tříd odpovídá spíše náhodnému tipu).

Důvodem může být to, že standardní metody strojového učení z grafu 5.24 mají jako vstup statistické parametry vibračního signálu, kdežto 2D CNN algoritmy mají jako vstup upravený obrázek. Problémový je zřejmě proces normalizace hodnot pixelů u obrázku. Kvůli velkému podílu šumu dojde ke kompletnímu zastínění podpisu

závady v obrázku.

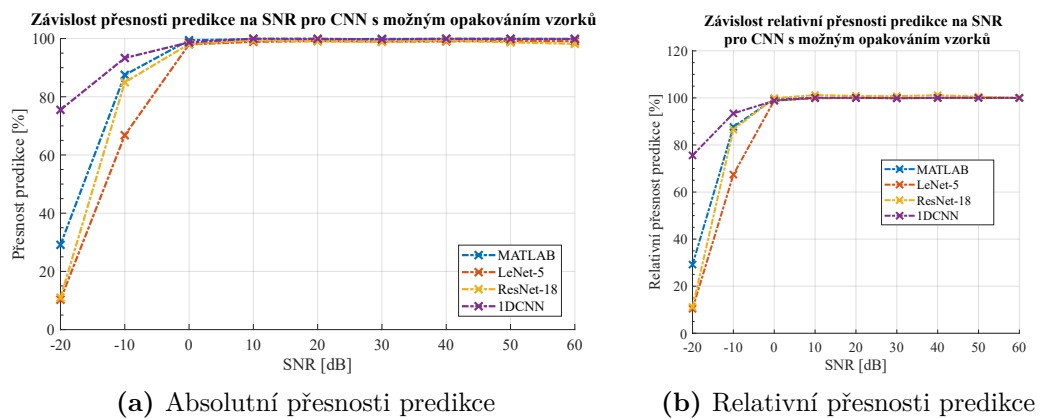
Pro velmi zašuměná data se tedy z architektur CNN hodí 1D konvoluční síť, která má nejlepší výsledky bez ohledu na množství přidaného šumu. Druhá nejlepší je architektura MATLAB CNN ze sekce 5.5.1. Ta dosahuje pro  $\text{SNR} = -10$  dB přesnosti predikce 69 % oproti nejlepší 1DCNN (79 %) a třetí nejlepší ResNet-18 (44 %).



**Obr. 5.26:** Odolnost CNN architektur na přidaný šum

V grafu 5.27 je zobrazena situace pro náhodně zvolené bloky dat. Větší velikost trénovací množiny nezlepší výsledky CNN pro velmi zašuměná data, situace je podobná jako pro graf 5.26. Pro  $\text{SNR} = -20$  dB je situace lepší jen pro MATLAB CNN, která začíná predikovat přesněji než náhodný tip (30 % oproti 10 %).

I tak se jedná oproti standardním metodám STU o velmi špatný výsledek.

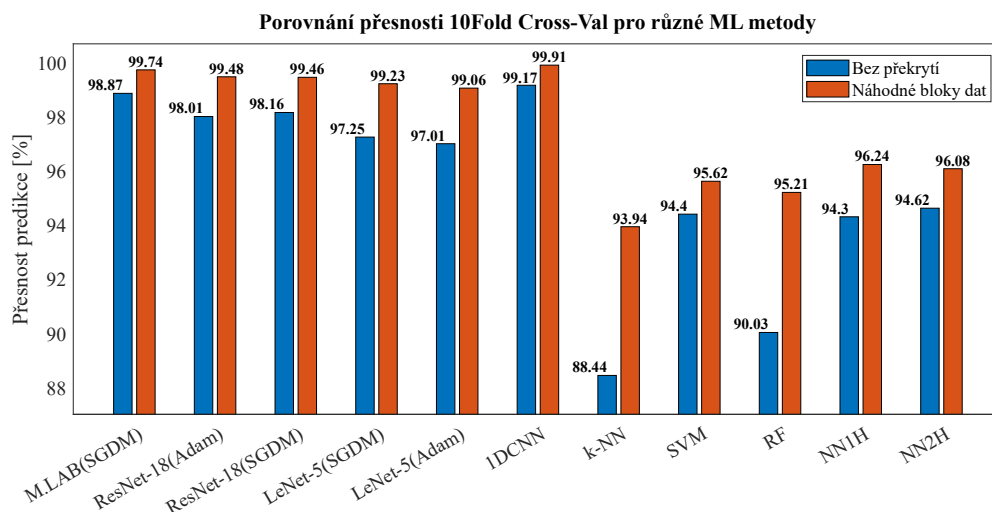


**Obr. 5.27:** Odolnost CNN architektur na přidaný šum, náhodně zvolené bloky dat

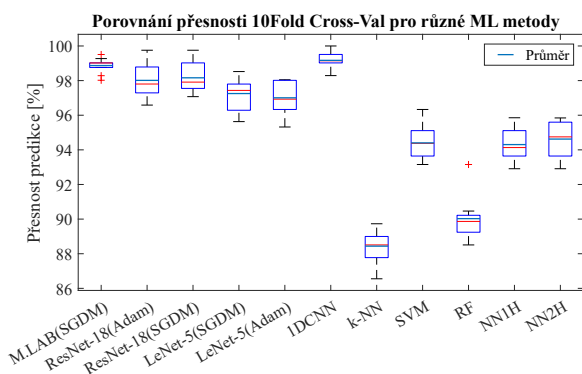
Závěrem se tedy dá říci, že v případě velmi zašuměných dat jsou vhodnější metody, které mají jako vstup statistické parametry signálu než CNN, které extrahují příznaky ze surových dat.

## 5.10 Srovnání klasických metod a CNN

V grafu 5.28 je zobrazena výsledná přesnost predikce pro jednotlivé metody strojového učení s vytvořeným datasetem bez překrývání vzorků (modré sloupce) a s možným překrýváním vzorků mezi pozorováními (červené sloupce). Uvedená přesnost je vždy průměrem přes 10 průběhů křížové validace. Standardní metody strojového učení mají parametry nastaveny na optimální hodnoty, viz sekce 5.3.1, CNN mají parametr kroku učení nalezený s pomocí aplikace *Experiment Manager*, viz 5.7.



