

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

DIPLOMOVÁ PRÁCE

Brno, 2020

Bc. Ondřej Váško



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

MONITOROVÁNÍ STAVU PŘI OBRÁBĚNÍ BLOKU MOTORU - PLAZMA

CONDITION MONITORING OF ENGINE BLOCK MACHINING - PLASMA

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. Ondřej Váško

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Stanislav Pikula, Ph.D.

BRNO 2020

Diplomová práce

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

Ústav automatizace a měřicí techniky

Student: Bc. Ondřej Váško

ID: 186593

Ročník: 2

Akademický rok: 2019/20

NÁZEV TÉMATU:

Monitorování stavu při obrábění bloku motoru - Plazma

POKYNY PRO VYPRACOVÁNÍ:

- 1) Vypracujte literární rešerši metod použitelných pro sledování stavu strojů a prediktivní diagnostiku, kterými bude možné najít souvislosti mezi daty z průběhu výroby a přesným měřením finálního kusu.
- 2) Nastudujte a popište systém diagnostiky při výrobě motorového bloku ve Škoda auto a.s., systém ověřování hotových motorových bloků a sledování stavu výrobních strojů a jejich kondice.
- 3) Proveďte základ analýzy datových bloků z výrobních strojů a dat z měření.
- 4) Vyberte vhodné metody pro analýzu reálných dat, realizujte je a porovnejte jejich výsledky.
- 5) Vyberte nejvhodnější metodu pro predikci výsledku obrábění bloku motoru a zhodnoťte přesnost její predikce.

DOPORUČENÁ LITERATURA:

Rao, B.K.N. ed., 1996. Handbook of condition monitoring. Elsevier.

Termín zadání: 3.2.2020

Termín odevzdání: 1.6.2020

Vedoucí práce: Ing. Stanislav Pikula, Ph.D.

doc. Ing. Václav Jirsík, CSc.
předseda oborové rady

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 této práce je návrh a realizace dvou metod prediktivní analýzy pro firmu Škoda Auto a.s. V první části jsem zpracoval literární rešerši na téma metody prediktivní diagnostiky. V další části je za pomoci konzultanta práce ve Škoda Auto a.s. provedena podrobná analýza výrobní linky a datových bloků z výrobních strojů a dat z měření. Následně jsem navrhl a naprogramoval generátoru dat založeného na skutečných datech. Dále jsem vytvořil dvě metody prediktivní diagnostiky schopné analyzovat vstupní data a na jejich základě rozhodnout o kondici obráběného kusu. Nakonec jsem tyto metody otestoval a zhodnotil přesnost jejich predikce. Hlavním výstupem mé práce jsou dvě metody prediktivní diagnostiky, které lze použít v praxi.

KLÍČOVÁ SLOVA

Prediktivní analýza, diagnostika, sledování stavu, neuronová síť

ABSTRACT

The aim of this thesis is to design and implement two methods of predictive analysis for company Škoda Auto a.s. In the first part I have conducted a literature search on methods of predictive diagnostics. In the next part, with help from thesis consultant in Škoda Auto a.s., the analysis of the assembly line and data blocks from machinery and measuring has been made. Then I designed and programmed data generator based on real data. I created two methods of predictive diagnostics, capable of analyzing input data and deciding about their condition. In the end I tested these two methods and evaluated accuracy of their prediction. Main output of my thesis is two methods of predictive diagnostics, feasible in real world.

KEYWORDS

Predictive analysis, diagnostics, condition monitoring, neural network

VÁŠKO, Ondřej. *Monitorování stavu při obrábění bloku motoru - Plazma*. Brno, 2020, 119 s. Diplomová práce. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav automatizace. Vedoucí práce: Ing. Stanislav Pikula, Ph.D

PROHLÁŠENÍ

Prohlašuji, že svou diplomovou práci na téma „Monitorování stavu při obrábění bloku motoru - Plazma“ jsem vypracoval samostatně pod vedením vedoucího diplomové 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 diplomové práce dále prohlašuji, že v souvislosti s vytvořením této diplomové 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 1.6.2020

.....
podpis autora

PODĚKOVÁNÍ

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

Brno 1.6.2020

.....

podpis autora

Obsah

Úvod	17
1 Teoretická část diplomové práce	19
1.1 Prediktivní analýza	19
1.2 Dolování dat	20
1.3 Kauzální vysvětlení vs. empirická predikce	20
1.3.1 Vysvětlování a vysvětlující modelování (Explanatory Modeling)	21
1.3.2 Prediktivní modelování	21
1.3.3 Deskriptivní modelování	21
1.4 Analytické techniky	22
1.4.1 Regresní techniky	22
1.5 Techniky strojového učení (machine learning)	33
1.5.1 Neuronové sítě	38
1.5.2 Perceptrony	40
1.5.3 Naive Bayes	41
1.5.4 K-nejbližší sousedé	42
1.5.5 Hluboké učení (Deep learning)	44
1.5.6 Neuronová síť typu LSTM	46
1.5.7 Struktura a vysvětlení LSTM sítě	46
1.5.8 Jednotlivé kroky LSTM sítě	48
1.5.9 Popis výsledků neuronové sítě	50
2 Popis výrobní linky a sběru dat	53
2.1 Jednotlivé operace na výrobní lince	53
2.1.1 Rozbor systému diagnostiky	54
2.2 Popis dat linky plazmy	62
2.2.1 Důležitá data z linky plazmy	62
2.2.2 Rozbor datových bloků (RFID)	64
2.2.3 Finální vybraná data pro prediktivní analýzu	64
3 Praktická část diplomové práce	65
3.1 Úvod	65
3.2 Hlavní funkce (main.c)	67
3.2.1 Volba regrese	67
3.2.2 Volba detailního nastavení metody regresní analýzy	67
3.2.3 Volba neuronové sítě	68
3.2.4 Volba detailního nastavení metody neuronové sítě	69

3.3	Metoda regresní analýzy	71
3.3.1	Načítání dat (skript data_loading.m)	71
3.3.2	Předzpracování dat (skript data_preproces.m)	71
3.3.3	Generování dat rychlosti otáček laseru pro metodu regrese (skript data_gen_rpm.m)	72
3.3.4	Generování dat teploty pro metodu regrese (skript data_gen_temp.m)	72
3.3.5	Rozdělení vstupních dat na intervaly (skript data_parsing.m)	72
3.3.6	Metoda regresní analýzy (skript regression3.m)	72
3.3.7	Metoda regresní analýzy (skript regr.m)	75
3.3.8	Tisk výsledku metody (skript data_print.m)	76
3.4	Neuronová síť	77
3.4.1	Skript pro tvorbu datasetů (DNN_data.m)	77
3.4.2	Generátor OK intervalů	79
3.4.3	Zamíchání dat (skript data_shuffler.m)	81
3.4.4	Generátor NOK intervalů	81
3.4.5	Parametry neuronové sítě (skript NN.m)	83
3.4.6	Průběh učení	86
3.4.7	Výsledky metody neuronové sítě (skript NN_results.m)	88
3.5	Porovnání výsledků obou metod	89
	Závěr	91
	Literatura	93
	Seznam symbolů, veličin a zkratk	97
	Seznam příloh	99
	A První příloha	101
A.1	Globální data	101
A.2	Poslední operace	101
A.3	Lokální data	102
	B Druhá příloha	111
B.1	Operace AF110 2x Stroj GROB	111
B.2	Operace AF130 1x Stroj WZ-Bau-Braunschweig - Laseranlage	112
B.3	Operace AF140 2x Stroj WZ-Bau-Braunschweig - Beschichtungsan- lage (Plazmování - potahování)	113

C Program	115
C.1 main.m	115
C.2 data_loading.m	115
C.3 data_preproces.m	115
C.4 data_parsing.m	115
C.5 regression3.m	115
C.6 regr.m	115
C.7 data_gen_rpm.m	115
C.8 data_gen_temp.m	115
C.9 data_shuffler.m	116
C.10 data_print.m	116
C.11 NOK_int2.xlsx	116
C.12 OK_int.xlsx	116
C.13 shuf.xlsx	116
C.14 temp.xlsx	116
C.15 random_stuff.m	116
C.16 DNN_data.m	116
C.17 NOK_generator_RPM	116
C.18 NOK_generator_temp.m	117
C.19 OK_generator_RPM	117
C.20 OK_generator_temp.m	117
C.21 NN.m	117
C.22 NN_results.m	117
C.23 Pregenerated_data_RPM.mat	117
C.24 Pregenerated_data_temp.mat	117
C.25 Pretrained_NN_rpm.m	117
C.26 Pretrained_NN_temp.m	117
D Obsah CD	119

Seznam obrázků

1.1	Proces prediktivní analýzy [28]	19
1.2	Tabulka životnosti [9]	26
1.3	Kaplan-Meierova křivka [29]	26
1.4	Pravděpodobnost přežití po dobu 0-10 měsíců [11]	27
1.5	Pravděpodobnost chyby v závislosti na délce doby mezi poruchami [11]	28
1.6	Rozhodovací klasifikační strom CART	29
1.7	Strom vytvořený algoritmem ID3	31
1.8	a) Lineární model b) MARS model [20]	32
1.9	Strojové učení. Rozdělení na bílé a černé kruhy [31]	33
1.10	Diagram zesíleného učení [31]	34
1.11	Vyčištění obrazu pomocí rozptýleného slovníkového učení [21]	36
1.12	Formální neuron s biasem [22]	38
1.13	Příklady aktivačních funkcí [22]	39
1.14	Třívrstvá neuronová síť [22]	40
1.15	Rozhodovací tabulka hraní golfu [24]	41
1.16	KNN algoritmus [25]	42
1.17	Přednost hlubokého učení na velkém množství dat [32]	44
1.18	Ukázka LSTM sítě[35]	46
1.19	LSTM hlavní větev[35]	47
1.20	LSTM brána[35]	47
1.21	LSTM krok 1[35]	48
1.22	LSTM krok 2[35]	48
1.23	LSTM krok 3[35]	49
1.24	LSTM krok 4[35]	49
1.25	Vysvětlení parametrů precision a recall [37]	50
2.1	Způsob ukládání dat z výroby	53
2.2	Blok válců[27]	54
2.3	Detail linky[27]	56
2.4	Zdrsnění pískováním [27]	57
2.5	Zdrsnění vysokolakovou vodou [27]	57
2.6	Zdrsnění mechanické [27]	58
2.7	Detail povrchu [27]	58
2.8	Detail plazmování bloku válců TWS [27]	60
2.9	Detail nanášení plazmy na blok válců TPS [27]	60
2.10	Podrobnosti inspekce bloku motor[27]	61
3.1	Diagram popisující běh programu	66
3.2	Okno uživatelského vstupu výběru metody	67

3.3	Okno uživatelského vstupu specifikace vstupních dat pro regresní analýzu	68
3.4	Speciální nastavení metody regrese	68
3.5	Okno uživatelského vstupu generátoru dat pro neuronovou síť	69
3.6	Speciální nastavení metody regrese	69
3.7	Příklad proložení hodnot teploty gaussovou křivkou	73
3.8	Příklad proložení hodnot otáček gaussovou křivkou	74
3.9	Příklad intervalu ležících mimo okolí střední hodnoty	75
3.10	Příklad zavolání funkce regr	76
3.11	Ukázka tisku dat	76
3.12	LSTM krok 4[35]	77
3.13	Ukázka složení trénovacích dat	78
3.14	Ukázka složení testovacích dat	78
3.15	Příklad vygenerovaných OK dat teploty	80
3.16	Příklad vygenerovaných NOK dat teploty	81
3.17	Příklad vygenerovaných NOK dat otáček	82
3.18	Příklad vygenerovaných NOK dat otáček	83
3.19	Parametry učení neuronové sítě	84
3.20	Průběh učení neuronové sítě pro teplotu	86
3.21	Průběh učení neuronové sítě pro teplotu	86
3.22	Průběh učení neuronové sítě pro rychlost otáček	87
3.23	Průběh učení neuronové sítě pro rychlost otáček	87
3.24	Porovnání výsledků metod	89
3.25	Jednotlivé gradienty kritického intervalu	90

Seznam tabulek

3.1	Tabulka NS rychlosti otáček	88
3.2	Tabulka NS teploty	88

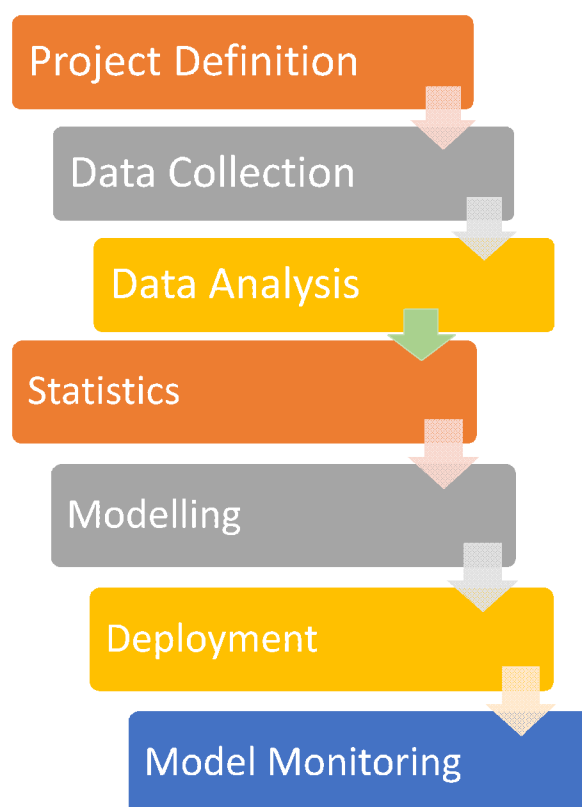
Úvod

Prediktivní analýza získává na popularitě z důvodu zlepšení efektivity výroby, redukce nákladů a hlavně snížení počtu poruch v automatizované výrobě. Dalším přínosem je snížení zmetkovitosti, což má za následek nižší zatížení životního prostředí. Ideálním cílem by pak byla úplná eliminace zmetků a potřeby šrotace neshodných kusů. Návrh prediktivní analýzy pro předcházení chybám na výrobní lince plazmy ve Škoda auto a.s. vyžaduje návrh komplexního modelu s ohledem na velké množství parametrů. Cílem této semestrální práce je nastudovat a popsat různé metody prediktivní analýzy a systém výroby motorového bloku ve Škoda auto a.s. Následně analyzovat a popsat datové bloky dostupné z jednotlivých strojů, vybrat vhodné metody pro analýzu reálných dat, realizovat je a porovnat jejich výsledky. První část této práce bude obsahovat jednotlivé kapitoly popisující metody prediktivní analýzy. Druhá část bude obsahovat popis výrobní linky bloku motoru a finální kontroly. Třetí částí pak bude popis jednotlivých druhů sběrů dat a důležité informace k těmto parametrům ovlivňujícím kvalitu výroby. Ve čtvrté části se pak bude nacházet návrh a realizace generátoru dat. Tato část je nezbytná pro úspěšné dokončení práce, protože situace ve světě nedovolila dokončení výrobní linky produkující data pro tuto práci, a proto je nutné tato data získat jinak. Pátá část bude výběr, navržení, realizování a popis dvou metod vhodných pro prediktivní analýzu. Poslední částí bude porovnání obou metod a výsledné hodnocení.

1 Teoretická část diplomové práce

1.1 Prediktivní analýza

Prediktivní analýzu je možné vytvářet za pomoci mnoha různých technik. Nejdůležitější části jsou dolování dat, prediktivní modelování, hluboké strojové učení a umělá inteligence. Pomocí těchto technik dochází k analýze nashromážděných dat a vytvoření hypotézy (predikce) chování systému v budoucnosti. Prediktivní analýza je pojem, který v sobě zahrnuje velké množství různých analytických a statistických postupů. Tyto postupy mají za úkol nějakým způsobem odhadovat a predikovat jak se v budoucnu budou chovat sledované procesy. Způsob jakým je tento odhad vyjadřován se liší v závislosti na druhu a chování procesu, jež popisují (predikují). Většina prediktivních modelů pracuje s takzvanou pravděpodobností odhadu. Pro jeden proces dojde k odhadu několika možných vzorců chování a těmto vzorcům je zároveň přidělena pravděpodobnost, s jakou se mohou vyskytnout. Jádrem celé prediktivní analýzy je zachycení vztahů mezi vysvětlujícími proměnnými a odhadnutými proměnnými z minulých událostí a jejich využití pro odhadnutí výsledku. Přesnost tohoto odhadu záleží především na kvalitě analýzy dat a validitě předpokladů.[1]



Obr. 1.1: Proces prediktivní analýzy [28]

Typický postup je znázorněn na obr 1.1 a podrobně lze popsat:

- Project Definition: V této části dochází k určení cílů projektu, požadovaných výsledků, finančních prostředků, datových sad atd.
- Data collection: Zde dochází k dolování dat z několika různých zdrojů, díky čemuž je získán široký pohled na věc pro prediktivní analýzu.
- Data analysis: V tomto kroku dochází k prohlídce dat. Hlavním cílem této operace je nalezení užitečných dat a zbavení se neužitečných.
- Statistics: Zde dochází k ověření hypotéz a předpokladů. Zároveň dochází k jejich testování pomocí statistických modelů.
- Predictive modeling: V této části dochází k vytvoření prediktivního modelu, který se snaží co nejvíce odpovídat budoucnosti. Je také možnost vytvoření více modelů podle jejich hodnocení.
- Deployment: Zde dochází k nasazení modelu do reálného běhu systému a jsou získávány informace ohledně jeho chování a výsledky jeho rozhodování.
- Model monitoring: Nakonec dochází ke sledování modelu a jeho průběžným kontrolám za účelem zjištění funkčnosti modelu a ověření správného rozhodování.