**Obr. 5.28:** Porovnání přesnosti predikce jednotlivých algoritmů pomocí 10Fold C-V



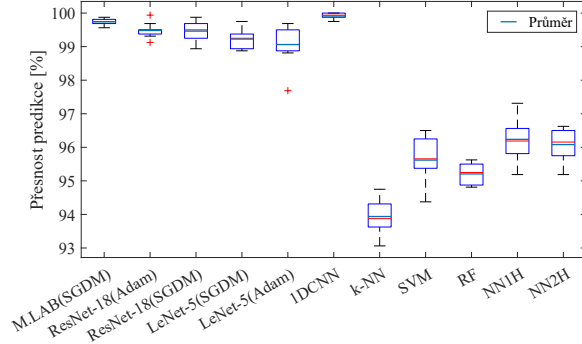
Metoda [-]	Mean [%]	St.D [%]
MATLAB CNN (SGDM)	98,869	0,441
ResNet-18 (Adam)	98,012	0,989
ResNet-18 (SGDM)	98,161	0,913
LeNet-5 (SGDM)	97,252	0,993
LeNet-5 (Adam)	97,006	0,940
IDCNN (SGDM)	99,169	0,567
k-NN	88,438	0,973
SVM	94,402	0,955
RF	90,027	1,256
NN1	94,304	0,932
NN2	94,622	1,038

**Obr. 5.29:** Srovnání přesnosti predikce pomocí 10Fold C-V

V grafech 5.29 a 5.30 jsou přesnosti zobrazeny ve formě krabicového grafu (box-plot). Tento typ grafu obsahuje více informací o testování, konkrétně medián, průměr, maximální a minimální hodnoty, 75. a 25. percentil a odlehlé hodnoty (outlier) pro jednotlivé algoritmy.

Porovnání algoritmů lze provést také na základě výpočetní náročnosti, viz grafy 5.31 pro standardní metody a 5.32 pro CNN. Časový údaj na ose y je v sekundách,

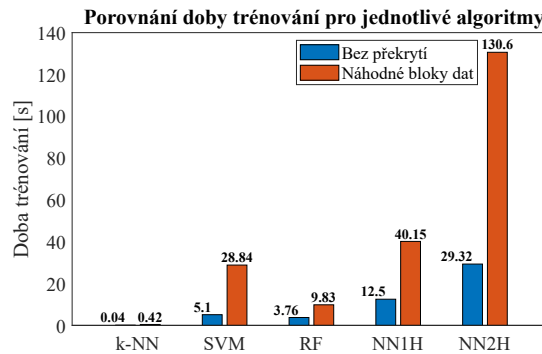
Porovnání přesnosti 10Fold Cross-Val pro různé ML metody, blok dat náhodně



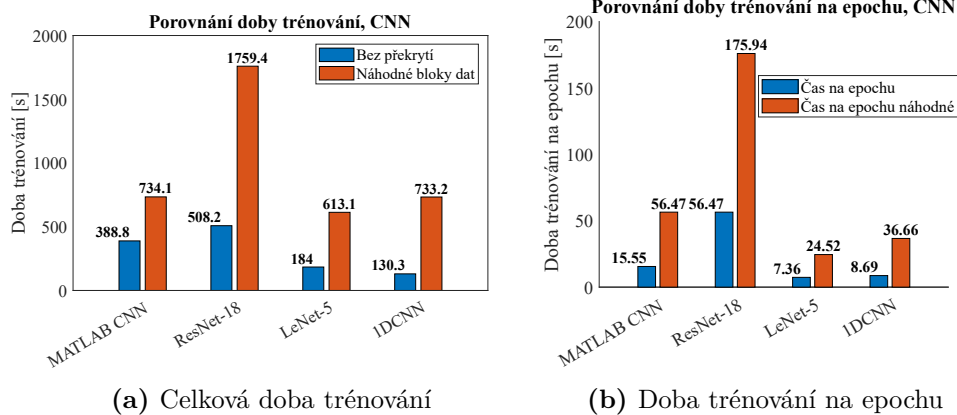
Metoda [-]	Mean [%]	St.D [%]
MATLAB CNN (SGDM)	99,738	0,105
ResNet-18 (Adam)	99,481	0,219
ResNet-18 (SGDM)	99,463	0,291
LeNet-5 (SGDM)	99,225	0,315
LeNet-5 (Adam)	99,063	0,575
1DCNN (SGDM)	99,913	0,094
k-NN	93,938	0,488
SVM	95,619	0,703
RF	95,206	0,316
NN1	96,238	0,624
NN2	96,081	0,472

Obr. 5.30: Srovnání přesnosti predikce pomocí 10Fold C-V, náhodně vybrané bloky

hodnoty jsou zjištěny pro MATLAB 2022a a procesor AMD Ryzen 5 5500U (6/12 jader/vláken) @2,1 GHz (4,0 GHz Turbo). Z konvolučních neuronových sítí z grafu 5.32 je nejnáročnější síť ResNet-18, což bylo očekávané, protože má o dva řády více parametrů než architektura LeNet-5 ( $11,2 \cdot 10^6$  vs  $14,8 \cdot 10^4$ ).



Obr. 5.31: Časové porovnání pro standardní algoritmy strojového učení



Obr. 5.32: Celková doba trénování a doba trénování na epochu pro CNN

		Matice záměn pro 1DCNN, 90% trénovací množina										
Predikovaná třída	Healthy	45	0	1	0	0	0	0	0	0	0	97.8%
		11.0%	0.0%	0.2%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	2.2%
	Ball7	0	35	0	0	0	0	0	0	0	0	100%
		0.0%	8.6%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
	Ball14	0	0	38	0	0	0	0	0	0	0	100%
		0.0%	0.0%	9.3%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
	Ball21	0	0	0	47	0	0	0	0	0	0	100%
		0.0%	0.0%	0.0%	11.5%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
	Inner7	0	0	0	0	39	0	0	0	0	0	100%
		0.0%	0.0%	0.0%	0.0%	9.5%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
Inner14	0	0	0	1	0	35	0	0	0	0	97.2%	
	0.0%	0.0%	0.0%	0.2%	0.0%	8.6%	0.0%	0.0%	0.0%	0.0%	2.8%	
Inner21	0	0	0	0	0	0	32	0	0	0	100%	
	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	7.8%	0.0%	0.0%	0.0%	0.0%	
Outer7	0	0	0	0	0	0	0	36	0	0	100%	
	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	8.8%	0.0%	0.0%	0.0%	
Outer14	0	1	1	0	0	0	0	0	46	0	95.8%	
	0.0%	0.2%	0.2%	0.0%	0.0%	0.0%	0.0%	0.0%	11.2%	0.0%	4.2%	
Outer21	0	0	0	0	0	0	0	0	0	52	100%	
	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	12.7%	0.0%	
	100%	97.2%	95.0%	97.9%	100%	100%	100%	100%	100%	100%	99.0%	
	0.0%	2.8%	5.0%	2.1%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	1.0%	
	Healthy	Ball7	Ball14	Ball21	Inner7	Inner14	Inner21	Outer7	Outer14	Outer21		
	Správná třída											