1.2 Dolování dat

Pojem dolování dat (z angl. Data mining) je velkou částí prediktivní analýzy dat. Hlavním úkolem tohoto procesu je nacházení trendu, vzorců, a různých vztahů napříč zkoumanými daty. Touto metodou jsou nalezeny různé informace. Těchto informací je následně využito pro vytvoření prediktivního modelu. Hlavním pilířem prediktivní analýzy, prediktivních modelů i dolování dat jsou stále sofistikovanější statistické modely. Součástí těchto modelů jsou také vícerozměrné statistické analýzy dat jako pokročilá regrese (advanced regression) nebo také model časové řady (time series model). Firmy využívají tyto techniky pro odhalení na první pohled neznámých souvislostí a pomáhají odhadovat budoucí chování sledovaných objektů.[1]

1.3 Kauzální vysvětlení vs. empirická predikce

V mnoha odvětvích (ekonomie, psychologie, výroba, věda) jsou používány statistické modely pro jednoduché vysvětlení problematiky a modely s vysokou výpovědní hodnotou pro prediktivní analýzu. Mezi těmito dvěma pohledy je velmi tenká hranice, ale pro jejich správné použití a vytvoření správného modelu je potřeba znát rozdíl. Pro vytvoření vědeckých teorií je třeba znát oba tyto pohledy, ale je potřeba vědět jakou roli hrají v tomto procesu. Oba tyto nástroje využívají vytváření modelu.

Tento model je vytvořen postupně od úvodního cíle přes definici, výzkum, sběr dat až po vědecké využití.[1]

1.3.1 Vysvětlování a vysvětlující modelování (Explanatory Modeling)

Ve většině vědeckých odvětví je tato metoda používána pro ověření kauzálních teorií. Na získaná data jsou aplikovány statistické modely pro ověření předpokládaných příčin. U tohoto modelu se předpokládá souhrn základních faktorů měřených proměnnou X , způsobujících základní efekt, měřený pomocí hodnot Y . Ve většině případů samotná teorie zaručuje kauzalitu. Jinými slovy se dá říci, že teorie je v tomto případě velmi silný faktor a věrohodnost dat a statistického modelu je závislá na teoretickém modelu. Vztah teorie a naměřených dat se velmi liší vzhledem k odvětvím použití. Například ve společenských vědách je na teorii kladen velký důraz, v bioinformatice je význam teorie mnohem menší. Existuje definice vysvětlování jako vysvětlení příčin a vysvětlující modelování. Toto modelování je používání statistických modelů pro testování příčinných vysvětlení.[1]

1.3.2 Prediktivní modelování

Jedna z definic prediktivní modelování říká, že se jedná o proces aplikování statistického modelu nebo dolování dat za účelem získání predikce nebo odhadnutí budoucího chování. Ve většině případů se jedná o proces odhadnutí výstupní veličiny Y při známé hodnotě vstupní veličiny X . Zároveň dochází k pozorování vstupu do času t a předpovídání výstupu hodnoty v čase $t+k$ (kde $k < 0$). Predikce se skládá z více částí. Například hodnotu nebo interval hodnot přisuzovaných výstupu s určitou pravděpodobností, oblasti predikce, prediktivní rozdělení nebo hodnocení jednotlivých predikcí. Model se nazývá prediktivním bez ohledu na to jakým přístupem byl získán. Základní přístupy jsou: Bayesovský nebo častý (frequentist), parametrický nebo neparametrický, metoda dolování dat nebo statický model atd.[1]

1.3.3 Deskriptivní modelování

Nejčastěji používaný model pro modelování je model deskriptivní. Hlavní záměrem tohoto modelování je shrnout a vyjádřit datové struktury přehledným způsobem. Toto modelování se nespolehá na základní popisnou teorii tolik jako vysvětlující modelování. Predikce se v tomto druhu modelování neuvažuje.[1][2]

1.4 Analytické techniky

Pro prediktivní analýzu se používají různé techniky, které mohou být dvou hlavních skupin. První skupinou jsou techniky regresní a druhou skupinou techniky strojového učení.

1.4.1 Regresní techniky

Tyto metody jsou základním pilířem prediktivní analýzy. Hlavním bodem těchto technik je vytvoření korektního modelu, jež bude schopen vytvořit propojení mezi různými proměnnými pro predikci.

Lineární regresní model

Tato technika se snaží nacházet vztah mezi odezvou (nebo závislou proměnnou) a predikovanou proměnnou (nezávisle proměnnou). Může se jednat o predikování jedné proměnné, pak se technika nazývá základní lineární regresí. Nebo o predikování více proměnných neboli vícenásobná lineární regrese. Dochází postupně k úpravě predikovaných parametrů, dokud model není dostatečně přizpůsoben požadavkům. Model je vytvářen z naměřených dat. Nejdůležitějšími prvky jsou pak jak minimalizování zbytku a také důraz na náhodné rozdělení s ohledem na predikci modelu. Nejčastějším předpokladem bývá, že vážený průměr hodnoty odezvy je afinní funkcí predikovaných proměnných. Jako další možné předpoklady jsou medián nebo jiný kvantil hodnot. Jako většina regresních analýz, tak i lineární regrese se zaměřuje především podmíněné rozdělení pravděpodobnosti odezvy vzhledem k hodnotám predikcí. Cílem lineární regrese je minimalizovat součet druhých mocnin zbytků. Toho se dosáhne za pomoci metody nejmenších čtverců. Výsledkem použití této metody dostáváme nejlepší nestranné lineární odhady (BLUE angl.), za předpokladu, že je splněn Gauss–Markovův teorém. Tento teorém říká, že v lineárním regresním modelu, kde jsou chyby nekorelované, mají stejnou odchylku a stejnou předpokládanou nulovou hodnotu, je nejlepší nestranný lineární odhad dán obecnou metodou nejmenších čtverců (OLS). Po vytvoření modelu je dalším krokem ověření spolehlivosti jednotlivých proměnných modelu. Pro to je nejčastěji použita kontrola statistické významnosti koeficientů modelu pomocí t-statistiky (poměr odlišnosti odhadované veličiny od teoretické hodnoty ke standardní chybě). Což odpovídá testování, jestli se koeficient významně liší od nuly.[2][4]

Diskrétní výběrové modely

U vícenásobné regrese je typické, že odezva je spojitá a má neomezený rozsah. Velmi často se může stát, že odezva bude diskrétní. Přestože by bylo matematicky korektní

použít vícenásobnou regresi na diskretní závislé proměnné, tak některé předpoklady teorie vícenásobné regrese přestávají platit. Právě proto je vhodnější používat modely s diskretním výběrem. Pokud je závislá proměnná diskretní, existuje několik vhodných metod. První metoda je logistická regrese. Tato metoda se používá k modelování pravděpodobnosti nějaké události (1 nebo 0, výhra nebo prohra). Možným rozšířením je hledání určitých objektů v obrazu. To spočívá v přidělování pravděpodobnosti každému detekovanému objektu od 0 do 1. Druhá metoda je multinomiální logistická regrese. U této metody je základem možnost více než dvou diskretních výstupů. To znamená, že model se používá pro predikování pravděpodobností různých výstupů kategoricky rozdělené závislé proměnné, danou množinu nezávisle proměnných (reálné, binární nebo kategorické). Třetí metoda je metoda probit. Teorie pravděpodobnosti definuje probit jako kvantil funkci spojenou s normálním rozdělením (zápis je $N(0,1)$). Přestože výsledky těchto modelů bývají podobné jako výsledky logistické regrese, základní rozdíl je v rozdělení. Často je tato metoda použita v sociálních vědách, jako je ekonomie. Druhá a třetí metoda se používá, když je závislá proměnná binární.[3][4]

Modely časových řad

Modely časových řad se používají pro predikci chování různých proměnných v budoucnosti. Hlavním předpokladem této metody je, že naměřená data v čase mohou mít interní strukturu (autokorelace, trend nebo kolísání) a proto na ně nelze použít klasické regresní metody. Speciálně pro tento druh dat byla vytvořena metoda modelů časových řad. Tato metoda rozděluje získaná data na trendové, sezónní a cyklické složky. Mezi často používané modely časových řad patří autoregresní model (AR). Tento model se používá ve statistice, ekonometrii a zpracování signálů. Jedná se o náhodný proces popisující časově proměnné jevy v přírodě, ekonomii atd. Definice modelu říká, že se jedná o model, jehož výstup závisí lineárně vždy na předchozích hodnotách a na stochastickém (má náhodné rozdělení pravděpodobnosti nebo vzorec, který může být analyzován, ale nemůže být predikován) členu. To znamená, že model je ve formě stochastické diferenční rovnice.[6]

$$x_t = c + \sum_{i=1}^p \varphi_i X_{t-i} + \varepsilon_t \quad (1.1)$$

Kde φ_1 až φ_p jsou parametry modelu, c je konstanta a ε_t je bílý šum. Další často používaný model je model klouzavého průměru (MA). V analýze časových průběhu se běžně používá pro modelování jednorozměrných časových řad. Tento model závisí pouze na aktuálních a minulých hodnotách stochastického členu.[6]

$$x_t = \mu + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \dots + \theta_q \varepsilon_{t-q} \quad (1.2)$$

Kde μ je průměr řady, θ_1 až θ_q jsou parametry modelu $\varepsilon_t, \varepsilon_{t-1}, \dots, \varepsilon_{t-q}$ jsou chyby způsobené bílým šumem. Tyto dva modely společně tvoří autoregresní model klouzavého průměru. Pokud máme nějakou časovou řadu X_t , je možné využít modelu ARMA pro popis a pochopení dat a možná pro predikování budoucího chování. V první části (AR) dochází k nalezení vztahu mezi aktuální hodnotou a jejími minulými hodnotami. Druhá část (MA) se zabývá vytvářením chybového modelu jako lineární kombinace odchylek vyskytujících se současně v různých časových okamžicích naměřených dat. Zápis modelu vypadá většinou jako ARMA(p,q) kde p je pořadí části AR a q je pořadí MA části.[4][6]

$$x_t = c + \varepsilon_t + \sum_{i=1}^p \varphi_i X_{t-i} + \sum_{i=1}^q \theta_i \varepsilon_{t-i} \quad (1.3)$$

Kde φ_1 až φ_p jsou parametry modelu AR, θ_1 až θ_q jsou parametry modelu MA, $\varepsilon_t, \varepsilon_{t-1}, \dots, \varepsilon_{t-q}$ jsou chyby způsobené bílým šumem a c je konstanta.

Pro popis stacionárních časových řad se používá především model ARMA. Pro popis nestacionárních časových řad se používá model ARIMA. U autoregresního integrovaného modelu klouzavého průměru lze použít opakovanou aplikaci rozdílového (rozdíl aktuální a předchozí hodnoty) kroku pro odstranění nestacionarity. Část AR v tomto modelu obsahuje regresní analýzu proměnných na jejich minulých hodnotách. Další část MA obsahuje lineární kombinaci odchylek vyskytujících se současně na více místech v minulosti. Poslední částí je označena písmenem I. Tato část zajišťuje nahrazení hodnoty, rozdílem aktuální a předchozí hodnoty. Tyto tři části mají za úkol co nejlépe upravit model, aby vyhovoval datům.[4][6]

$$X_t - \alpha_1 X_{t-1} - \dots - \alpha_{p'} X_{t-p'} = \varepsilon_t + \theta_1 \varepsilon_{t-1} + \dots + \theta_q \varepsilon_{t-q} \quad (1.4)$$

$$\left(1 - \sum_{i=1}^{p'} \alpha_i L^i\right) X_t = \left(1 + \sum_{i=1}^q \theta_i L^i\right) \varepsilon_t \quad (1.5)$$

Kde X_t je datová řada (real), t je index (int), L je operátor zpoždění o jeden krok (lag) α_1 jsou parametry autoregresní části modelu, θ_i jsou parametry MA části a ε_t je odchylka.[6]

Pánové Box a Jenkins (tvůrci metody aplikace metod ARMA a ARIMA na datové řady), navrhli třífázovou metodu aplikace metod na datové řady zahrnující identifikaci modelu, odhad a validaci. Proces identifikace zahrnuje ověření, zda je

soubor dat stacionární a jestli se data opakují pomocí zkoumání grafů řady aplikováním autokorelačních a částečně autokorelačních funkcí. Tato analýza pomáhá rozhodnout, zda-li by měla být použita autoregresní analýza nebo klouzavý průměr. V druhé části dochází k odhadu parametrů pomocí výpočtových algoritmů, aby se zajistilo dosažení co nejpřesnějších koeficientů pro model ARIMA. K tomu se nejčastěji používá nelineární metoda nejmenších čtverců. Poslední částí této aplikace Box-Jenkins je kontrola modelu. Kontroluje se, jestli odhadovaný model odpovídá původním specifikacím stacionárního procesu. Residua (zbytky) by měla být ideálně na sobě nezávislé a jejich průběh v čase by měl být nezávislé a mít konstantní průměr a rozptyl v čase. Při zjištění nedostatečně kvalitního odhadu je nutné se vrátit zpět k první části a proces opakovat sestavením lepšího modelu.[13][14]

Analýza „přežívání“ nebo trvání (Survival or duration analysis)

Další název této metody je analýza času do události. Hlavním odvětvím kde se tyto techniky používají je zdravotnictví a biologické vědy. Také jsou používány v ekonomii a strojírenství (spolehlivost, doba provozu, doba selhání atd.). Tato analýza se snaží odpovědět na otázky typu: Jsou nějak určitá selhání propojena? Za jak dlouho dojde k poruše? Jak konkrétní události zvyšují nebo snižují pravděpodobnost selhání? Hlavním prvek této analýzy je životnost. U biologických aplikací je smrt jednoznačná, ale mechanická spolehlivost může být poněkud komplikovanější. Existuje částečná porucha, systém se může sám zotavit nebo může dojít k opravě. V případě biologických selhání mohou mít některé události stejnou dvojnásobnost (infarkt, selhání některého orgánu). Pro kvalitní analýzu je důležité mít události dobře definované společně s časovými okamžiky jejich výskytu. Existují i modely, které se přímo specifikují na nejednoznačné události. Obecně analýza přežití zahrnuje pokus o namodelování časových údajů do události (událost např. selhání, smrt). Většinou existuje pouze jediná událost, která když nastane, způsobí smrt nebo rozbití. Avšak vyskytují se opakující se události, které předpoklad jediného selhání uvolňují. V mnoha oblastech jsou právě tyto opakující se události klíčové a mají velký vliv na spolehlivost systému.

- Událost: smrt, porucha, opakování poruchy, zotavení nebo jiná zajímavá událost
- Čas: čas od začátku pozorování, čas do události, čas do ukončení pozorování, čas do ztráty kontaktu.
- Cenzura/cenzurované pozorování: Období kde se během doby pozorování nevyskytne žádná událost, se popisuje jako cenzurované. Po této době není o objektu nic známo. U objektu se po ukončení cenzury může, a nebo nemusí vyskytnout událost.

- Funkce přežití $S(t)$: Určuje pravděpodobnost bezporuchového provozu (přežití) objektu po dobu t . [7][8]

Analýza přežívání se používá pro různé aplikace:

1. K popisu určitých událostí

- (a) Tabulka života: Ukazuje jaká je pravděpodobnost selhání (přežití) určité pro časové okamžiky života objektu. [9]

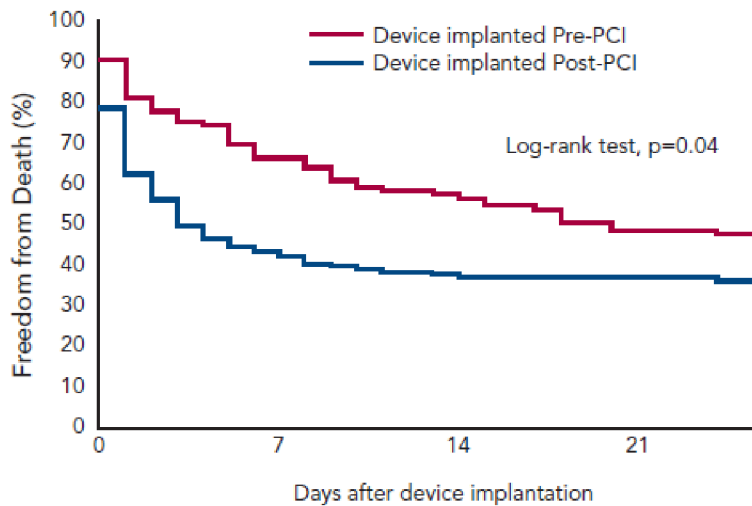
Table 1. Life table for the total population: United States, 2003 Click here for spreadsheet version

Age	Probability of dying between ages x to $x+1$	Number surviving to age x	Number dying between ages x to $x+1$	Person-years lived between ages x to $x+1$	Total number of person-years lived above age x	Expectation of life at age x
	q_x	l_x	d_x	L_x	T_x	e_x
0-1	0.006665	100,000	667	99,394	7,743,016	77.4
1-2	0.004669	99,313	47	99,290	7,643,622	77.0
2-3	0.00337	99,267	33	99,250	7,544,332	76.0
3-4	0.00254	99,233	25	99,221	7,445,082	75.0
4-5	0.00194	99,208	19	99,199	7,345,861	74.0
5-6	0.00177	99,189	18	99,180	7,246,663	73.1
6-7	0.00160	99,171	16	99,163	7,147,482	72.1

Obr. 1.2: Tabulka životnosti [9]

- (b) Kaplan-Meierova křivka: Tato křivka ukazuje, jak se vyvíjí určitý aspekt sledovaného objektu v čase (pravděpodobnost přežití, nezaměstnanosti atd.). [10]

Figure 2: Kaplan–Meier curves for freedom from death (to 30 days) by device implanted pre/post PCI



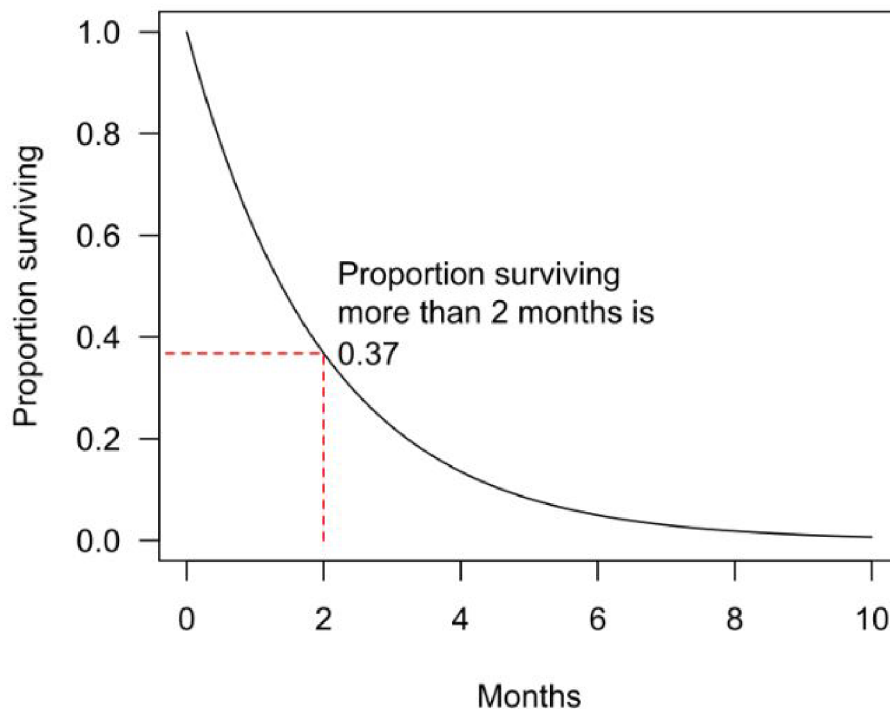
Source: Basir et al, 2017a

Obr. 1.3: Kaplan-Meierova křivka [29]

- (c) Funkce přežití: Tato funkce vyjadřuje pravděpodobnost bezporuchového provozu (přežití) sledovaného objektu, neboli že nenastane porucha do určitého okamžiku. Další známý název je funkce spolehlivosti.[11]

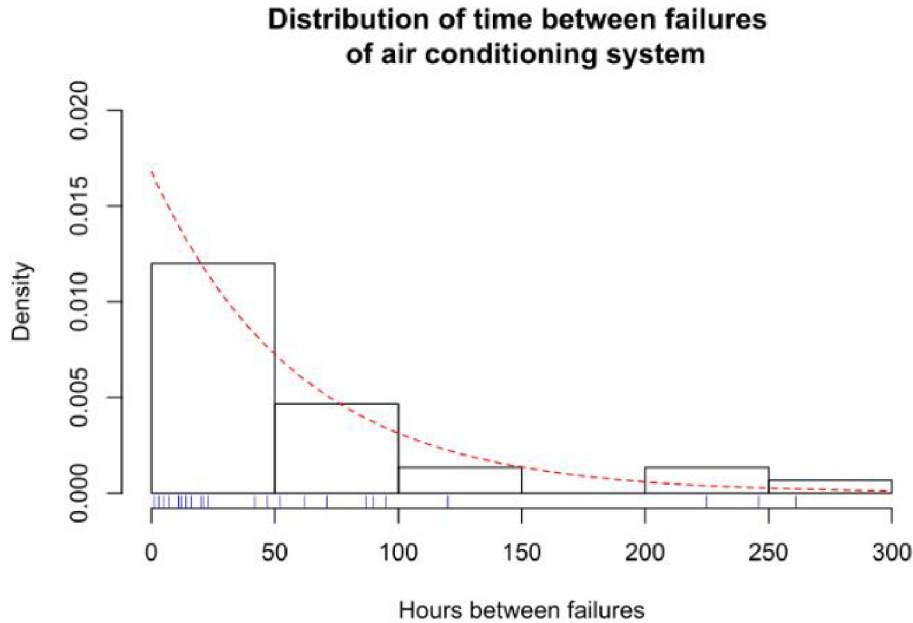
$$S(t) = P(\{T > t\}) = \int_t^{\infty} f(u)du = 1 - F(t) \quad (1.6)$$

Kde T je spojitá náhodná veličina s kumulovanou distribuční funkcí F(t) na intervalu $[0, \infty)$. Příklady grafů funkcí spolehlivosti:



Obr. 1.4: Pravděpodobnost přežití po dobu 0-10 měsíců [11]

- (d) Poruchovost: Tímto se vyjadřuje, s jakou četností dochází na systému k poruše, v jednotkách poruchy za jednotku času. Často se pro označení používá písmeno λ . Míra poruchovosti se u objektu může s časem výrazně měnit. Například u různých strojů (automobil, výrobní linka) nejspíše bude v desátém roce používání mnohem vyšší poruchovost než v prvním. Nejčastěji dochází k opotřebení mechanických částí. Největší opotřebení je způsobeno chybou obsluhy (např. nedoplnění chladící kapaliny, nevyměnění oleje atd.). V praxi často vyskytuje termín MTBF (střední doba mezi poruchami z angl. Mean time between failures) označovaný jako $1/\lambda$. [9][11]



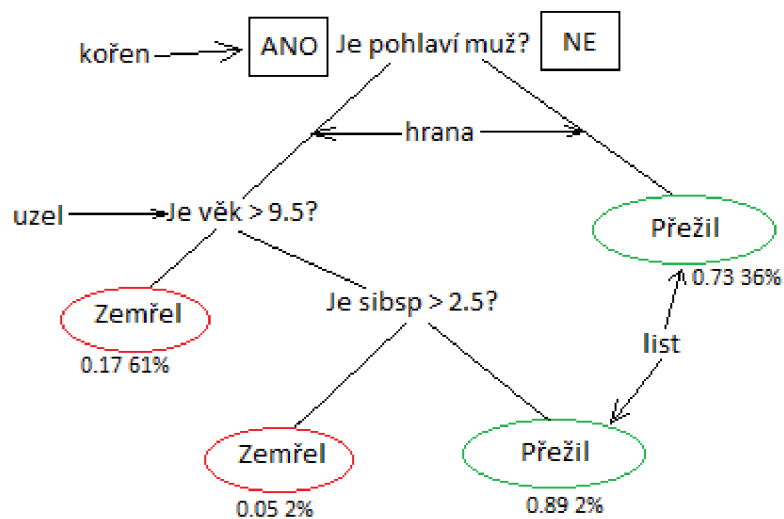
Obr. 1.5: Pravděpodobnost chyby v závislosti na délce doby mezi poruchami [11]

2. Pro porovnání doby životnosti dvou nebo více skupin
 - (a) 2.1. Log-rank test: Je test hypotézy (hypotéza je testovatelná na základě testování procesu pomocí sady náhodných dat) pro porovnání rozdělení doby přežití dvou vzorků. Jedná se o neparametrický test, jehož hlavní použití je na zkreslená a cenzurovaná data. V medicíně se často používá na porovnání nové léčebné metody a kontrolní léčby kdy je měřen čas do události (např. doba o počátku léčby do infarktu). Definice tohoto testu říká, že dochází k porovnání odhadů poruchovosti dvou skupin podle času událostí. Test je vytvořen výpočtem z pozorovaného a předpokládaného počtu událostí v jedné ze skupin v každém pozorovaném okamžiku a jejich následné sečtení pro získání celkového shrnutí ve všech časových okamžicích kdy se objevila událost.[12]
3. Pro popis vlivu kategorických a kvantitativních proměnných na dobu přežití
 - (a) Coxova proporcionální rizikovost (Cox proportional hazards regression): Modely proporcionální rizikovosti tvoří třídu modelů přežití ve statistice. Tyto modely vypovídají o době, která uběhne, než nastane událost. Tato doba je vztažena k jednomu nebo více kovariátům (data nashíraná před skutečným experimentem, mohou zpřesnit analýzu) spojenými s touto událostí. Účinek přidání kovariátu má multiplikační charakter s ohledem na míru poruchovosti (například změna materiálu výrobku může zvýšit životnost na dvojnásobek).[15]

4. Parametrické modely přežití (Parametric survival models)
5. Stromy přežití (Survival trees)
6. Survival random forests

Klasifikační a regresní stromy (CART)

Rozhodovací stromy se používají jako prediktivní modely k propojení pozorované veličiny a cílové hodnoty (listy). Tento prediktivní model je používán ve statistice, dolování dat a strojovém učení. U modelů, kde cílová veličina nabývá diskretních hodnot, nazýváme tyto modely klasifikační stromy. U těchto stromů představují listy jednotlivé třídy a větve spojení vedoucí k těmto listům. O regresních stromech mluvíme, pokud cílová proměnná může nabývat spojitých hodnot (real).[16][17][18] U rozhodovací analýzy se stromy používají k vizualizaci dat a zobrazení procesu roz-



Obr. 1.6: Rozhodovací klasifikační strom CART

hodování. Při procesu dolování dat tento strom popisuje data, ale výsledný strom může být vstupem pro rozhodování. Metoda rozhodovacích stromů se často používá při dolování dat. Cílem použití této metody je vytvoření modelu, který bude predikovat výstupní hodnoty systému, s ohledem na vstupní veličiny (viz. Obr. 1.6). Každý vnitřní uzel odpovídá jedné vstupní veličině, a z každého uzlu vystupují hrany pro každou z možných hodnot, kterou může vstupní proměnná nabývat. V každém listu se nachází cílová proměnná reprezentovaná cestou od kořene až po list. Aby se strom mohl nazývat rozhodovacím, musí mít každý jeho interní (nelistový) uzel vstupní funkci.[16][17][18]

Výhody rozhodovacích stromů:

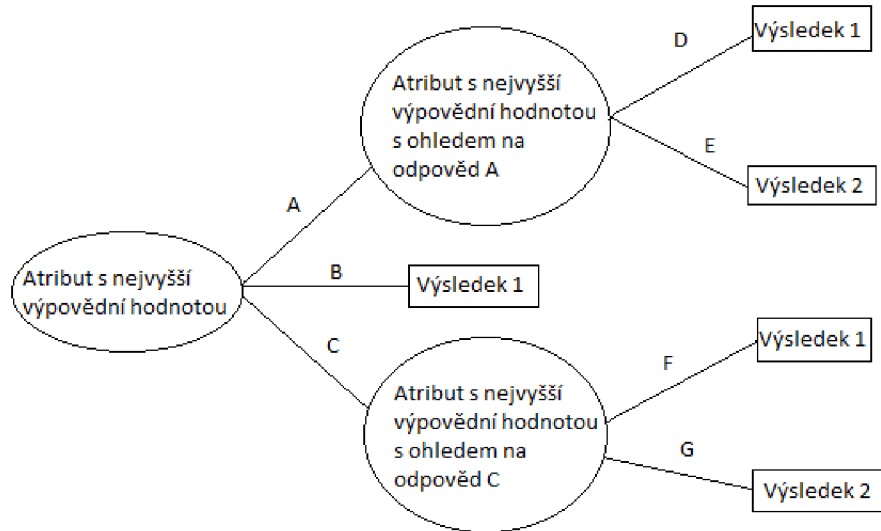
- Velmi jednoduché na pochopení a interpretaci. Pro pochopení této techniky stručné vysvětlení. Snadná interpretace spočívá v grafickém zobrazení.

- Zvládají jak numerická tak kategorická (nabývá jednu z možných hodnot) data. Ostatní metody většinou zvládají pouze proměnné jednoho typu proměnné.
- Data před použitím nevyžadují velkou přípravu. Nevyžadují normalizaci dat.
- Umožňují použití modelu white box (subsystém jehož prvky lze zobrazit a obvykle se nemění). Přístup k vnitřní struktuře systému umožňuje snadnější pochopení systému, ale také ho usnadňuje jeho zneužití (hacknutí). Programátor může vidět do kódu a tím najít jeho slabiny, což činí testování systému jednodušší.
- Možnost ověření modelu použitím statistických testů. To umožňuje ověření spolehlivosti modelu.
- Mohou dobře pracovat s velkými datsety. Velké množství dat se pomocí standardních výpočetních zdrojů da zvládnout v přiměřeném čase.
- Dokáží velmi dobře imitovat rozhodování člověka. To se může hodit při modelování lidského rozhodovacího algoritmu (chování).
- Lze je aproximovat jakoukoli booleovskou funkcí (př. XOR).[18]

Omezení rozhodovacích stromů:

- Stromy mohou být málo robustní. I malá změna v trénovacích datech může způsobit obrovskou změnu v celém stromu a změnit finální predikce.
- Problém učení optimálního rozhodovacího stromu je NP-kompletní (lze ho vyřešit pomocí hrubé síly (brute force algoritmu) a může být využit pro jiné problémy s podobným algoritmem) při dodržení určitých podmínek dokonce i pro jednoduché modely. Z toho důvodu je učení rozhodovacích stromů heuristické (chamtivý algoritmus z angl. Greedy algorithm), to znamená, že v každém uzlu se provádí lokálně optimální rozhodnutí. Kvůli tomu není možné zaručit globální optimálnost stromu. Z toho důvodu byli vyvinuty složitější modely stromů pro zdvojnásobení dosahu informace (DID, dual information distance).
- Učící algoritmy mohou vytvořit příliš složité stromy, které příliš nezobecní trénovací data. Tento jev je známý jako přesahování (angl. overfitting), to znamená, že analýza odpovídá testovaným datům. To by mohlo způsobit nemožnost použití algoritmu na další sérii testovacích dat. Je zapotřebí použití algoritmu prořezávání (angl. pruning), která sníží velikost stromu pomocí odstranění některých částí, jež mají malou klasifikační sílu. To umožní snížit komplexnost stromu a zároveň zvýší jeho univerzálnost a přesnost.
- Pro data obsahující kategorické proměnné s různým počtem úrovní, dochází k upřednostnění atributů s více úrovněmi. Tomu se dá vyhnout použitím přístupu podmíněného odvození (Conditional Inference approach), dvou úrovnovým přístupem (two-stage approach) nebo adaptivním vynecháním jednoho (adaptive leave-one-out feature selection).[18]

Algoritmus ID3 patří mezi často používané rozhodovací stromy. Slouží ke generování stromu z datové sady. Je to předchůdce C4.5 a používán je v doménách strojového učení a zpracování přirozeného jazyka.