(a) Matice záměn pro 1DCNN

(b) Matice záměn pro k-NN

Obr. 5.33: Srovnání matice záměn pro 1DCNN a k-NN algoritmus

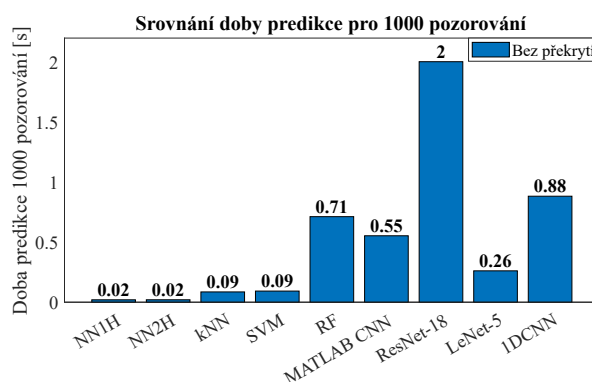
Pro skutečně objektivní porovnání nestačí pouze srovnat přesnost predikce přes všechny třídy, tak jak je uvedena v grafu 5.28. V určitých případech je důležité zjistit, která třída je často chybně klasifikována, jelikož u některých aplikací může být cena chybné klasifikace pro různé třídy různá. V případě ložisek je například zřejmě horší, když algoritmus predikuje pozorování jako bez poškození, i když je ložisko poškozené. Příklad, kdy dojde k záměně poškozené komponenty nebo míry poškození, není až tak nebezpečný. V reálném provozu totiž až tak nesejde na tom, která konkrétní komponenta je poškozena, stejně se ložisko musí vyměnit. Pokud je ovšem ložisko klasifikováno jako zdravé i přes existující poškození, jedná se o bezpečnostní riziko. K odhalení těchto rizik slouží matice záměn, která porovnává skutečnou třídu pozorování s třídou predikovanou algoritmem. Pro každou třídu je tak zvlášť dostupné procento chybné klasifikace.

V grafu 5.33 je srovnání matice záměn pro jednoduchou metodu k-NN a pro 1DCNN konvoluční síť. Na posledním řádku matice záměn je uvedena přesnost/-chybná klasifikace pro každou jednotlivou třídu. Je patrné, že míra chybné klasifikace je pro 1DCNN velmi nízká (1 %) přes všechny třídy, nejvyšší hodnoty (5,0 %) dosahuje pro třídu *Ball14* – poškození kuličky. Pro tuto třídu dojde jednou k záměně za poškození *Outer14* a jednou k záměně za třídu *Healthy* – zdravé ložisko.

Matice záměn pro k-NN je podstatně horší. Přes všechny třídy je míra chybné klasifikace 10 %, nejčastěji dochází k chybné klasifikaci třídy *Ball7*. Zde je hodnota chybné klasifikace 37,1 %, nejčastěji dochází k záměně třídy *Ball7* za třídu *Outer14*. Z matice záměn tak vyplývá, že metoda k-NN buďto nedokáže ze své podstaty

rozlišit tyto dvě třídy, nebo jsou ji poskytnuty špatné příznaky, které neobsahují informaci potřebnou k rozlišení těchto tříd. Vzhledem k tomu, že k záměně tříd *Outer14* a *Ball7*, případně naopak dochází i u všech ostatních standardních metod strojového učení, dá se předpokládat, že na vině je chybná selekce příznaků. Pro rozlišení těchto tříd by tedy bylo nutné přidat jeden či více příznaků, které by umožnily jejich separaci.

V grafu 5.34 je porovnání doby predikce na 1000 pozorování. Jedná se o průměr přes 10 průběhů, při každém průběhu predikce bylo predikováno cca 4000 vzorků najednou. Celková doba predikce byla vydělena počtem pozorování (v tomto případě například cca 4000), pro přehlednost je v grafu udávána na 1000 pozorování.



**Obr. 5.34:** Doba predikce 1000 pozorování

Údaje v grafu 5.34 jsou důležitější než údaje v grafech 5.31–5.32 výše pro celkovou dobu trénování. Trénovat totiž lze na výkonných zařízeních, predikce poté probíhá už s využitím vytrénovaného modelu. Trénování k-NN se podle grafu 5.31 pro dobu trénování je jednoznačně nejkratší. V grafu 5.34 pro dobu predikce je ovšem pomalejší než neuronové sítě. To je dáno podstatou metody, jelikož k-NN je zástupcem líného učení (lazy learning), samotný algoritmus modelu se tedy počítá až při predikci. Zajímavé také je, že například pro CNN architekturu LeNet-5 je predikce rychlejší než s využitím standardního algoritmu RF. Když se porovnají přesnosti predikce pro tyto dva modely, LeNet-5 dosahuje přesnosti predikce pro možnost bloků dat bez překrytí 97,25 %, algoritmus RF přesnosti 90,03 %. Z pohledu výpočetní náročnosti predikce i z pohledu přesnosti je tedy LeNet-5 jednoznačně lepší volbou.

# Závěr

V této práci jsou teoreticky popsány a implementovány metody na diagnostiku valivých ložisek. Jedná se například o metodu obálkové analýzy a také o metody využívající vyhodnocení statistických parametrů, jako je například špičatost, RMS zrychlení a crest faktor.

V další části jsou popsány standardní algoritmy strojového učení a konvoluční neuronové sítě. Tyto algoritmy jsou v praktické části práce použity ke klasifikaci místa a závady na ložiskách.

V první polovině praktické části práce je popsáno měření vlastních dat v laboratoři. Byla poškozena dvě ložiska, první pomocí deformace a koroze, druhé mělo vyfrézované poškození na vnější dráze. Tato dvě ložiska byla poté provozována na přípravku a byly na nich po deseti minutách periodicky měřeny vibrace. První ložisko bylo provozováno cca 39 hodin, druhé cca 9,5 hodin. Byla vytvořena aplikace v MATLABu, ve které je možné načíst .csv soubory s vibracemi z akcelerometru. Aplikace usnadňuje prvotní inspekci dat a také nastavení parametrů metod pro diagnostiku ložisek. U vybraných metod lze vypočítat a zobrazit vývoj hodnot v čase. Z časového vývoje hodnot jednotlivých parametrů byla obě ložiska vyhodnocena jako poškozená. Statistické indikátory špičatost a RMS zrychlení v čase rostou, stejně tak roste pro obě ložiska i efektivní hodnota obálky. Časový vývoj indikátoru crest faktor není průkazný – není zřejmý jasný trend hodnot s časem. Z obálkového spektra lze vyvodit zahlazení závady na vnějším kroužku a postupný rozvoj poškození na kleci.