Obr. 1.7: Strom vytvořený algoritmem ID3

Jednotlivé kroky algoritmu:

1. Vypočítání entropie (náhodnosti) každého atributu a z datasetu S.
2. Rozdělení datasetu S na subsety pomocí atributu s nejmenší entropií (neboli s nejvyšší výpovědní hodnotou).
3. Vytvoření uzlu s daným vybraným atributem
4. Rekurzivní rozdělení na zbývající podmnožiny pomocí zbývajících atributů.[19]

Multivariační adaptivní regresní splajny (MARS)

Jedná se o jednu z forem regresní analýzy, která je neparametrická a lze ji považovat za rozšíření lineárních modelů, s tím rozdílem, že automaticky analyzuje a modeluje nelinearity a interakce mezi proměnnými.[20] **Model MARS:**

$$\hat{f}(x) = \sum_{i=1}^k c_i B_i(x) \quad (1.7)$$

Model se skládá z vážené sumy základních funkcí $B_i(x)$ a konstantních koeficientů c_i . To znamená, že každá základní funkce je násobená koeficientem.[20]

Příklad funkce MARS:

$$ozone = 5.2^{1)} \tag{1.8}$$

$$+ 0.93max(0, temp - 58)^{2)} \tag{1.9}$$

$$- 0.64max(0, temp - 68)^{2)} \tag{1.10}$$

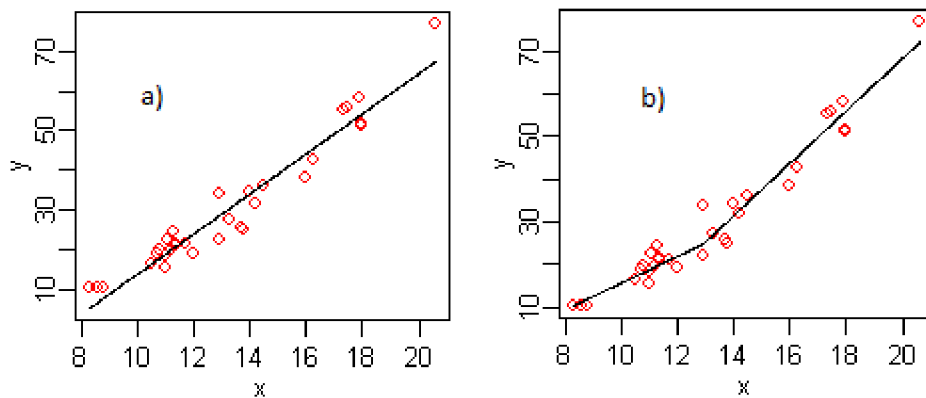
$$- 0.046max(0, 234 - ibt)^{2)} \tag{1.11}$$

$$- 0.016max(0, wind - 7)max(0, 200 - vis)^{3)} \tag{1.12}$$

$$\tag{1.13}$$

Popis funkce MARS:

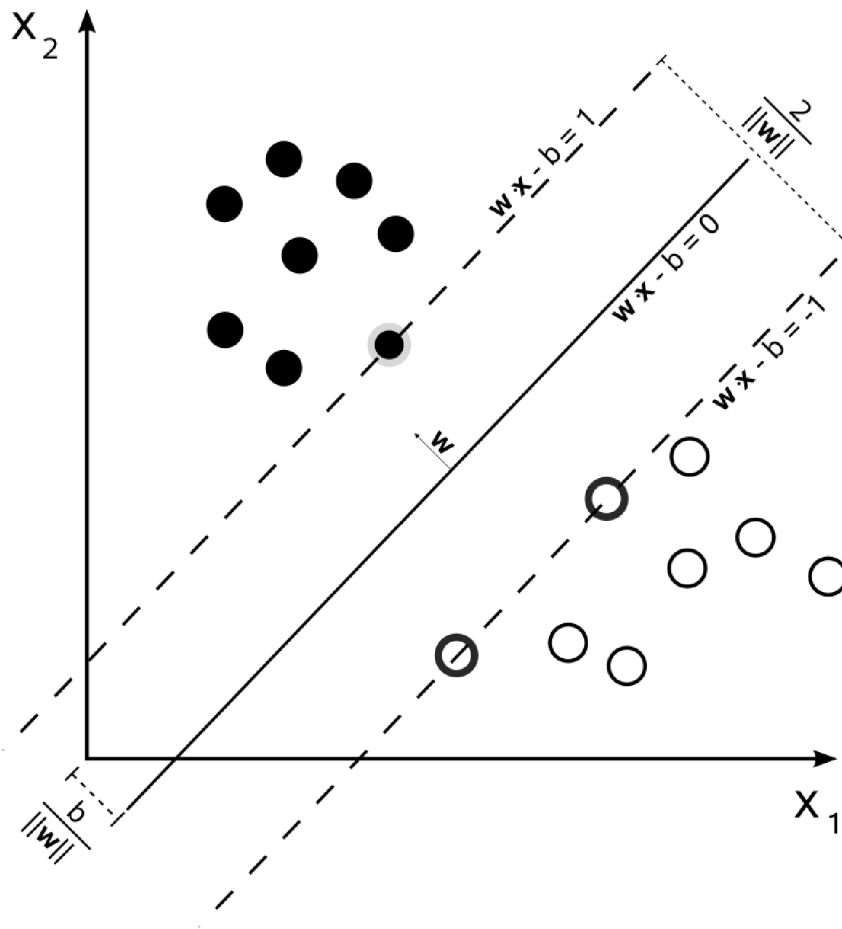
1. Konstanta (viz. Obr.1.8).
2. „Závěsná funkce“ (hinge function). Tato funkce je ve tvaru $max(0, x - constant)$ nebo $max(0, constant - x)$ (viz. Obr.1.9).
3. Součin dvou nebo více „závěsných funkcí“. Tyto základní funkce mohou modelovat vztah mezi dvěma a více proměnnými (viz. Obr.1.12). [20]



Obr. 1.8: a) Lineární model b) MARS model [20]

1.5 Techniky strojového učení (machine learning)

Jedná se o studium algoritmů a statistických modelů, používaných počítačovými systémy bez obsluhy nebo konkrétních příkazů (explicitních). Při rozhodování se spoléhají především na vzorce a odvození. Algoritmus strojového učení vytváří matematický model na základně tréninkových dat, aby dokázal předpovědět budoucnost nebo provedl rozhodnutí bez striktně naprogramovaného rozhodovacího algoritmu. Strojového učení se využívá v mnoha odvětvích. Například při filtrování emailů, nebo v počítačovém vidění, kde je poměrně složité nebo dokonce nemožné navrhnout konvenční algoritmus pro kvalitní splnění úkolu.[21]



Obr. 1.9: Strojové učení. Rozdělení na bílé a černé kruhy [31]

Druhy strojového učení:

1. Řízené učení

V tomto druhu učení se algoritmus pokouší o vytvoření matematického modelu pomocí zpracování daného datasetu, obsahujícího jak vstupní data, tak požadovaná výstupní data. Například u hledání určitého vzoru v obraze, bude testovací dataset obsahovat jak obrázky obsahující daný vzor, tak i obrázky

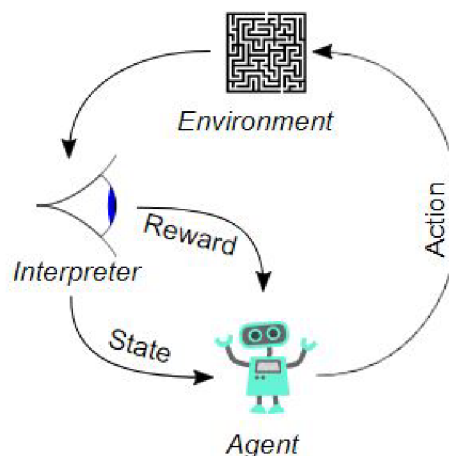
bez daného vzoru. Každý obrázek také obsahuje popis, zda daný vzor obsahuje. Každý objekt tréninkových dat může mít jeden nebo více vstupů a jeden žádaný výstup. Pomocí optimalizačního procesu, který se iterativně opakuje, dochází k nalezení funkce použitelné pro predikování výstupu na základě nových vstupů. Optimálně naučený algoritmus dokáže nalézt správný výstup z dat, jež nebyla součástí tréninku. Naučeným algoritmem lze nazvat algoritmus, který v průběhu času dokáže zlepšit své odhady. Řízené učení obsahuje jak regresní tak klasifikační algoritmy. Pokud jsou vstupní hodnoty omezeny na určitou sadu hodnot pak lze použít klasifikační algoritmy. Například filtr emailů. Vstupem je příchozí email a výstup je složka pro uložení daného emailu. Pro algoritmus detekující spam bude výstupem nastavení parametru spam na true nebo false. Regresní algoritmy mají oproti tomu nepřetržité výstupy, což znamená, že mohou nabývat jakékoli hodnoty z určitého rozsahu. Příkladem jsou spojité veličiny jako teplota, délka nebo váha.[21]

2. Částečně řízené učení

Tyto algoritmy vyvíjejí matematický model pouze z neúplných tréninkových dat. Část vstupních dat neobsahuje popis.

3. Zesílené učení

Algoritmy zesíleného učení dostávají odezvu buď jako pozitivní nebo negativní posílení v dynamickém prostředí. Tyto algoritmy neuvažují přesnou znalost matematického modelu. Často se využívají v autonomních vozidlech nebo učení bota do hry. Jedná se o strojové učení určující jaké akce by softwarový agent měl podnikat do prostředí, aby dosáhl maximální odměny (viz. obr.1.10). Toto učení se liší od řízeného učení hlavně tak, že výsledkem není prezentování páru vstup výstup, ale nalezení rovnováhy průzkumem (neznámá oblast) a využitím (současných znalostí).[21]



Obr. 1.10: Diagram zesíleného učení [31]

4. Neřízené učení

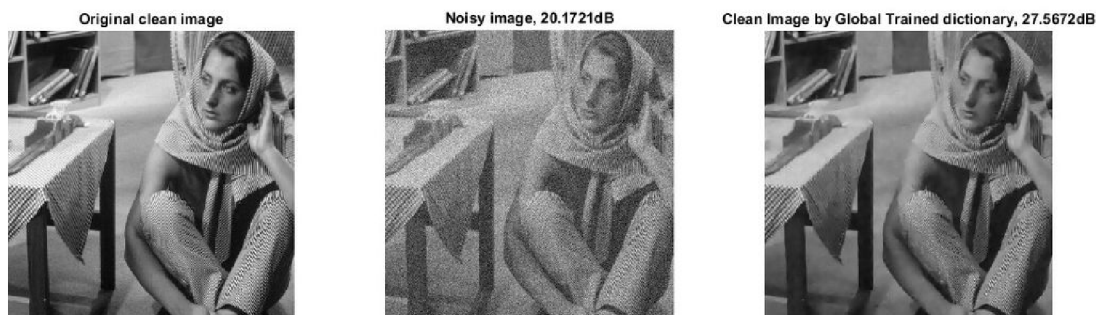
Data pro neřízené učení obsahují pouze vstupní data a snaží se v nich nalézt nějakou posloupnost nebo vzorec (seskupení nebo rozdělení dat). Tento algoritmus dokáže najít vzory v datech a dokáže seskupovat vstupy do kategorií. Snížení výpočetní náročnosti může být například v postupném snižování počtu funkcí ve vstupních datech. Algoritmus nachází společné rysy jednotlivých vstupů, a na nové vstupy reaguje podle toho, zdali tyto vstupy dané rysy obsahují nebo ne. Příkladem takového učení je klastrová analýza. Tato analýza se pokouší rozdělit data na části, takzvané klastry. Poté jsou data v rámci jednoho klastru analyzována podobně (podle jednoho nebo více kritérií), nýbrž data mezi jednotlivými klastry jsou analyzována různě.

5. Charakteristické učení

Tento typ učení podporuje nalezení reprezentací potřebných pro detekci nebo klasifikaci prvků v surových datech. Tím je nahrazena ručně programovaná konstrukce a stroj se dokáže sám naučit algoritmus a použít ho pro vyřešení konkrétní úlohy. Hlavní motivací pro využití pro toto učení je, že vstupy pro klasické strojové učení musí být dobře matematicky výpočetně zpracovatelné. Nicméně v reálném světě je většina dat (obrázky, videa, data ze snímačů) nedokáže algoritmicky definovat specifické vlastnosti. Možným řešením je pak nalezení významných rysů pomocí zkoumání bez použití explicitních algoritmů. Tento druh učení se snaží uchovávat informaci na svém vstupu a transformovat jí tak, aby byla lépe využitelná, často jako proces předzpracování. Tím je umožněno zpracování dat s neznámým rozdělením a dovoluje stroji se naučit na zpracování specifického úkolu. Charakteristické učení může být řízené nebo neřízené. U řízeného jsou tréninková vstupní data označena (názvem hledané věci vyskytující se v datech). U neřízeného jsou data neoznačena.

6. Rozptýlené slovníkové učení

Tato metoda se pokouší najít řídkou reprezentaci vstupních dat (řídké kódování), ve formě základních prvků a jejich kombinací. Tyto prvky se nazývají atomy a dohromady tvoří slovník. Pokud se prvek nachází ve slovníku, nemusí být ortogonální.[21]



Obr. 1.11: Vyčištění obrazu pomocí rozptýleného slovníkového učení [21]

7. Sebevzdělávání (Self learning)

Je to druh učení pracující bez jakýchkoli externích zásahů. Tím se tedy liší od zesíleného učení (odměny) a aktivního učení (zásahy učitele). Systém je řízen hlavně interakcí mezi poznáváním a emocemi. Samoučící algoritmus v každé iteraci aktualizuje paměť pomocí $W = ||W(A,S)||$. Algoritmus funguje následovně:

- (a) V situaci S proved akci A ;
- (b) Dochází k reakci a přijetí následků S' ;
- (c) Výpočet emoce na danou situaci $V(S')$;
- (d) Aktualizace paměti $W'(A,S) = W(A,S) + V(S')$

Jedná se o systém s jedním výstupem a jedním vstupem. Existují dva stavy, ve kterých tento typ učení může nacházet. První stav je stav inicializační kde získá počáteční informace o situacích s nimiž se může setkat ve druhém možném stavu. Druhým stavem, je stav kde se systém učí. V tomto stavu se setkává se situacemi žádoucími i nežádoucími.

8. Charakteristické učení (Feature learning)

Tato metoda se při předzpracování vstupních dat snaží o uchování informace a zároveň o jejich transformování na užitečná data. Touto technikou je možné rekonstruovat i vstupy, které mají neznámou distribuci, ale nemusí být věrohodné pro nepravděpodobné konfigurace dané distribuce. To nahrazuje nutnost manuálního vytváření funkcí a dovoluje tak stroji naučit se používat funkce k plnění konkrétních úkolů. Charakteristické učení je možné rozdělit na učení řízené a neřízené. U řízeného jsou zpracovávána označená vstupní data. Jedná se se hlavně o neuronové sítě, vícevrstvé perceptrony a slovníkové učení. U neřízeného učení jsou vstupní data neoznačená. Jsou to hlavně metody jako slovníkové učení, analýza nezávislých komponent, autoenkodéry, maticová faktorizace a další formy shlukování.

9. Detektor anomálií (Anomaly detection)

Při procesu dolování dat se vyskytují určité anomálie, které se dají popsat jako náhodná data významně lišící od ostatních dat. Tyto anomálie mohou následně způsobit nesprávné vyhodnocení. Algoritmus dokáže detekovat například narušení sítě z vnějšku. Existují tři hlavní metody pro detekci anomálií. První je detekce bez dozoru, kde dochází k detekci anomálií v neoznačeném prostoru. Princip je, že většina dat je normální a hledají se data, která se nejméně hodí ke zbytku dat. Další metodou je detekce anomálií pod dohledem. Pro tuto metodu je potřeba soubor dat označen jako normální a abnormální, který je použit pro naučení klasifikátoru. Poslední metodou je detekce anomálií s polo-
vičným dohledem. Tato metoda se snaží o vytvoření modelu z normálních dat a následně testují pravděpodobnost, že testovaná instance bude generována naučeným modelem. Detekce anomálií se často používá při detekcích narušení, detekce chyb, monitorování stavu systému. Také se používá při předzpracování pro odstranění anomálií z datové sady. Často dokáže zlepšit přesnost celkové analýzy dat.

10. Pravidla sdružování (Association rules)

Hlavním úkolem této metody je nacházení vztahů mezi proměnnými ve velkých databázích pomocí strojového učení založeného na pravidlech. Snaha je nacházet určitou míru zajímavosti, která určuje silná pravidla mezi daty. Strojové učení založené na pravidlech je jakékoli strojové učení, které nachází propojení mezi daty, vyvíjí pravidla pro ukládání, použití nebo manipulaci znalostí. Tyto metody strojového učení používají sady relačních pravidel, která představují znalosti zachycené systémem, na rozdíl od klasického strojového učení, které vytváří univerzální model. Přístupy strojového učení založené na pravidlech jsou například učení asociačních pravidel, systémy klasifikace učení a umělé imunitní systémy. Na základě konceptu silných pravidel, byl vytvořen algoritmus pro supermarket, který určoval pravděpodobnost zakoupení určitých výrobků na základě zakoupení jiných výrobků. Například pokud zákazník zakoupil cibuli, sýr a brambory, je vysoká pravděpodobnost, že také zakoupí maso na burgery. Toho lze využít při vytváření reklamy nebo rozložení regálů v obchodě nebo také různých slev atd.

11. Systémy klasifikace učení (Learning classifier systems):

Jsou skupinou algoritmů strojového učení založeného na pravidlech snažící se kombinovat nějaký genetický algoritmus s učícím algoritmem (řízené/neřízené učení nebo zesílené učení). Hlavním cílem je identifikování kontextově nezávislých pravidel, která jsou skupinově ukládána a pokouší se vytvářet předpovědi.

12. Induktivní logické programování (Inductive logic programming):

Je speciální přístup k učení pravidel za pomoci logického programování. ILP

system odvozuje hypotézu o logickém programu (zahrnující samé pozitivní příklady a žádné negativní) z kódování základních znalostí a několika příkladů prezentovaných jako logická databáze faktů. Induktivní programování zahrnuje jakoukoli metodu programování pro reprezentaci hypotéz, jako jsou funkční programy. [21]

1.5.1 Neuronové sítě

Základním stavebním kamenem neuronové sítě je neuron. Každý neuron má x vstupů, přičemž každý vstup má svojí váhu, která určuje jeho propustnost. Váhy mohou nabývat záporných hodnot, což určuje míru jejich zpomalení. Vnitřní potenciál j -tého neuronu y_{jin} je určen součtem jeho vstupů x , které jsou vážené pomocí vah w . [22]

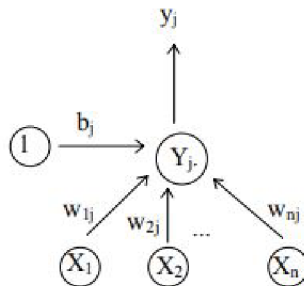
$$y_{jin} = \sum_{i=1}^n w_{ij}x_i \quad (1.14)$$

Bias, neboli aktivační práh neuronu, je přidán jako první do vektoru x na první místo. Následně je s biasem pracováno jako s další vahou. Celkový stup neuronu je tedy dán vztahem:

$$y_j = \sum_{i=1}^n w_{ij}x_i = w_{0j} + \sum_{i=1}^n x_iw_{ij} \quad (1.15)$$

kde w_{0j} je hodnota biasu:

$$w_{0j} = b_j \quad (1.16)$$



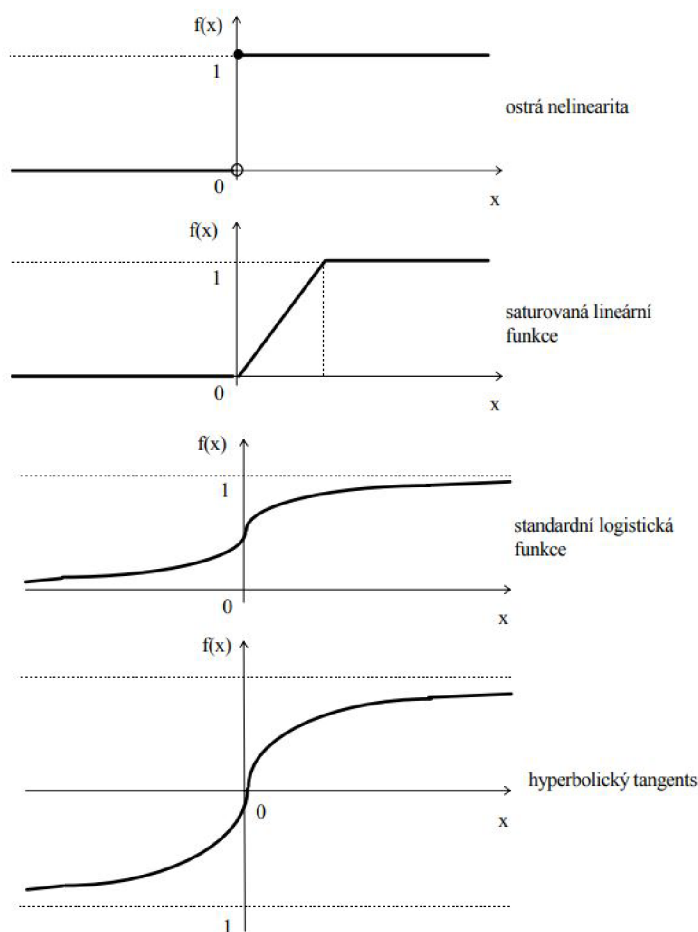
Obrázek 2: Formální neuron s biasem.

Obr. 1.12: Formální neuron s biasem [22]

Vnitřní potenciál neuronu y_{jin} indikuje výstupní stav neuronu y_j po dosažení hodnoty biasu b_j .

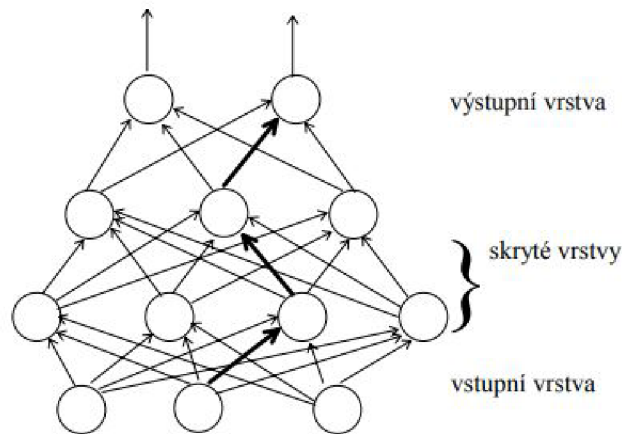
$$f(y_{jin}) = \begin{cases} 1, & \text{pokud } y_{jin} \geq 0 \\ 0, & \text{pokud } y_{jin} < 0 \end{cases} \quad (1.17)$$

Tato funkce je nejjednodušší aktivační funkcí (ostrá nelinearita). Dále existují další přenosové funkce:



Obr. 1.13: Příklad aktivačních funkcí [22]

Formální neurony propojené dohromady tvoří neuronovou síť. Jejich propojení je realizované tak, že výstup každého neuronu je propojen s jedním nebo více neurony další vrstvy. Právě množstvím a způsobem propojení neuronů je určena topologie sítě. Každá neuronová síť je rozdělena na vstupní vrstvu, pracovní (skryté) vrstvy a výstupní vrstvu. Mění se stav jednotlivých neuronů ovládá způsob a rychlost zpracování informace. Různá váha propojení neuronů pak upravuje konfiguraci sítě. Učení neuronové sítě probíhá měnícím se stavem neuronů a adaptací jednotlivých vah.[22]



Obr. 1.14: Třívrstvá neuronová síť [22]

Architekturu takovéto sítě je možné měnit pouze přidáním nebo odebráním neuronů v jednotlivých vrstvách, protože každý neuron je spojený se všemi neurony další vrstvy. Na obrázku je vidět třívrstvá neuronová síť (3-4-3-2), s vyznačenou cestou. [22]

1.5.2 Perceptrony

Jednoduchý perceptron

Perceptron je nejjednodušší neuronová síť s jedním neuronem. Tento neuron má typicky n vstupů a jeden výstup. Obor vstupních hodnot může být unipolární (0,1) nebo bipolární (-1,1). Každá vstupní větev navíc obsahuje váhu, která určuje její validitu.[22]

Vícevrstvé perceptrony

jedná se o jednoduchou neuronovou síť s jednou vstupní vrstvou, jednou výstupní a libovolný počet skrytých vrstev. Všechny neurony následující vrstvy jsou propojeny se všemi neurony vrstvy předchozí. Vstupní vrstva zpracovává vstupní signál, výstupní vrstva dělá rozhodnutí a predikuje vstupy. Například architektura s jednou skrytou vrstvou dokáže aproximovat jakoukoli spojitou funkci. často se používají při učení pod dohledem, kde se učí na párech dat vstup - výstup a nacházejí korelaci mezi nimi. Opět je součástí učení nastavování vah, úprava parametrů a biasu, pro co nejefektivnější potlačení chyby. Použití zpětné propagace (backpropagation) má za následek, že změny vah a biasu jsou relativní vůči chybě. Vícevrstvý perceptron se dá představit jako ping pong. Každý odpal míčku je odhad odpovědi (předpovědi) a každý návrat míčku je zpětná vazba o tom jak moc byl odhad chybný. V dopředném průchodu se signál postupně dostává ze vstupní vrstvy, přes skrytou vrstvu do

vrstvy výstupní a je porovnán se skutečnými pravdivými výstupy. Zpětný průchod používá zpětnou propagaci, řetězové pravidlo a parciální derivace chybové funkce, pro zpětné zpracování některých vah a biasu. Díky tomu je zjištěno, které parametry je nejspíše potřeba upravit. Díky tomu dochází ke konvergenci chyby.

1.5.3 Naive Bayes

Tato metoda je založena na bayesově theorému.

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)} \quad (1.18)$$

$P(A|B)$ určuje pravděpodobnost události A za předpokladu, že se uskutečnila událost B. $P(B|A)$ vyjadřuje pravděpodobnost události B za předpokladu, že se uskutečnila událost A. Dále $P(A)$ a $P(B)$ určují pravděpodobnosti událostí A a B. Na tabulce níže je vidět rozhodovací tabulka týkající se hraní golfu.[24]

	OUTLOOK	TEMPERATURE	HUMIDITY	WINDY	PLAY GOLF
0	Rainy	Hot	High	False	No
1	Rainy	Hot	High	True	No
2	Overcast	Hot	High	False	Yes
3	Sunny	Mild	High	False	Yes
4	Sunny	Cool	Normal	False	Yes

Obr. 1.15: Rozhodovací tabulka hraní golfu [24]

V tabulce je vidět klasifikace zda-li je den vhodný pro hraní golfu. Při propojení se vzorce na obr. 1.16 je událost A hraní golfu a vektor B jsou podmínky počasí kde:

$$B = (b_1, b_2, b_3, \dots, b_n)$$

Tato metoda se používá od 60 let 20. století a stále je oblíbenou metodou pro analyzování textu (příslušnost určitých dokumentů, detektor spamu atd.). Při správném předpracování dat je možné použít tuto metodu při automatických zdravotních diagnózách. Výhodou oproti jiným klasifikátorům je, že při výpočtu maximální pravděpodobnosti je možné provést hodnocení výrazu v uzavřené formě, což má lineární náročnost (spíše než zdlouhavou lineární aproximaci). Lineární náročnost znamená, že doba výpočtu roste lineárně s rostoucím množstvím vstupních dat.

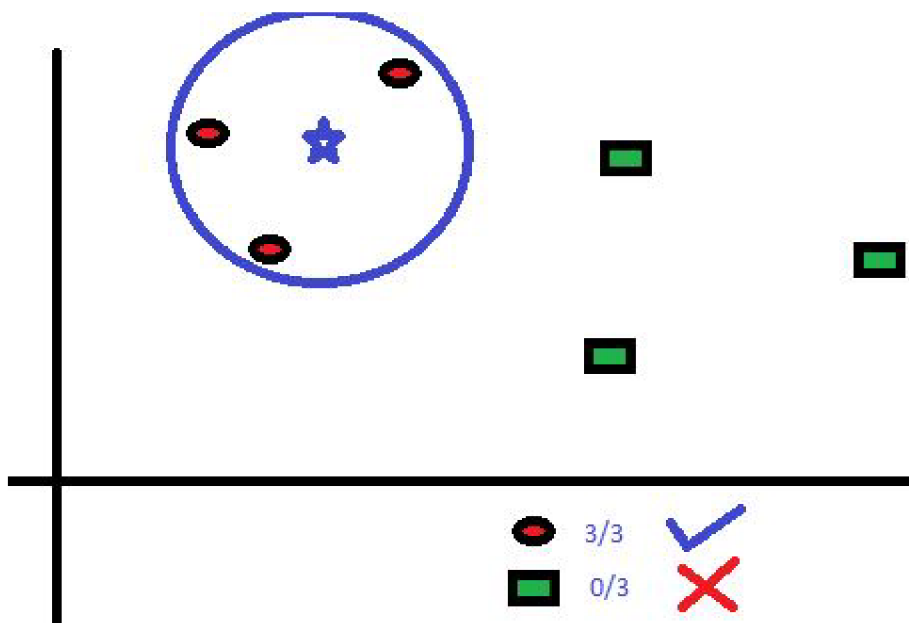
1.5.4 K-nejbližší sousedé

Tato neparametrická metoda se používá jak ke klasifikaci tak k regresi. Pro obě použití je klíčový parametr k , který určuje jak velké okolí upravovaného bodu je bráno v potaz.

- Při regresi je výstupní hodnota bodu vypočtena jako aritmetický průměr jeho k -nejbližších sousedů
- U klasifikace je bodu přiřazena hodnota nejčastěji se vyskytující v jeho k -okolí. Pokud je $k=1$ pak mu bude přiřazena hodnota nejbližšího souseda.

Dalším užitečným trikem může být váhování sousedů. To spočívá v přiřazení parametru $1/d$ každému sousedovi, přičemž d je vzdálenost od souseda.

Na obrázku je vidět použití algoritmus kde $k=3$. Je zde vidět, že modrá hvězda patří s vysokou pravděpodobností do skupiny červených koleček. Volba parametru k je velmi důležitá pro tento algoritmus.



Obr. 1.16: KNN algoritmus [25]

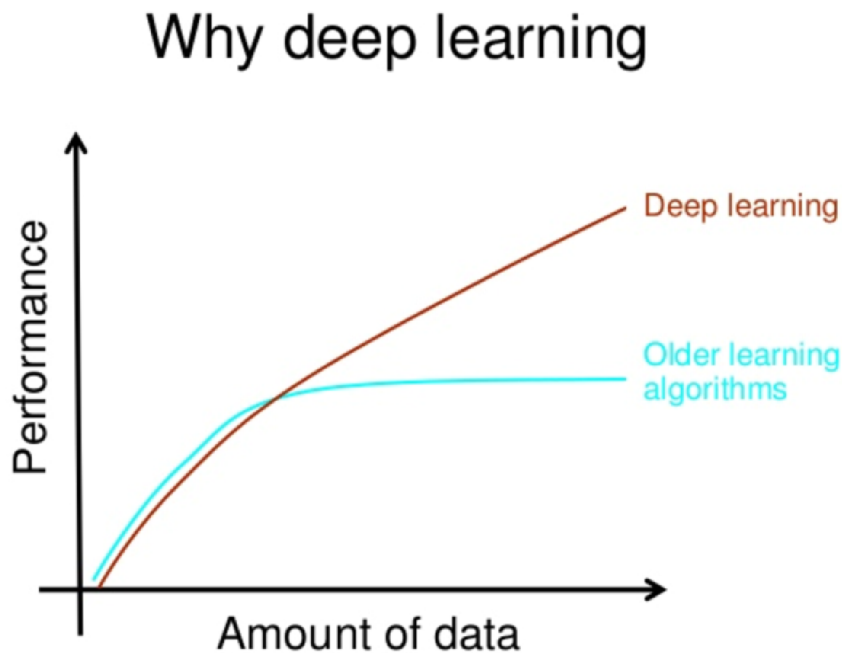
Parametr k se volí podle mnoha faktorů. Jedním z nich je například množství šumu. Větší hodnoty k redukuje dobře šum, ale činí hranice mezi třídami méně výrazné. Přesnost algoritmu může být poměrně dost snížena přítomností hlučných nebo irelevantních prvků nebo když se měřítka prvků výrazně liší od jejich významu. Někdy je také potřeba použít takzvanou redukci dat, která se používá u velkých datasetů. Pro přesnou klasifikaci jsou většinou potřeba jen některé datové typy.

Výhody a nevýhody algoritmu K-nejbližší sousedé (KNN)

- Výhody
 1. Jednoduchá implementace
 2. Jedná se o tzv. líný algoritmus učení a proto nevyžaduje žádnou fázi učení. Díky tomu je algoritmus podstatně rychlejší než třeba lineární regrese atd.
 3. Algoritmus nevyžaduje učení a tak mohou být nová data přidána bezproblémově.
 4. Vyžaduje pouze 2 parametry. Parametr k a funkci vzdálenosti (Euklidovská nebo Hamingova atd.).
- Nevýhody
 1. Nefunguje velmi dobře s více dimenzionálními daty. Pro algoritmus je obtížné vypočítat každé dimenze.
 2. U velkých souborů dat dochází k problému vysokých nákladů na výpočet vzdálenosti mezi novým a každým dalším bodem.
 3. Protože algoritmus těžko nachází vzdálenosti mezi dimenzemi s kategoričnými vlastnostmi, není pro tento typ dat vhodný.

1.5.5 Hluboké učení (Deep learning)

Hluboké učení patří do pole strojového učení. Algoritmus byl odvozen z metody umělých neuronových sítí. Jeho hlavním znakem je možnost učení na nestrukturovaných neoznačených datech. Tento algoritmus je charakteristický hledáním vzorců a podobností v datech a rozhodování na základě nich. Na obrázku 1.5.5 je vidět jak moc se liší rozdíl ve výkonnosti hlubokého učení a starších učících algoritmů v závislosti na narůstajícím množství dat. [33][32]



How do data science techniques scale with amount of data?

Obr. 1.17: Přednost hlubokého učení na velkém množství dat [32]

Hluboké učení se vyvíjelo spolu s digitální dobou. V dnešní době jsou ve všech odvětvích sbírána obrovská množství dat (big data), která je velmi složitá a výpočetně náročná zpracovávat. Tato data jsou převážně nestrukturovaná a lidem by trvalo roky z nich extrahovat důležité informace. Firmy si uvědomují potenciál skrytý v těchto datech a stále více se snaží investovat do systému umělé inteligence pro automatizaci sběru a vyhodnocení dat. Nejmodernější metodou hlubokého učení jsou konvoluční neuronové sítě. [33][32]

Konvoluční neuronové síť (Convolutional neural network, KNS, CNN)

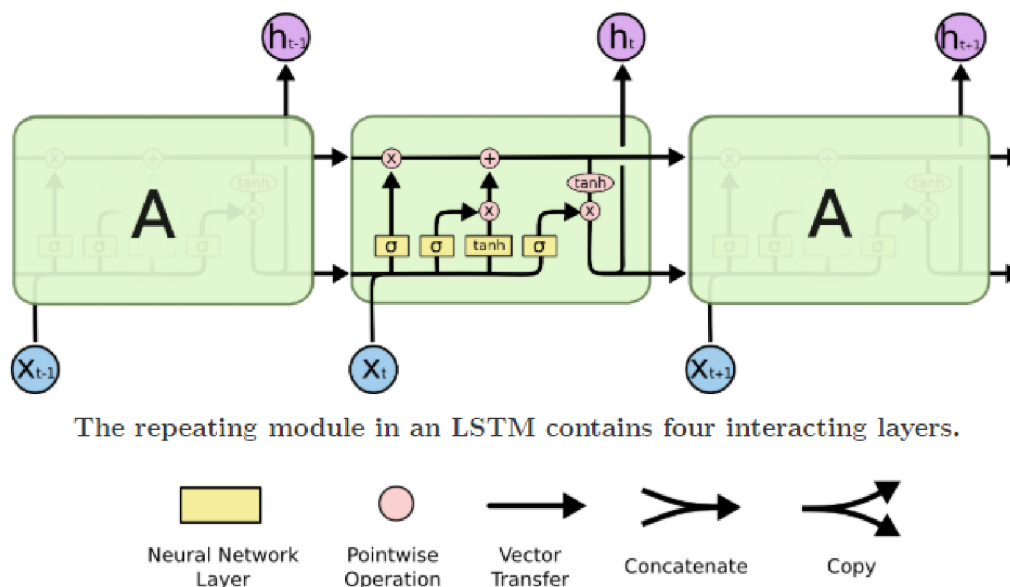
Tato metoda se velmi často používá při zpracování obrazu. Jedná se o vícevrstvou neuronovou síť, která má vrstvy plně propojeny (každý neuron je propojen se všemi neurony následující vrstvy). Díky tomuto propojení jsou tyto sítě náchylné na přeučení. Metoda využívá hierarchický vzorec v datech a pomocí jednodušších vzorců se snaží sestavit složitější vzorce. Konvoluční sítě jsou inspirovány biologickými procesy, protože vzorec propojení neuronů připomíná organizaci lidských neuronů. KNS používá poměrně málo předzpracování oproti ostatním klasifikačním algoritmům. Síť si sama vytvoří filtry, které je nutně v jiných algoritmech vytvořit ručně na základě apriorní znalosti nebo předchozích výsledků. Tato samostatnost je hlavní výhodou KNS. Princip spočívá v procházení vstupního obrazů těmito filtry a hledání vzorců. Parametry filtrů jsou velikost, krok procházení a hloubka filtru (musí odpovídat hloubce vstupního obrazu). Data jsou pak předávána do dalších vrstev, například složitějších filtrů a vyhodnocována. U učení KNS je kritické množství vstupních dat. Pokud je vyhodnocována proměnlivá třída (zvíře, člověk) je důležité mít vhodný dataset obsahující velké množství různých fotografií. [34]

1.5.6 Neuronová síť typu LSTM

V této práci je použita neuronová síť typu sequence-to-sequence LSTM network, neboli síť typu sekvence k sekvenci, dlouhá krátkodobá paměť. LSTM je umělá rekurentní neuronová síť (RNN). Nejedná se o obyčejnou dopřednou architekturu, ale obsahuje zpětnovazební propojení. Síť je schopna zpracovávat jak obrázky tak sekvence dat. Jejím hlavním úkolem je rozpoznávat písmo, hlas nebo jiné intervaly opakující se v datových sekvencích. To je pro potřeby této práce vyhovující. Síť si při učení dokáže zapamatovat informace z předchozích iterací a ty pak využít ke klasifikaci dalších vzorků.