V druhé polovině praktické části práce jsou ověřeny metody strojového učení na klasifikaci velikosti a místa závady na ložiscích. Jsou použita data z veřejně přístupné databáze Case Western Reserve University. Vzhledem k malému počtu dat byl vibrační signál z ložisek rozdělen na bloky dvěma způsoby. První způsob byl vybírat bloky vzorků ze signálu bez opakování. Druhý způsob byl vybírat bloky vzorků s náhodným počátkem ve vibračním signálu. Pro druhý způsob se některé vzorky v blocích mohou opakovat. Díky tomu byl ale značně zvýšen počet pozorování v testovací a trénovací množině. Pro konvoluční síť s 2D vstupem byly z bloků dat vytvořeny vibrační obrázky. V následujícím textu je popisován výsledek první metody výběru dat, tedy bloky bez opakování vzorků. U této metody je zajištěno, že vzorky z trénovací množiny se neobjeví v testovací množině, zároveň dojde kvůli tomu ke značnému zmenšení počtu pozorování. Výsledkem je tedy menší přesnost predikce pro všechny metody, viz graf srovnání 5.28.

Aby byla přesnost predikce co největší, byly pro jednotlivé metody strojového učení nalezeny optimální parametry. U standardních metod strojového učení byly optimální parametry ověřeny pomocí křížové validace. Nejvyšší přesnosti z klasic-



kých metod strojového učení dosáhly neuronové sítě a algoritmus SVM, přesnost predikce byla pro tyto modely větší než 94,4 %. Z konvolučních neuronových sítí je nejlepší architektura 1DCNN, ta dosáhla přesnosti 99,17 %. Tyto výsledky byly dosaženy pro 90 % dat v trénovací množině a 10 % dat v testovací množině.

Dále je pro metody ověřen vliv zmenšení velikosti trénovací množiny na přesnost predikce. Z klasických metod strojového učení se jako nejvhodnější algoritmus pro malé množství dat ukázal SVM. Z metod konvolučních neuronových sítí se jako nejlepší ukázala architektura MATLAB CNN, druhou nejlepší byla architektura 1DCNN. Poté byla také otestována odolnost jednotlivých metod na přidaný šum. Největší odolnost proti přidanému šumu mají klasické metody strojového učení, konkrétně Random Forest a neuronová síť s dvěma skrytými vrstvami. Pro zašuměná data se naopak jako nevhodné prokázaly konvoluční neuronové sítě, pravděpodobně kvůli způsobu transformace surového 1D signálu z akcelerometru na vibrační obrázek.

V případě, že máme k dispozici malé množství trénovacích dat je tedy vhodné použít algoritmus SVM nebo architekturu konvoluční neuronové sítě, v práci nazvanou jako MATLAB CNN. Pokud víme, že jsou data velmi zašuměná, je vhodné použít Random Forest, případně neuronovou síť s dvěma skrytými vrstvami. Na zašuměná data se nehodí konvoluční neuronové sítě v případě, že jako vstup jim poskytneme surová data ve formě vibračního obrázku. Když se vezme v potaz i doba na výpočet predikce, jako nejuniverzálnější se jeví algoritmus SVM.

## Literatura

- [1] BILOŠ, Jan a Alena BILOŠOVÁ. *Aplikovaný mechanik jako součást týmů konstruktérů a vývojářů: studijní opora*. Ostrava: Vysoká škola báňská - Technická univerzita Ostrava, Fakulta strojní, 2012. ISBN 978-80-248-2755-1.
- [2] SADEGHI, Farshid, Behrooz JALALAHMADI, Trevor S. SLACK, Nihar RAJE a Nagaraj K. ARAKERE. A Review of Rolling Contact Fatigue. *Journal of Tribology* [online]. 2009, 131(4), 1-15 [cit. 2021-10-18]. ISSN 0742-4787. Dostupné z: doi:10.1115/1.3209132
- [3] SKF Group. *Bearing damage and failure analysis* [online dokument]. ©2017 [cit. 2021-10-18]. PUB BU/I3 14219/2 EN. Dostupné z: <[https://www.skf.com/binaries/pub12/Images/0901d1968064c148-Bearing-failures---14219\\_2-EN\\_tcm\\_12-297619.pdf](https://www.skf.com/binaries/pub12/Images/0901d1968064c148-Bearing-failures---14219_2-EN_tcm_12-297619.pdf)>.
- [4] DWYER-JOYCE, R.S. Predicting the abrasive wear of ball bearings by lubricant debris. *Wear* [online]. 1999, 233-235, 692-701 [cit. 2021-10-18]. ISSN 00431648. Dostupné z: doi:10.1016/S0043-1648(99)00184-2
- [5] ADAMS, Maurice L. *Bearings: basic concepts and design applications*. Boca Raton: CRC Press, Taylor & Francis Group, [2018]. ISBN 9781315169798.
- [6] ČSN ISO 281. *Valivá ložiska – Dynamická únosnost a trvanlivost* Praha: Úřad pro technickou normalizaci, metrologii a státní zkušebnictví, 2008, 52 s. Třídící znak 02 4607.
- [7] ČSN ISO 76. *Valivá ložiska – Statická únosnost* Praha: Úřad pro technickou normalizaci, metrologii a státní zkušebnictví, 2008, 20 s. Třídící znak 02 4610.
- [8] SKF Bearing Select *SKF Group* [online]. Göteborg: AB SKF, bez data vydání [cit. 2022-05-07]. Dostupné z: <<https://www.skfbearingselect.com/>>.
- [9] SKF Group. *SKF Vibration Sensors Catalog* [online dokument]. ©2018 [cit. 2021-12-31]. PUB CM/P1 11604/16 EN. Dostupné z:<[https://www.skf.com/binaries/pub12/Images/0901d196804926fe-11604\\_16-EN-Vibration-Sensor-Catalogue---0K\\_tcm\\_12-267858.pdf](https://www.skf.com/binaries/pub12/Images/0901d196804926fe-11604_16-EN-Vibration-Sensor-Catalogue---0K_tcm_12-267858.pdf)>.
- [10] MCFADDEN, P.D. a J.D. SMITH. The vibration produced by multiple point defects in a rolling element bearing. *Journal of Sound and Vibration* [online]. 1985, 98(2), 263-273 [cit. 2021-10-20]. ISSN 0022460X. Dostupné z: doi:10.1016/0022-460X(85)90390-6

- [11] KONSTANTIN-HANSEN, H a H. HERLUFSEN, Envelope and cepstrum analyses for machinery fault identification. *Sound & vibration* [online]. 2010, 10-12. ISSN 1541-0161. Dostupné z: <<http://www.sandv.com/downloads/1005hans.pdf>>.
- [12] SU, Y.-T. a S.-J. LIN. On initial fault detection of a tapered roller bearing: Frequency domain analysis. *Journal of Sound and Vibration* [online]. 1992, 155(1), 75-84 [cit. 2021-10-20]. ISSN 0022460X. Dostupné z: doi:10.1016/0022-460X(92)90646-F
- [13] Case Western Reserve University, *Bearing Data Center* [online]. bez data vydání [cit. 2021-10-24]. Dostupné z: <<https://engineering.case.edu/bearingdatacenter/download-data-file>>.
- [14] ČSN ISO 10816-3. *Vibrace – Hodnocení vibrací strojů na základě měření na nerotujících částech – Část 3: Průmyslové stroje se jmenovitým výkonem nad 15 kW a jmenovitými otáčkami mezi 120 1/min a 15 000 1/min při měření in situ* Praha: Úřad pro technickou normalizaci, metrologii a státní zkušebnictví, 2010, 20 s. Třídící znak 01 1412.
- [15] WILLIAMS, T., X. RIBADENEIRA, S. BILLINGTON a T. KURFESS. ROLLING ELEMENT BEARING DIAGNOSTICS IN RUN-TO-FAILURE LIFETIME TESTING. *Mechanical Systems and Signal Processing* [online]. 2001, 15(5), 979-993 [cit. 2021-11-09]. ISSN 08883270. Dostupné z: doi:10.1006/mssp.2001.1418
- [16] HU, Aijun, Ling XIANG a Lijia ZHU. An engineering condition indicator for condition monitoring of wind turbine bearings. *Wind Energy* [online]. 2020, 23(2), 207-219 [cit. 2021-11-06]. ISSN 1095-4244. Dostupné z: doi:10.1002/we.2423
- [17] DE ALMEIDA, Rui Gomes Teixeira, Silmara Alexandra DA SILVA VICENTE a Linilson Rodrigues PADOVESE. New Technique for Evaluation of Global Vibration Levels in Rolling Bearings. *Shock and Vibration* [online]. 2002, 9(4-5), 225-234 [cit. 2021-11-09]. ISSN 1070-9622. Dostupné z: doi:10.1155/2002/647652
- [18] BARON, Petr, Marek KOČIŠKO, Lukáš BLAŠKO a Patrik SZENTIVANYI. Verification of the operating condition of stationary industrial gearbox through analysis of dynamic signal, measured on the pinion bearing housing. *Measurement* [online]. 2017, 96, 24-33 [cit. 2021-11-06]. ISSN 02632241. Dostupné z: doi:10.1016/j.measurement.2016.10.048

- [19] Brüel & Kjær Vibro A/S. *Detecting Faulty Rolling Element Bearings* [online application note]. bez data vydání, [cit. 2021-11-06]. Dostupné z: <[https://www.bkvbvibro.com/fileadmin/mediapool/Internet/Application\\_Notes/detecting\\_faulty\\_rolling\\_element\\_bearings.pdf](https://www.bkvbvibro.com/fileadmin/mediapool/Internet/Application_Notes/detecting_faulty_rolling_element_bearings.pdf)>.
- [20] MCFADDEN, P.D. a J.D. SMITH. Vibration monitoring of rolling element bearings by the high-frequency resonance technique — a review. *Tribology International* [online]. 1984, 17(1), 3-10 [cit. 2021-10-23]. ISSN 0301679X. Dostupné z: doi:10.1016/0301-679X(84)90076-8
- [21] YAN, Ruqiang a Robert X. GAO. Multi-scale enveloping spectrogram for vibration analysis in bearing defect diagnosis. *Tribology International* [online]. 2009, 42(2), 293-302 [cit. 2021-10-31]. ISSN 0301679X. Dostupné z: doi:10.1016/j.triboint.2008.06.013
- [22] KUYUK, H SERDAR. On the use of Stockwell transform in structural dynamic analysis. *Sadhana* [online]. 2015, 40(1), 295-306 [cit. 2021-11-07]. ISSN 0256-2499. Dostupné z: doi:10.1007/s12046-014-0301-2
- [23] SCHOLL, Stefan. *Fourier, Gabor, Morlet or Wigner: Comparison of Time-Frequency Transforms* [online]. 17.01.2021, 1-10 [cit. 2021-11-07]. Dostupné z: <<https://arxiv.org/pdf/2101.06707.pdf>>.
- [24] MALLAT, S.G. A theory for multiresolution signal decomposition: the wavelet representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* [online]. 1989, 11(7), 674-693 [cit. 2021-11-16]. ISSN 01628828. Dostupné z: doi:10.1109/34.192463
- [25] 1-D Decimated Wavelet Transforms. *The MathWorks, Inc* [online]. Natick (Massachusetts), bez data vydání [cit. 2021-11-15]. Dostupné z: <<https://www.mathworks.com/help/wavelet/ug/one-dimensional-discrete-wavelet-analysis.html>>.
- [26] 1-D wavelet decomposition. *The MathWorks, Inc* [online]. Natick (Massachusetts), bez data vydání, [cit. 2021-11-15]. Dostupné z: <<https://www.mathworks.com/help/wavelet/ref/wavedec.html>>.
- [27] Critically-Sampled Wavelet Reconstruction. *The MathWorks, Inc* [online]. Natick (Massachusetts), bez data vydání, [cit. 2021-11-15]. Dostupné z: <<https://www.mathworks.com/help/wavelet/gs/wavelet-reconstruction.html>>.

- [28] Fast Wavelet Transform (FWT) Algorithm. *The MathWorks, Inc* [online]. Natick (Massachusetts), bez data vydání, [cit. 2021-11-15]. Dostupné z: <<https://www.mathworks.com/help/wavelet/ug/fast-wavelet-transform-fwt-algorithm.html>>.
- [29] NIZWAN, C K E, S A ONG, M F M YUSOF a M Z BAHAROM. A wavelet decomposition analysis of vibration signal for bearing fault detection. *IOP Conference Series: Materials Science and Engineering* [online]. 2013, 50, 1-10 [cit. 2021-11-15]. ISSN 1757-8981. Dostupné z: doi:10.1088/1757-8981/50/1/012026
- [30] ALPAYDIN, Ethem. *Introduction to machine learning*. 3rd ed. Cambridge (Massachusetts): The MIT Press, 2014. ISBN 978-0-262-02818-9.
- [31] HONZÍK, Petr. *Strojové učení*. Brno, 2006, 85 s.
- [32] GÉRON, Aurélien. *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: concepts, tools, and techniques to build intelligent systems*. Second edition. Beijing: O'Reilly, 2019. ISBN 978-1-492-03264-9.
- [33] Support Vector Machine (SVM) *The MathWorks, Inc* [online]. Natick (Massachusetts), bez data vydání, [cit. 2022-05-07]. Dostupné z: <<https://www.mathworks.com/discovery/support-vector-machine.html>>.
- [34] Fit multiclass models for support vector machines or other classifiers (fitcecoc) *The MathWorks, Inc* [online]. Natick (Massachusetts), bez data vydání, [cit. 2022-05-07]. Dostupné z: <<https://www.mathworks.com/help/stats/fitcecoc.html>>.
- [35] BREIMAN, Leo. Random Forests. *Machine Learning* [online]. 2001, 45(1), 5-32 [cit. 2022-05-07]. ISSN 08856125. Dostupné z: doi:10.1023/A:1010933404324
- [36] RUMELHART, David E., Geoffrey E. HINTON a Ronald J. WILLIAMS. Learning representations by back-propagating errors. *Nature* [online]. 1986, 323(6088), 533-536 [cit. 2022-04-30]. ISSN 0028-0836. Dostupné z: doi:10.1038/323533a0
- [37] DOSEDĚL, Martin. *Vibrodiagnostics of the machines* [prezentace]. MPC-MTD: Měřicí technika pro diagnostiku 2020/2021. [cit. 2021-12-27]
- [38] 6206 ETN9 - Deep groove ball bearings. *SKF Group* [online]. Göteborg: AB SKF, bez data vydání [cit. 2021-12-27]. Dostupné z: <<https://www.skf.com/uk/products/rolling-bearings/ball-bearings/deep-groove-ball-bearings/productid-6206%20ETN9>>.