LSTM sítě jsou používány pro vyhnutí se problémům se závislostmi v datech v širším časovém rozmezí. Dlouhodobé zapamatování informace není věc, kterou se snaží naučit, ale jejich vlastnost. Všechny rekurentní neuronové sítě se skládají z řetězce propojených modulů. V jednodušších sítích jsou tyto moduly složeny například jen z funkce tanh nebo jako na obrázku 1.18 se skládají ze speciální struktury.[36]

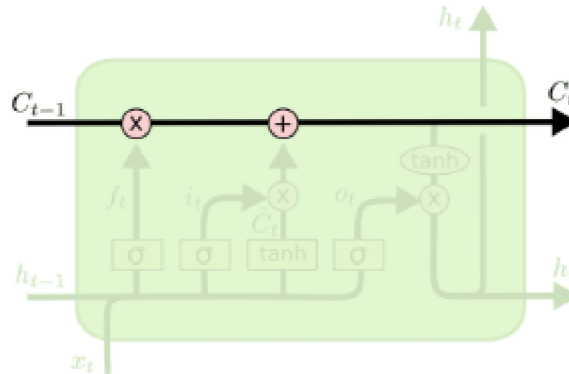
1.5.7 Struktura a vysvětlení LSTM sítě



Obr. 1.18: Ukázka LSTM sítě[35]

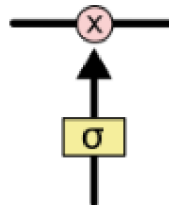
Na obrázku jsou vidět jednotlivé šipky z nichž každá přenáší celý vektor dat. V různých oblastech pak dochází k různým vektorovým operacím. Ve žlutých oblastech se nachází naučené vrstvy neuronové sítě. Spojené čáry naznačují zřetězení a naopak rozdělující se čáry znamenají kopírování vektoru na více míst.

Na obrázku 1.19 je vidět hlavní větev probíhající v horní části diagramu. Tato větev vyjadřuje stav celé buňky. Do této větve jsou přidány pouze jednoduché lineární operace pro jednodušší průchod informace. [35][36]



Obr. 1.19: LSTM hlavní větev[35]

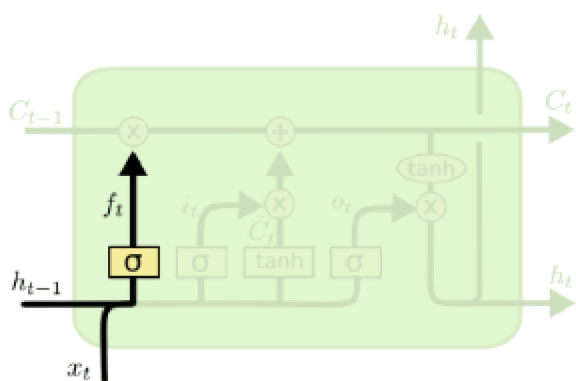
Pomocí bran (gates), jako je na obrázku 1.20, je opatrně přidávána nebo odebrána informace. Brány se skládají z operace lineárního násobení a sigmoidní vrstvy neuronové sítě. Výstup z této vrstvy je mezi 1 a 0 a popisuje kolik procent informace má být propuštěno. Hodnota nula znamená nic a jedna znamená všechno. V buňce se nachází tři takové brány pro její ochranu a kontrolu stavu.[35][36]



Obr. 1.20: LSTM brána[35]

1.5.8 Jednotlivé kroky LSTM sítě

V prvním kroku dochází k vybrání informace, která bude zapomenuta. O to se stará vrstva se zapomínající bránou (forget gate layer). Vrstva se podívá na h_{t-1} a x_t a na výstup je posláno číslo mezi 1 a 0, které pak ovlivní informaci z předchozí buňky C_{t-1} . V předchozí buňce se může nacházet užitečná informace pro aktuální buňku a nebo naopak informace úplně zbytečná.[35][36]

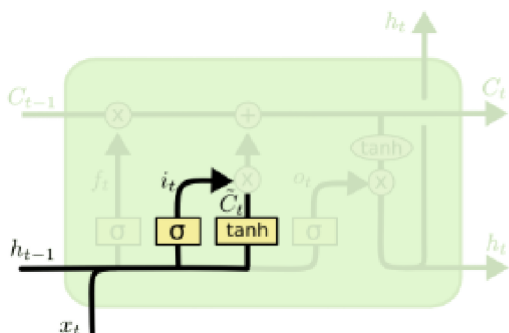


$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$$

Obr. 1.21: LSTM krok 1[35]

Druhým krokem je pak rozhodnutí o způsobu uložení nové informace v aktuální buňce. Tento krok se skládá ze dvou částí. První z nich je sigmoid vstupní vrstva, která má za úkol rozhodnout jakou hodnotu budeme aktualizovat. Druhá část je vrstva tanh, která vytvoří vektor nových hodnot, které jsou přidány do stavu buňky. Poté jsou výsledky těchto dvou částí spojeny do jednoho a je aktualizován stav buňky.

Když je nějaká část informace zapomenuta, je nutné jí nahradit novou informací.[35][36]

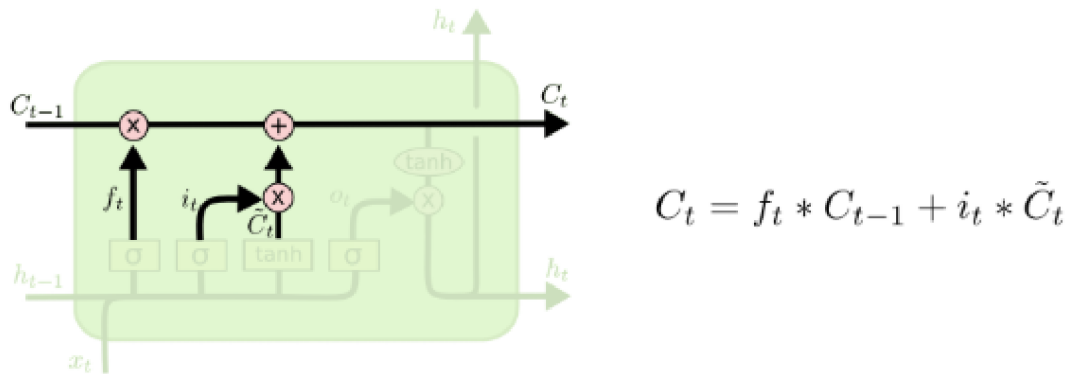


$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$$

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C)$$

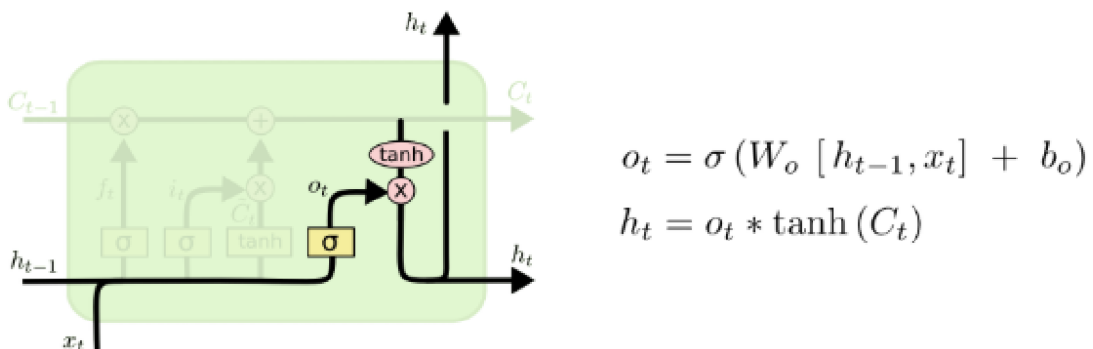
Obr. 1.22: LSTM krok 2[35]

V třetím kroku (1.24) je vidět skutečné aktualizování stavu buňky C_{t-1} na stav aktuální C_t . Data pro aktualizaci byla vybrána v předchozích krocích a nyní dojde k aktualizaci. Pomocí vynásobení C_{t-1} s f_t dojde k zapomenutí již nepotřebné informace. Poté násobením i_t s \tilde{C}_t dojde k filtrování nové informace podle toho jak moc je potřeba aktualizovat stav. V tomto kroku dochází k nahrazení nepotřebné informace novou informací. [35][36]



Obr. 1.23: LSTM krok 3[35]

V posledním kroku dochází ke stanovení výstupu. Výstup je založen na stavu buňky, ale bude změněn. Na začátku prochází informace přes sigmoid vrstvu, která rozhodne jaká část stavu buňky bude zobrazena na výstupu. Pomocí funkce tanh jsou hodnoty normalizovány mezi 1 a -1 a pak násobením s výstupem z funkce sigmoid jsou na výstup poslány pouze relevantní hodnoty. Díky tomu jsou na výstupu pouze potřebné informace pro predikci. Nepoužitelné informace jsou zapomenuty.[35][36]



Obr. 1.24: LSTM krok 4[35]

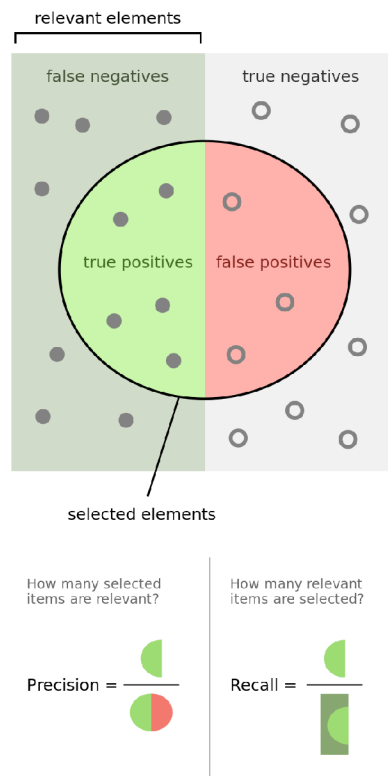
V praxi se používá více variant této metody, ale ty nejsou v této práci použity.

1.5.9 Popis výsledků neuronové sítě

Prvním parametrem je přesnost (accuracy), která v procentech vyjadřuje kolik predikovaných hodnot odpovídá skutečným hodnotám. Počítá se jednoduše z predikovaných parametrů a známých skutečných hodnot.

Druhým parametrem je určitost (precision), která vyjadřuje kolik vybraných vzorků je relevantních. Jedná se o zlomek relevantních instancí oproti všem získaným instancím. Detailní vysvětlení se nachází na obrázku 1.25.

Třetím parametrem je schopnost vybavení si (recall), který vyjadřuje kolik relevantních vzorků bylo vybráno. Tento parametr je znám také jako citlivost. Jedná se o zlomek celkového množství relevantních instancí, které byly skutečně získány. Podrobnější vysvětlení se nachází na obrázku 1.25.



Obr. 1.25: Vysvětlení parametrů precision a recall [37]

Posledním a nejspíše nejdůležitějším parametrem je F1 skóre. Tento parametr vyjadřuje přesnost testu vzhledem k parametrům recall a precision.

Výpočet všech těchto parametrů je vidět na obrázku 1.19, 1.20 a 1.21.

$$Precision = \frac{TP}{TP + FP} \tag{1.19}$$

$$Recall = \frac{TP}{TP + FN} \quad (1.20)$$

$$F1 = 2 * \frac{Precision * Recall}{Precision + Recall} \quad (1.21)$$

2 Popis výrobní linky a sběru dat

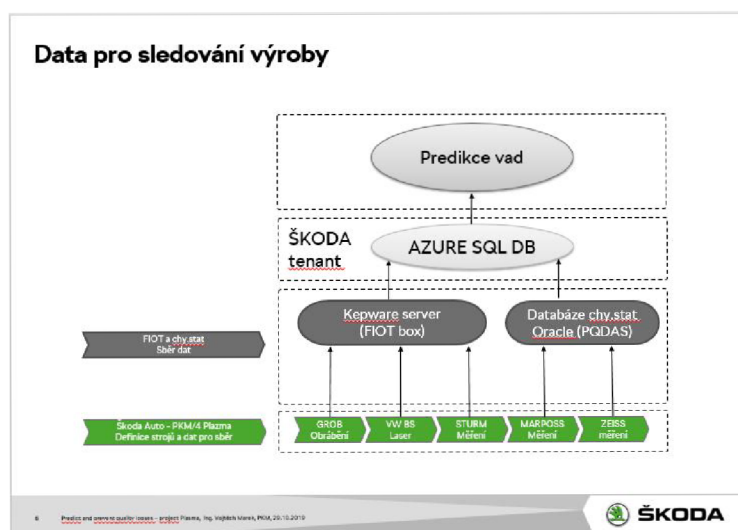
Výrobní linka realizuje sérii operací, jejichž cílem je výroba finálního kusu motorového bloku. jednotlivé operace jsou pak vypsány v kapitole 2.1.

2.1 Jednotlivé operace na výrobní lince

1. Grob - Obrábění
2. EcoClean - Praní
3. VW Braunschweig - Laser
4. Sturm - Měření
5. VW Braunschweig - Plazma
6. Sturm - Měření
7. EcoClean - Praní
8. Grob - Obrábění
9. Gehring - Honování
10. EcoClean - Praní
11. VW Braunschweig - Vířivé proudy
12. VW Braunschweig - Výstupní kontrola
13. Aktivit - Paletizace

Sběr dat z linky

Na následujícím obrázku je vidět způsob ukládání dat ze strojů na server. Při výrobě dochází ke sběru velkého množství dat. Tyto data se nachází v příloze A.



Obr. 2.1: Způsob ukládání dat z výroby

2.1.1 Rozbor systému diagnostiky

V této kapitole se nachází popis diagnostiky při výrobě motorového bloku ve Škoda auto a.s., systém ověřování hotových motorových bloků a sledování stavu výrobních strojů a jejich kondice. V posledních letech celosvětově roste množství použití hliníku při výrobě bloků válců motoru. Hlavní výhodou jsou náklady na výrobu z důvodu menší spotřeby energie. Předpokládá se, že v roce 2020 budou alespoň 2/3 motorů vyrobeny z hliníku. Ve Škoda Auto a.s. se dochází při výrobním procesu k mechanickému zdrsnění a plazmování (potahování).



Obr. 2.2: Blok válců[27]

Hlavní výhody tepelně potažených otvorů válce oproti litým vložkám z litiny

- Značné snížení váhy
- V sériové výrobě se snižují náklady
- Zlepšení přenosu tepla
- Snížení spotřeby oleje
- Extrémně robustní povrch válce
- Odolné oproti moderním agresivním palivům
- Nízké tření a vysoká odolnost vůči mechanickému opotřebení

Vlivy na kvalitu a náklady výroby

Vlivů na kvalitu je velké množství, mezi hlavní patří:

- Design
 - Materiál, šířka stěny
 - Design honovacího vybavení, zkosení ...
- Slévárna
 - Kvalita lití (Dutiny a Poréznosti)
- Předběžné obrábění
 - Řetězec tolerančních mezí
 - Vlhký vs. MQL (Minimum quantity lubrication)
 - Výrobní strategie
- Aktivace
 - Procesy
- Proces potahování
 - Předčištění
 - Druh povlaku
 - Minimalizace povrchových úprav
 - Šířka (Tloušťka)
- Dodatečné zpracování
 - Přestříkání a čištění
 - Chlazení
- Dokončení
 - Řetězec tolerančních mezí
 - Charakteristické vlastnosti povrchu
 - Výrobní strategie a procesy

celý přehled je možné najít v popisu na stránce [27].