- [39] BK Connect - Sound and Vibration Software. *BRÜEL & KJÆR* [online]. Göteborg: HOTTINGER BRÜEL & KJÆR A/S, bez data vydání, [cit. 2021-12-27]. Dostupné z: <<https://www.bksv.com/en/analysis-software/data-acquisition-software/bk-connect>>.
- [40] Generator Module | LAN-Xi 3160. *BRÜEL & KJÆR* [online]. Göteborg: HOTTINGER BRÜEL & KJÆR A/S, bez data vydání, [cit. 2021-12-27]. Dostupné z: <<https://www.bksv.com/en/instruments/daq-data-acquisition/lan-xi-daq-system/daq-modules/type-3160>>.
- [41] Case Western Reserve University, *Bearing Data Center* [online]. bez data vydání [cit. 2022-04-10]. Dostupné z: <<https://engineering.case.edu/bearingdatacenter/apparatus-and-procedures>>.
- [42] Wind Turbine High-Speed Bearing Prognosis. *The MathWorks, Inc* [online]. Natick (Massachusetts), bez data vydání, [cit. 2022-04-20]. Dostupné z: <<https://www.mathworks.com/help/predmaint/ug/wind-turbine-high-speed-bearing-prognosis.html>>.
- [43] Deep Learning Using Bayesian Optimization. *The MathWorks, Inc* [online]. Natick (Massachusetts), bez data vydání, [cit. 2022-04-20]. Dostupné z: <<https://www.mathworks.com/help/deeplearning/ug/deep-learning-using-bayesian-optimization.html>>.
- [44] WAN, Lanjun, Yiwei CHEN, Hongyang LI a Changyun LI. Rolling-Element Bearing Fault Diagnosis Using Improved LeNet-5 Network. *Sensors* [online]. 2020, 20(6), 1-23 [cit. 2022-04-20]. ISSN 1424-8220. Dostupné z: doi:10.3390/s20061693
- [45] LECUN, Y., L. BOTTOU, Y. BENGIO a P. HAFFNER. Gradient-based learning applied to document recognition. *Proceedings of the IEEE* [online]. 1998, 86(11), 2278-2324 [cit. 2022-04-20]. ISSN 00189219. Dostupné z: doi:10.1109/5.726791
- [46] HE, Kaiming, Xiangyu ZHANG, Shaoqing REN a Jian SUN. Deep Residual Learning for Image Recognition. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* [online]. Las Vegas: IEEE, 2016, 2016, s. 770-778 [cit. 2022-05-05]. ISBN 978-1-4673-8851-1. ISSN 1063-6919. Dostupné z: doi:10.1109/CVPR.2016.90

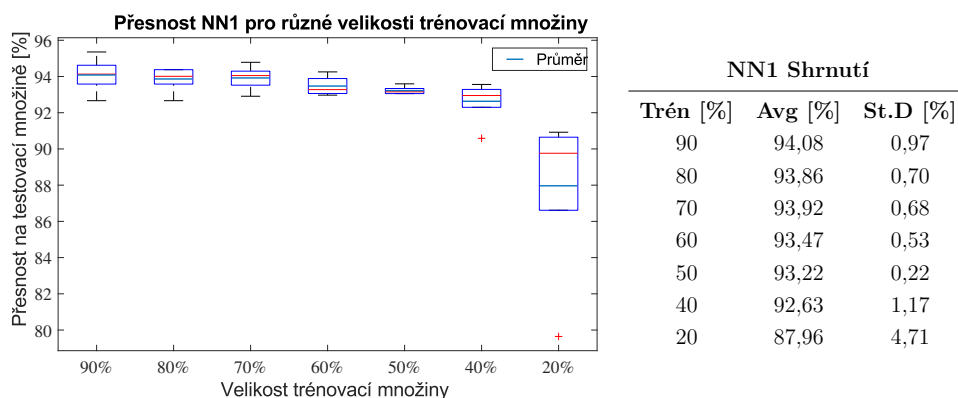
## Seznam symbolů a zkratek

$L_{10}$	základní stanovená trvanlivost ložiska [ $10^6$ RPM]
$C_r$	základní radiální dynamická únosnost [N]
$P_r$	ekvivalent dynamického radiálního zatížení [N]
$L_{10h}$	základní stanovená trvanlivost ložiska v hodinách [hod]
<i>RPM</i>	Revolutions Per Minute – otáčky za minutu [1/min]
$b_m$	koeficient pro ocel závislý na typu a konstrukci ložiska pro výpočet $C_r$
$f_c$	koeficient daný geometrií ložiska pro výpočet $C_r$
$i$	počet řad valivých elementů pro výpočet $C_r$ [-]
$Z$	počet valivých elementů pro výpočet $C_r$ [-]
$D_w$	průměr kuličky pro výpočet $C_r$ [mm]
$X$	koeficient radiálního dynamického zatížení pro výpočet $P_r$ [-]
$Y$	koeficient axiálního dynamického zatížení pro výpočet $P_r$ [-]
$F_r$	radiální složka skutečného zatížení ložiska pro výpočet $P_r$ [N]
$F_a$	axiální složka skutečného zatížení ložiska pro výpočet $P_r$ [N]
$C_{0r}$	základní radiální statická únosnost [N]
$L_{mn}$	modifikovaná trvanlivost [ $10^6$ RPM]
$a_1$	koeficient modifikované trvanlivosti pro spolehlivost
$a_{iso}$	koeficient modifikované trvanlivosti z normy ISO281
<b>BPMI</b>	Ball Pass Frequency Inner Race – frekvence pro závadu na vnitřním kroužku [Hz]
<b>BPFO</b>	Ball Pass Frequency Inner Race – frekvence pro závadu na vnějším kroužku [Hz]
<b>BSF</b>	Ball Spin Frequency – frekvence pro závadu na valivém elementu [Hz]
<b>FTF</b>	Fundamental Train Frequency – frekvence pro závadu na kleci [Hz]