Proces tepelného potahování bloků válců

Tento proces se skládá z šesti kroků:

- Předzpracování - Hrubé obrábění
- Jemné vrtání
- Mechanické zdrsňení
- Potahování
- Dokončovací proces
- Jemné vrtání a honování[27]

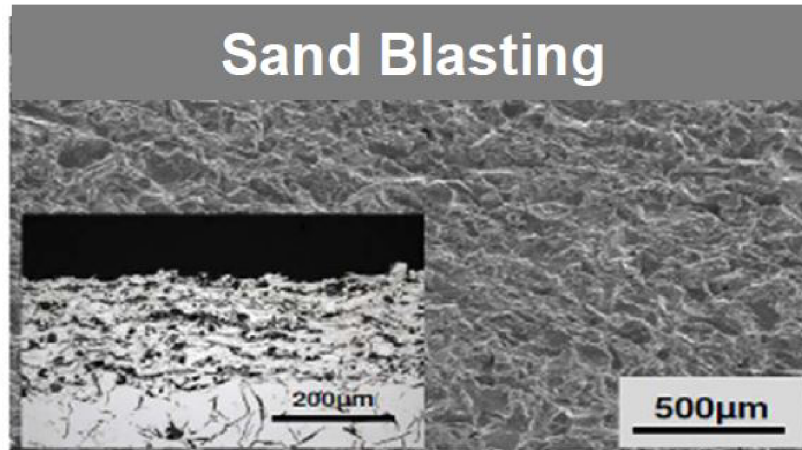
Každý krok výrobního procesu vyžaduje konkrétní znalosti technologie. Veškeré interakce mezi jednotlivými procesy musí být přesně sladěny. Specifikace a konkrétní nastavení jednotlivých procesů je know-how firmy a není veřejné. Pouze správná kombinace těchto procesů vede k prvotřídní kvalitě a ekonomickému procesu.[27]



Obr. 2.3: Detail linky[27]

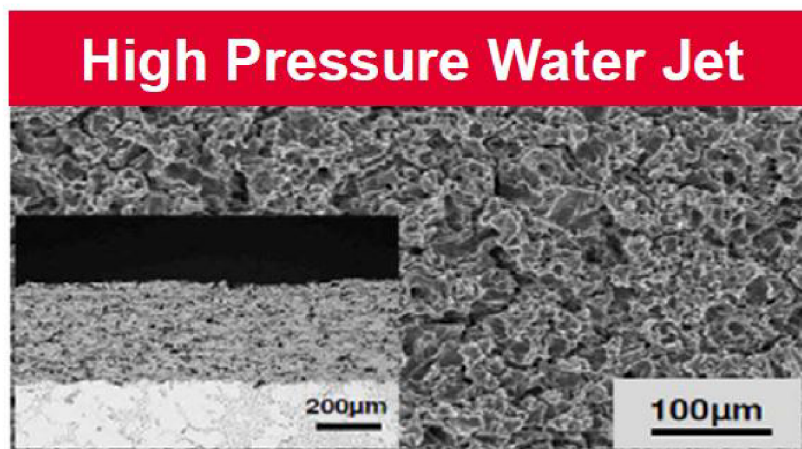
Proces zdrsňení povrchu

- Pískování povrchu často není schváleno pro vysokoobjemovou automobilovou výrobu.



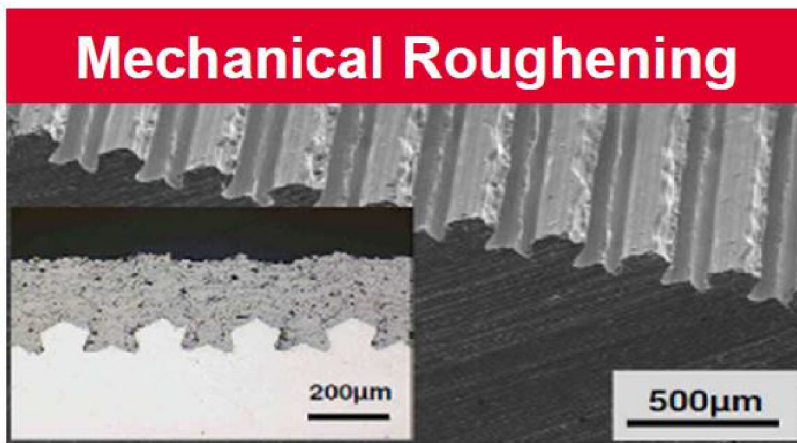
Obr. 2.4: Zdrsňení pískováním [27]

- Zdrsňování pomocí vysokotlakého proudu vody je spolehlivé řešení, ale také velmi drahé.



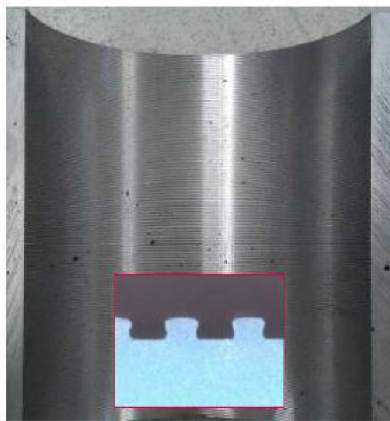
Obr. 2.5: Zdrsňení vysokotlakou vodou [27]

- Mechanické zdrsňování je ekonomicky výhodné a je možné použít různé výrobní procesy. Tento typ je použit na výrobní lince ve Škoda Auto a.s.



Obr. 2.6: Zdrsňení mechanické [27]

Po mechanickém zdrsňení se na povrchu bloku válců vytvoří pravidelné zuby (viz. obr 27.). Pro optimální ekonomické řešení je nejlepší použít MQL.



Obr. 2.7: Detail povrchu [27]

Plazmování

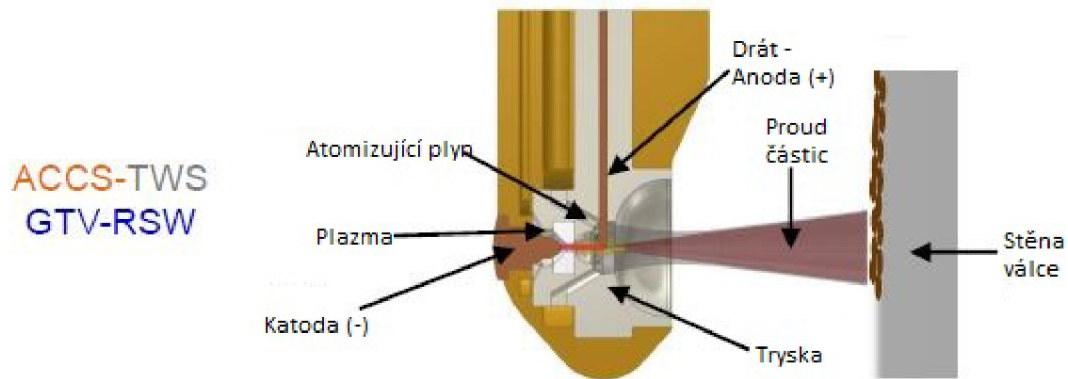
Tento proces spočívá v nanášení povlaku na vnitřní povrch bloku válců motoru. Nejčastěji se používá právě pro potahování otvorů pro válce v automobilovém průmyslu. Tato technika pak umožňuje použití hliníkových bloků místo těžkých litinových kusů. Jako materiál je použit jeden vodivý drát. Plazmový nadzvukový paprsek nejdříve drát roztaví, atomizuje jej a následně nahání na substrát. Paprsek plazmy je tvořen obloukem mezi katodou a drátem (anoda). Po atomizování nanáší vytvořený plyn proud roztavených kapiček na stěnu válce. Částice mají velkou kinetickou energii a tak se po dopadu na povrch zplošťují. Po dopadu velmi rychle tuhnou a vytváří krystalickou amorfní fázi. Je možné vytvářet vícevrstvé povrchy. Ty pak tvoří vysoce odolný povlak. Jako materiál je možné použít všechny vodivé dráty do průměru 1.6mm. Plazmování lze použít pro nanášení povlaku na opotřebitelnou plochu součástí motoru nebo převodovky, čímž se nahradí pouzdro nebo ložisko. Pokrytí ochranou vrstvou nabízí velkou řadu výhod jako snížení hmotnosti, nákladů, tření a napětí v ojnicích.

ACCS Pokročilý systém pro plazmování bloků válců

Vlastnosti systému jsou:

- Integrovaný systém kontroly bloků válců
 - Nakládání a vykládání kusů probíhá zároveň při měření
 - Rychlá výměna nástrojů
 - Ruční a automatické ovládání se senzorem vzduchové mezery (pojistka)
 - Možnost vkládání materiálu ručně, automaticky (robotem nebo portálem)
1. Povrchová úprava (potahování)
 - Nejnovější generace systému APS potahování
 - Super rychlá výměna hotového kusu za další
 - Maskování bloků a vyprázdnění s poloautomatickou výměnou sprejovacího kolektoru
 - Velmi účinný odsávací systém nahoře i dole
 - Automatický systém čištění hořáku pomocí CO_2
 - Průhledný systém výfuku pro možnost vizuální kontroly proudu částic
 - Stlačený vzduch nebo dusík je možné použít jako atomizační plyn
 2. Řízení dodávky a systém výfuků
 - TWS: Přívod vlánka ze sudu (max.600kg) nebo z vřetene (15kg)
 - TPS: Dvojitě napajení pro zajištění kontinuálního potahování
 - TWS/TPS: Obsahuje všechny kontrolní moduly, automatizovaný 3-kanálový výfukový systém, centrální rozhraní pro všechna možná požadovaná média

Způsob nanášení TWS:

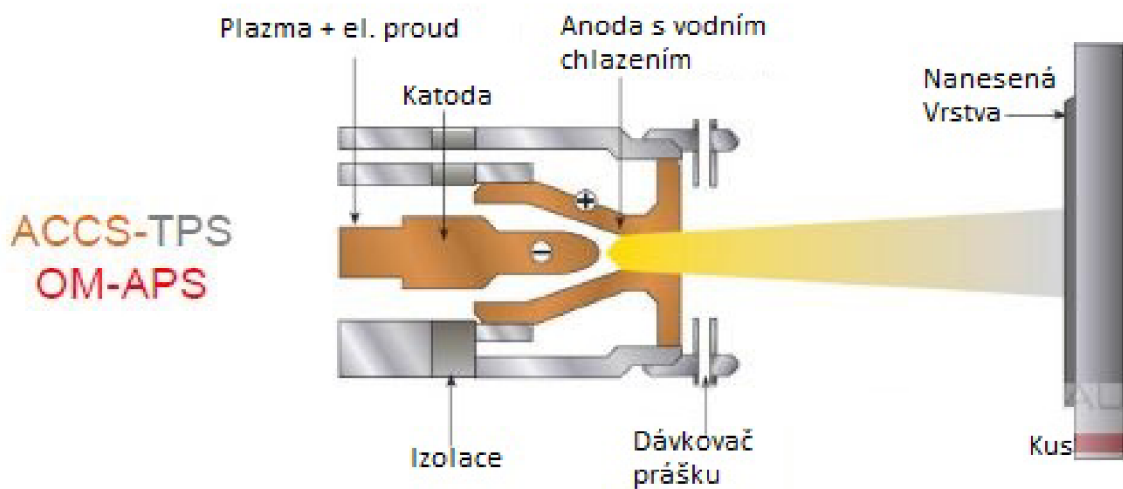


Obr. 2.8: Detail plazmování bloku válců TWS [27]

Popis:

- Atomizující plyn: Při atomizaci plynu je proud tekutého kovu nanášen na stěnu bloku válců, pomocí plynu o vysoké rychlosti (vzduch, dusík, argon, helium)

Způsob nanášení TPS:



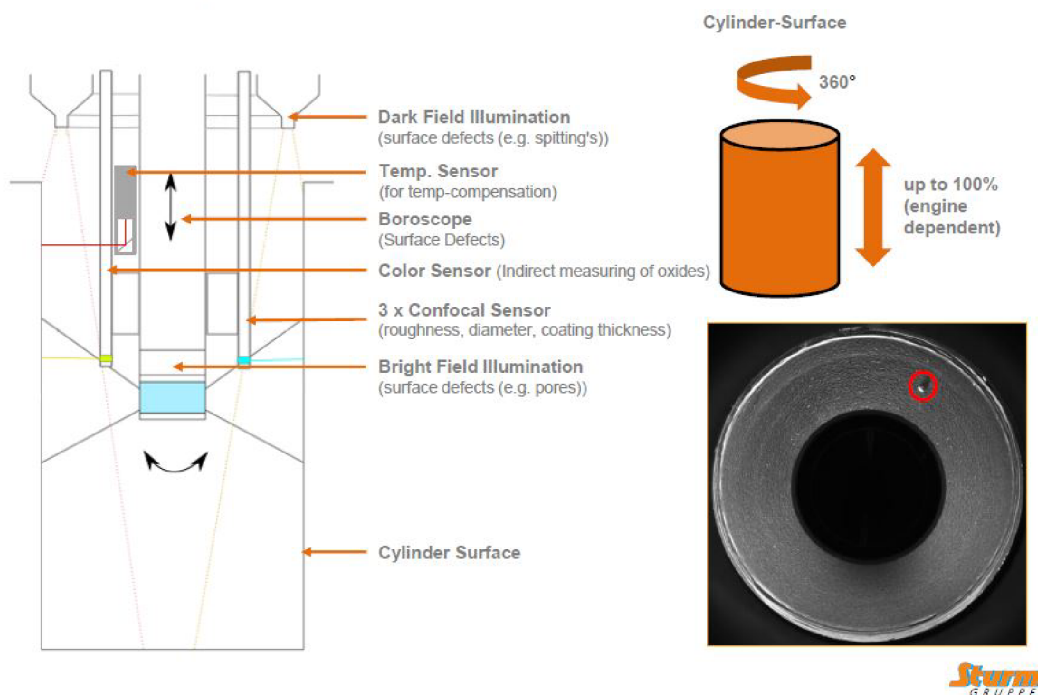
Obr. 2.9: Detail nanášení plazmy na blok válců TPS [27]

Schéma finální kontroly bloku válců

V následující kapitole je ukázán systém kontroly finálních bloků motoru. Na obrázku je vidět kde se nachází jednotlivé senzory a jaké vady jsou hledané. Na obrázku je také vidět příklad možné vady. Mezi procesem obrábění a finální kontrolou mohou vznikat delší prodlevy, ve kterých může vlivem různých problémů, docházet k neustálému vyrábění vadných kusů. Právě tomu se snaží metody popsané v této práci zamezit a odhalit vadný kus na základě dat ze strojů již při jeho vyrobení. Tím dojde ke snížení počtu vadných kusů a tím i nákladů na výrobu.

ACCS-CI

Schematic set-up



Obr. 2.10: Podrobnosti inspekce bloku motor[27]

- Dark Field Illumination (surface defects (e.g. spitting's)): Osvětlení tmavých částí, nalezení povrchových vad (např. plivanců)
- Boroscope (Surface Defects): Boroskop (povrchové vady)
- Color Sensor (Indirect measuring of oxides): Barevný senzor (nepřímé měření oxidů)
- 3 x Confocal Sensor (roughness, diameter, coating thickness): 3x konfokální senzor (drsnot, průměr, šířka potahu)
- Bright Field Illumination (surface defects (e.g. pores)): Osvětlení světlých míst (detekce povrchových vad jako pórů atd.)
- Cylinder Surface: Povrch bloku válců

2.2 Popis dat linky plazmy

Prvním druhem jsou data vázaná na každý vyráběný kus a uložena na RFID čipu svázaného s konkrétním kusem. Možnost využití těchto dat spočívá v tvorbě modelu pro následnou predikci. Každý stroj na každé operaci zapisuje na daný čip veškerá data týkající se daného kroku. To znamená nejenom data o výrobku, jako například jeho rozměry a status (OK/NOK), ale i provozní data stroje. Jedná se o informace o stroji jako takovém a zároveň o data ze snímačů z doby obrábění daného kusu (např. teplota, tlak atd.). Zároveň jsou k dispozici informace o době trvání jednotlivých úkonů v daném stroji. V této kapitole se nachází popis jednotlivých dostupných dat z linky plazmy včetně jejich zdrojů.

Dalším zdrojem dat je ukládání dat z výrobních strojů na server. Jedná se o data vytipovaná odborníky ve Škoda Auto a.s. Tato data obsahují převážně data o strojích, například stav stroje a počty vyrobených kusů, počet OK a NOK kusů. Tato data by bylo možné zčásti také použít pro tvorbu modelu. V těchto datech se snadněji hledají vzorce, protože jsou na server odlévána v reálném čase. Pokud by bylo k dispozici dostatečné množství dat, bylo by možné je analyzovat pomocí různých algoritmů a najít spojitosti s daty ze stroje a výslednou kvalitou kusu. Hlavním využitím těchto dat je jejich následné sledování a predikce vad a poruch. Pokud bude správně navrhnut a vytvořen model, je možné sledovat tato data v reálném čase a z jejich průběhu a především nenadálých změn předpovídat poruchy a nebezpečí výroby NOK kusů.

2.2.1 Důležitá data z linky plazmy

Zde se nachází seznam dat, která nejčastěji ukazují nedostatečnou kvalitu kusu a také veličiny, které tyto chyby způsobují. Například otáčky laseru za minutu mohou způsobit chybnou drsnost válce.