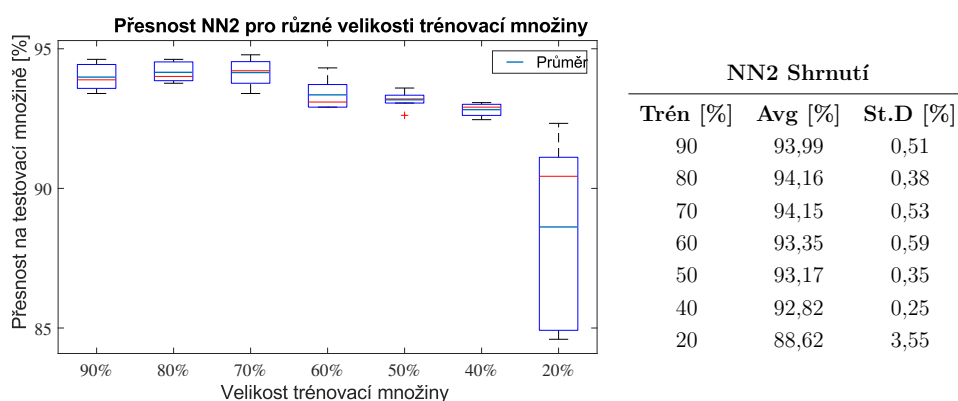
$f_s$	frekvence otáčení hřídele [Hz]
$N$	počet valivých elementů ložiska [-]
$B_d$	průměr valivého elementu [mm]
$P_d$	roztečný průměr [mm]
$\varphi$	kontaktní úhel [°]
$T$	perioda [s]
<b>RMS</b>	Root Mean Square – efektivní hodnota
$P_{jmen}$	jmenovitý výkon [W]
$K$	Kurtosis – špičatost
$\sigma$	směrodatná odchylka
$c_f$	Crest faktor
$x_{peak}$	Špičková hodnota signálu
$\bar{x}$	průměrná hodnota
$a_s$	analytický signál
<b>FWT</b>	Fast Wavelet Transform – rychlá vlnková transformace
<b>STU</b>	Strojové učení
<b>k-NN</b>	k-Nearest Neighbors – algoritmus strojového učení
<b>SVM</b>	Support-Vector Machines – algoritmus strojového učení
<b>RF</b>	Random Forest – ensemble algoritmus strojového učení
<b>MSE</b>	Mean Square Error – střední kvadratická chyba
<b>GD</b>	Gradient Descent – gradientní sestup
<b>MBGD</b>	Mini-Batch Gradient Descent – gradientní sestup přes dávky
<b>CNN</b>	Convolutional Neural Network – konvoluční neuronová síť
<b>FC</b>	Fully Connected Layer – plně propojená vrstva u CNN



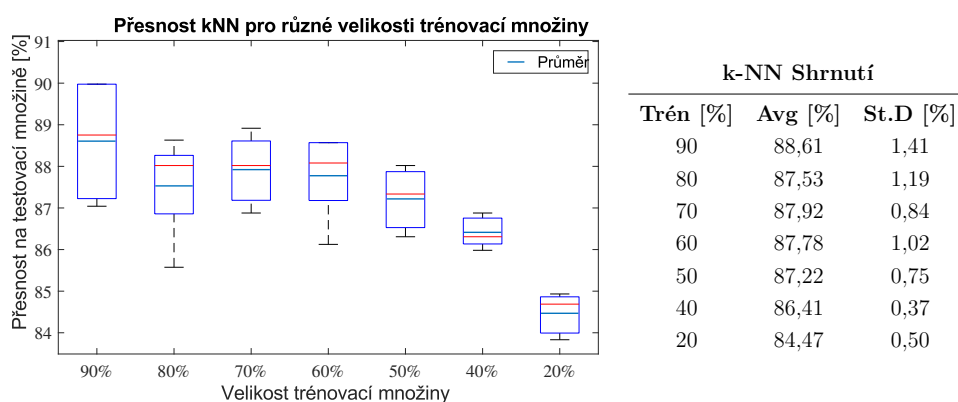
# A Grafické znázornění závislosti přesnosti predikce na velikosti trénovací množiny



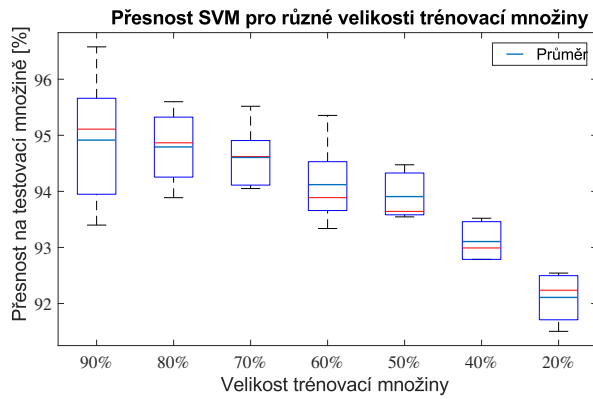
Obr. A.1: Vliv velikosti trénovací množiny na přesnost predikce, NN1



Obr. A.2: Vliv velikosti trénovací množiny na přesnost predikce, NN2

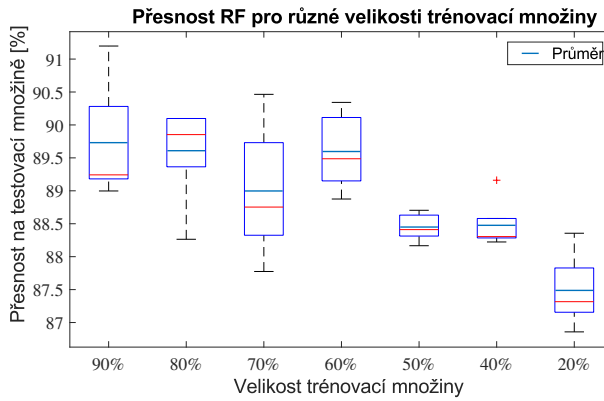


Obr. A.3: Vliv velikosti trénovací množiny na přesnost predikce, k-NN



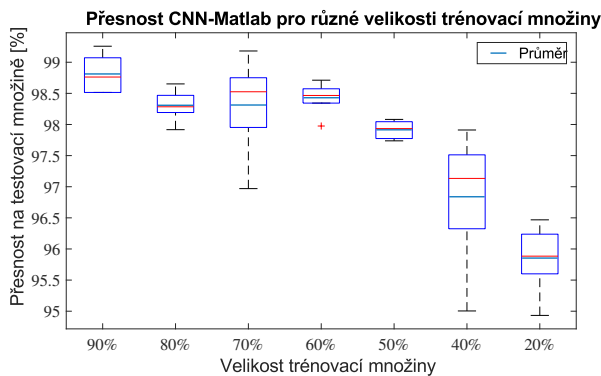
SVM Shrnutí		
Trén [%]	Avg [%]	St.D [%]
90	94,91	1,22
80	94,79	0,68
70	94,60	0,59
60	94,12	0,76
50	93,91	0,44
40	93,11	0,35
20	92,11	0,45

Obr. A.4: Vliv velikosti trénovací množiny na přesnost predikce, SVM



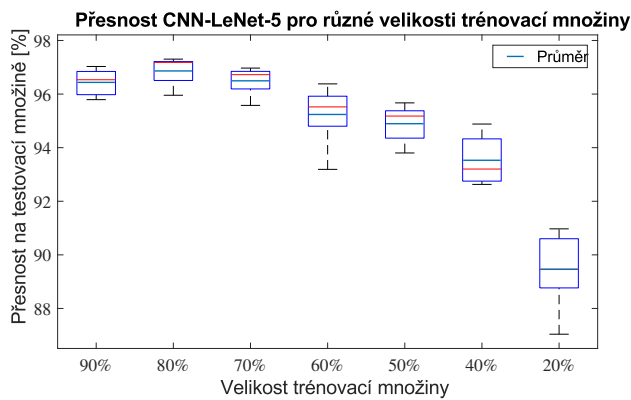
RF Shrnutí		
Trén [%]	Avg [%]	St.D [%]
90	89,73	0,90
80	89,61	0,77
70	89,00	1,02
60	89,60	0,59
50	88,45	0,21
40	88,48	0,39
20	87,49	0,56

Obr. A.5: Vliv velikosti trénovací množiny na přesnost predikce, RF



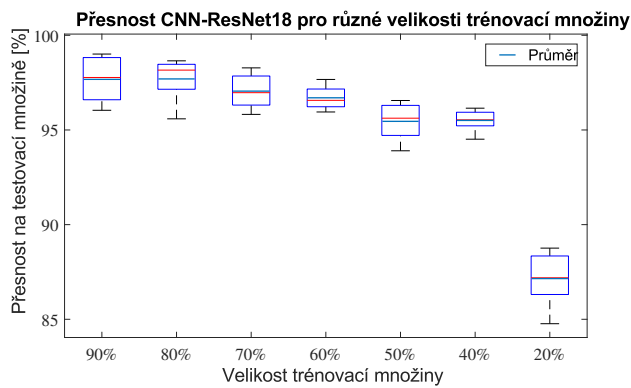
MATLAB CNN Shrnutí		
Trén [%]	Avg [%]	St.D [%]
90	98,81	0,32
80	98,31	0,27
70	98,31	0,82
60	98,43	0,27
50	97,91	0,15
40	96,84	1,11
20	95,85	0,57

Obr. A.6: Vliv velikosti trénovací množiny na přesnost predikce, MATLAB CNN



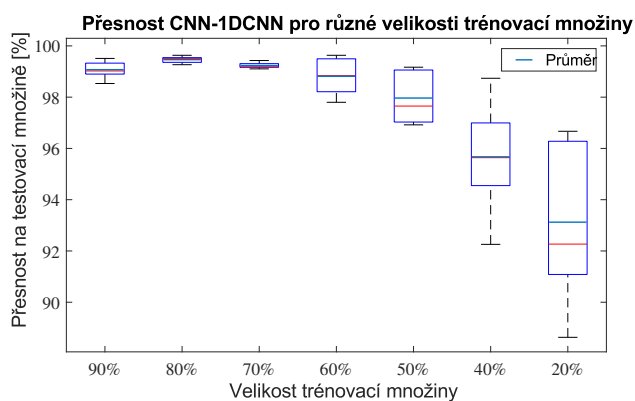
LeNet-5 Shrnutí		
Trén [%]	Avg [%]	St.D [%]
90	96,44	0,51
80	96,86	0,56
70	96,49	0,55
60	95,24	1,21
50	94,89	0,73
40	93,53	0,96
20	89,46	1,52

**Obr. A.7:** Vliv velikosti trénovací množiny na přesnost predikce, LeNet-5 CNN



ResNet-18 Shrnutí		
Trén [%]	Avg [%]	St.D [%]
90	97,67	1,27
80	97,70	1,23
70	97,05	0,97
60	96,70	0,66
50	95,45	1,06
40	95,50	0,62
20	87,15	1,54

**Obr. A.8:** Vliv velikosti trénovací množiny na přesnost predikce, ResNet-18 CNN



1DCNN Shrnutí		
Trén [%]	Avg [%]	St.D [%]
90	99,07	0,36
80	99,46	0,14
70	99,23	0,12
60	98,81	0,76
50	97,97	1,07
40	95,67	2,33
20	93,12	3,32

**Obr. A.9:** Vliv velikosti trénovací množiny na přesnost predikce, 1DCNN

## B Obsah přiloženého DVD

Na přiloženém DVD jsou soubory uloženy podle následujícího popisu.

```
/ ..... kořenový adresář přiloženého DVD
├── DP_Martin_Bar_Loziska.pdf ..... text práce
├── Holdout_test ..... test velikosti trénovací množiny, grafy v .svg formátu
├── Krizova_validace ..... křížová validace, složka s grafy v .svg formátu
├── Krok_uceni_test ..... test kroku učení, složka s grafy v .svg formátu
├── Matice_zamen ..... matice záměn pro modely, složka s grafy v .svg formátu
├── Ostatni_grafy ..... složka s ostatními grafy v .svg formátu
├── SNR_test ..... přidání šumu do dat, složka s grafy v .svg formátu
├── Vibracni_obrazky ..... vygenerované vibrační obrázky v .png formátu
│   ├── ball7 ..... obrázky pro třídu ball7
│   ├── ball14 ..... obrázky pro třídu ball14
│   ├── ball21 ..... obrázky pro třídu ball21
│   ├── healthy ..... obrázky pro třídu healthy
│   ├── inner7 ..... obrázky pro třídu inner7
│   ├── inner14 ..... obrázky pro třídu inner14
│   ├── inner21 ..... obrázky pro třídu inner21
│   ├── outer7 ..... obrázky pro třídu outer7
│   ├── outer14 ..... obrázky pro třídu outer14
│   └── outer21 ..... obrázky pro třídu outer21
```