1. Grob - obrábění

1 Data ze strojů

- Životnost nástroje D (kusy)
- Životnost nástroje Rz (kusy)
- Otáčky za minutu (3500 RPM)
- Posuv (1050 mm/min)
- Tlak emulze (20 Bar)
- Životnost ostatních nástrojů (kusy)

2 Data z měření

- Drsnost válce (10-20 μm)

- Pozice válce (mm)
 - Průměr válce ($74,16 \pm 0,1$)
 - Průměr indexu (14 H7)
 - Průměr indexu (12 H7)
2. VW Braunschweig - Laser (Data ze strojů)
- 1 Životnost optiky (2000 kusů)
 - 2 Rychlost spotřeby (28 m/s)
 - 3 Posuv (681,8 mm/min)
 - 4 Výkon laseru (1000W)
 - 5 Otáčky za minutu (6136 RPM)
3. Sturm - (Data z měření)
- Drsnost (30-80 μm)
 - Linky - přesnost (10 linek/1mm)
 - Průměr válce (mm)
 - Poréznost (množství)

2.2.2 Rozbor datových bloků (RFID)

Zde se nachází data o kusu bloku válců (Zylinderkurbelgehäuse - ZKG) Ta se nachází na RFID unikátním čipu, který má každý vyráběný kus na lince. Data jsou sbírána s periodou vzorkování 1 vteřina. Sběr dat probíhá kontinuálně v průběhu obrábění kusu na daném přístroji. Po obrobení kusu jsou data, která se nasbírala při obrábění, zapsána na RFID čip. Konkrétní popis těchto dat se nachází v příloze A.

2.2.3 Finální vybraná data pro prediktivní analýzu

Parametry v této kapitole byly vybrány jako důležité na základě konzultace se zkušenými odborníky. Tyto veličiny nejvíce ovlivňují kvalitu výsledného kusu. Není nutné, aby byla všechna z nich použita pro predikci. Pokud se při zpracovávání diplomové práce ukáže, že některá data jsou nepodstatná, budou z analýzy vyřazena. Bude se dle dohody jednat o sběr surových dat přes Kepware a DataLogger do DB MS SQL (vytvořena bude ŠKODA SQL adminy). Sběr dat probíhá s periodou vzorkování 1 vteřina. Tato data jsou přístupná v reálném čase, takže je možné je použít pro prediktivní analýzu v reálném čase. Tabulka obsahující všechny vybrané parametry se nachází v příloze B. V této práci budou z důvodu rozsahu analyzována pouze data teploty a rychlosti otáček (RPM).

3 Praktická část diplomové práce

3.1 Úvod

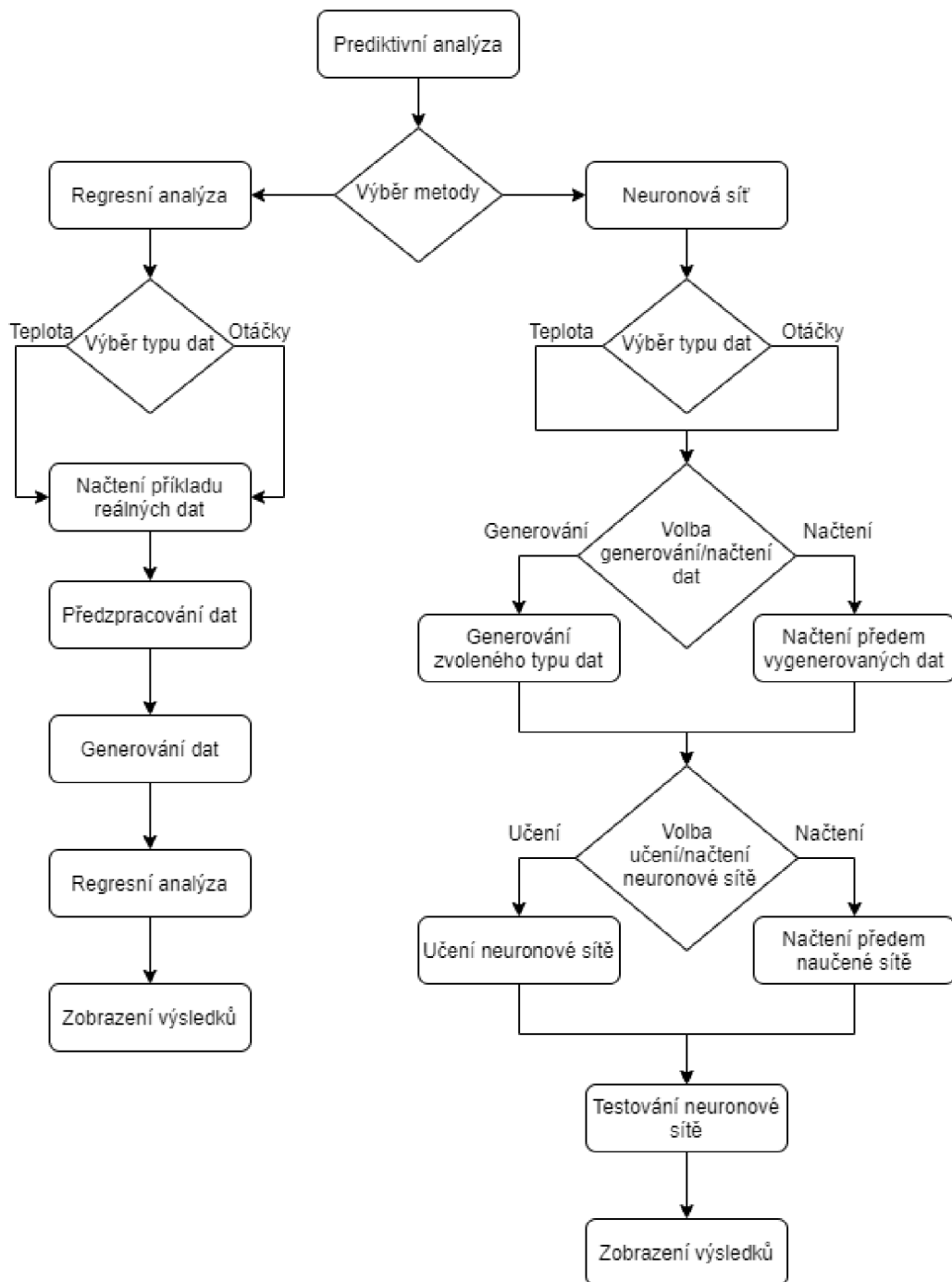
Cílem praktické části diplomové práce je analýza dat z výrobní linky ve Škoda auto a.s. Při výrobě je obrábění kusu a jeho detailní měření uskutečňováno na několika strojích. To způsobuje vznik dopravního zpoždění mezi těmito operacemi. Pokud dojde k vyrobení vadného kusu, respektive rozladění nebo poruše obráběcího stroje, vzniká dopravní zpoždění mezi chybou a zjištěním chyby při měření tohoto vadného kusu. Hlavním cílem této práce tedy je odhalení této poruchy hned při jejím vzniku analýzou dat z obráběcího stroje a okamžitému zastavení výroby a tím zamezení ztrátám.

Aktuální situace ve světě způsobená virem COVID-19, bohužel zapříčinila i uzavření výrobního závodu a tím i zpoždění prací při sestavování dané výrobní linky. Výrobní linku se bohužel nepodařilo připravit v potřebném termínu, proto se v projektu místo reálných dat nachází několik generátorů, které náhodně generují data. K dispozici byly pouze příklady, jak by měla finální data vypadat, ručně nasbíraná přímo ze strojů. Tato data se nachází v xlsx souborech přiložených k práci. Načtené hodnoty jsou použity jako základ generátorů dat. Jedná se například o náhodné dogenerování dat na základě skutečných dat nebo o opakované vybírání částí dat a jejich zamíchání a následné uložení. Dané generátory tedy vytvářejí dostatečně velké datasety a také uměle generují chyby v datech. Nutnost generovat chyby vzniká, protože při ručním sběru dat neprobíhala výroba. Ve finále tyto datasety simulují nepřetržitý tok dat z výrobních strojů, ve kterých potom metody hledají různé výkyvy a nesrovnalosti. Blokovaný diagram programu se nachází na obrázku 3.1.

Praktická část diplomové práce obsahuje dvě metody. První metodou je metoda regresní analýzy. Ta je teoreticky probírána v kapitole 1.4.1 a její praktická aplikace na data v kapitole 3.3. Cílem této metody je vytvoření nástroje, který na základě vstupních dat dokáže nalézt určité kritické prahy parametrů. Poté lze pomocí těchto parametrů sledovat v reálném čase vývoj veličin ve stroji a po překročení daných prahů o určité procento nahlásit obsluze stroje varování.

V druhé části se nachází analýza dat pomocí neuronové sítě, která je podrobně probírána v kapitole 1.5.1 teoreticky a v kapitole 1.5.6. prakticky. Pro každou vybranou veličinu (teplota, otáčky) existuje jedna neuronová síť, která je naučená na odhalování anomálií v datech. Hlavním cílem této metody je odhalení intervalů, které regresní analýza nedokáže odhalit.

Na konci se bude nacházet porovnání obou metod vzhledem k úspěšnosti odhalení chyb a výpočetní náročnosti.

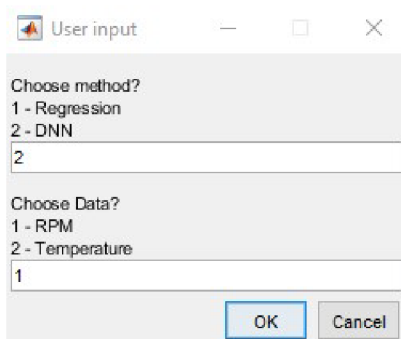


Obr. 3.1: Diagram popisující běh programu

Analýza dat byla realizována na stolním PC s grafickou kartou NVidia 980 GTX, 16gb RAM paměti (2800Mhz) a procesorem Intel Core i7 6700K. Jako úložiště byl použit Samsung 970 EVO NVMe M.2 SSD 500 GB s čtecí rychlostí až 3400 MB/-sec a zapisovací rychlostí až 2300 MB/sec. Hlavním softwarem pro zpracování byl Matlab 2020a. Základem pro metodu regresní analýzy byl toolbox Curve fitting. Pro trénování neuronové sítě pak byl použit toolbox trainNetwork.

3.2 Hlavní funkce (main.c)

Úkolem hlavní funkce main.c je ukládat volby uživatele a podle nich volit a spouštět další funkce. Po spuštění skriptu se objeví okno s prvními otázkami. Prvním krokem je zvolení metody, kterou chce uživatel zobrazit. Jsou k dispozici dvě volby, regresní analýza (Regression) a neuronová síť (DNN). Druhá volba spočívá ve výběru generátoru dat. Pro obě metody je možné zvolit rychlost otáčet laseru (RPM) nebo teplotu (Temperature). Celkově jsou tedy k dispozici 4 možné scénáře. Vstupní hodnoty jsou omezeny na 1 a 2. Jakákoli jiná zadaná hodnota způsobí chybovou hlášku. Toto ošetření je použito pro všechna vyskakovací okna. Okno je vidět na obrázku 3.2. Funkce používá zadaná data uživatelem pro nastavení vnitřních proměnných, pomocí nichž poté spouští další funkce.



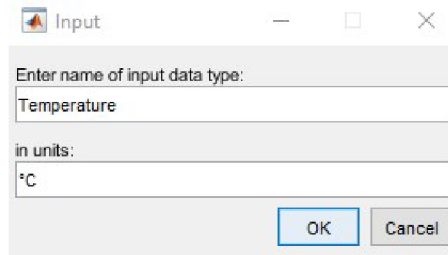
Obr. 3.2: Okno uživatelského vstupu výběru metody

3.2.1 Volba regrese

Při zvolení metody regrese dojde k zobrazení dalšího okna kde je uživatel požádán o vyplnění konkrétních informací k vstupním datům. Původním cílem byla možnost přímého nahrání dat a jejich specifikace ve vyskakovacím okně. Vzhledem k nutnosti generovat data je již okno předvyplněné podle volby generátoru dat. Okno je vidět na obrázku 3.3. Po proběhnutí metoda v příkazovém okně zobrazí výsledky a vytiskne graf s příkladem kritického intervalu.

3.2.2 Volba detailního nastavení metody regresní analýzy

V programu se nachází také možnost detailního nastavení této metody. První parametr je citlivost (sensitivity). Ta ovlivňuje citlivost regresní analýzy vzhledem k velikosti prohledávaného intervalu a četnosti výskytu podezřelé hodnoty. Vyšší citlivost znamená, že jsou označeny i méně časté kritické výkyvy hodnot ve větším okolí



Obr. 3.3: Okno uživatelského vstupu specifikace vstupních dat pro regresní analýzu

bodů, kde došlo ke zvýšení počtu neshodných kusů (detailně vysvětleno v kapitole 3.3).

Další parametry jsou pak velikost jednotlivých generovaných intervalů (`gen_interval_size`) a množství vygenerovaných intervalů (`amount_of_int`) při zavolání funkce generátoru.

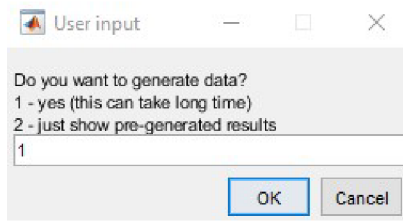
Posledním možným nastavením je pak počet vzorků v okolí podezřelého bodu, pro funkci dělení dat na intervaly (kapitola 3.3.5).

```
%% Options for regression
if (method == 3)
    sensitivity = 2; % 1-high 2-medium 3-low
    gen_interval_size = 200; %velikost genero
    amount_of_int = 5; %pocet generovanych in
    nos = 300; %pocet vzorku v okoli hotspotu
    min_perc_diff_opt = 30;%minimalni odlisno
end
```

Obr. 3.4: Speciální nastavení metody regrese

3.2.3 Volba neuronové sítě

Při zvolení metody neuronové sítě dojde k zobrazení vyskakovacího okna a program čeká na reakci uživatele. Je potřeba zvolit jestli je žádoucí vygenerování nového datasetu nebo pouze zobrazení připraveného. Tato možnost se v programu nachází z důvodu velké výpočetní náročnosti generování dat pro neuronovou síť. Generování velkého datasetu může trvat i desítky minut a proto je žádoucí možnost načtení již dříve vygenerovaných dat. Okno je vidět na obrázku 3.5.



Obr. 3.5: Okno uživatelského vstupu generátoru dat pro neuronovou síť

Další zobrazené vyskakovací okno nabízí volbu mezi učením neuronové sítě a načtením a zobrazením výsledků neuronové sítě navržené autorem práce. Při volbě učení se může doba výpočtu protáhnout až na několik hodin.

Nakonec po zadání všech možností program zobrazí výsledky. Při zvolení zobrazení výsledků na připravených datech a připravené neuronové síti může zobrazení výsledku zabrat několik minut. Program zobrazí přesnost neuronové sítě přes všechny sekvence testovacích dat a vypočítá tři další parametry (recall, precision a F1 score) pro jednu vybranou sekvenci dat (z důvodu zkrácení výpočetní doby).

3.2.4 Volba detailního nastavení metody neuronové sítě

Program také nabízí možnost specifického nastavení metody neuronové sítě. Jednotlivá nastavení jsou vidět na obrázku 3.6.

```

num_of_int = 20;
nos = 6;
nof = 3;
%volby pro teplotu
nok_interval_size_temp_opt = [200 3000];
ok_interval_size_temp_opt = [2500 7500];
num_of_noks_opt = [2,5];%pocet NOK inter
%volby pro RPM
nok_interval_size_rpm_opt = [40 90];%roz
ok_interval_size_rpm_opt = [2000 4000];%

```

Obr. 3.6: Speciální nastavení metody regrese

První nastavení je počet intervalů (`num_of_int`). Tento počet znamená, kolik intervalů bude vloženo do jednoho prvku jedné sekvence. Ovlivňuje délku jednotlivých prvků tak, že nastavuje kolikrát proběhne cyklus vložení kombinace intervalů (detailně vysvětleno v kapitole 1.5.6).

Další nastavení jsou počet trénovacích/testovacích sekvencí (`nos`) a počet prvků v jednotlivých sekvencích (`nof`), které prakticky znamenají počet řad dat v jednotlivých sekvencích.

Další možnost se týká jednotlivých vkládaných intervalů. V programu se nacházejí generátory OK dat a NOK dat jak pro teplotu tak pro rychlost otáček

laseru. To znamená, že jsou k dispozici celkem čtyři nastavení pro tyto generátory. Jedná se o specifikaci maximální a minimální velikosti generovaných intervalů. Z těchto navolených intervalů pak při zavolání funkce vybere náhodně číslo a vygeneruje daný interval v dané délce. Jedná se o nastavení `nok_interval_size_temp_opt`, `ok_interval_size_temp_opt`, `nok_interval_size_rpm_opt` a `ok_interval_size_rpm_opt`.

Poslední možnou volbou je počet NOK intervalů vložených na jeden OK interval (`num_of_noks_opt`). Tato možnost se týká testovacích dat, kde jsou NOK intervaly náhodně vkládány do OK intervalů.

3.3 Metoda regresní analýzy

Programové řešení této metody se skládá z několika postupně volaných funkcí. V následujících kapitolách budou tyto funkce vysvětleny. Výstupem této metody je nalezení opakujících se výkyvů v datasetu obsahujícím záznam nepřetržitého toku dat z výrobního stroje. Program je navržen tak, aby akceptoval libovolný typ datasetu, jakkoli dlouhý. Jediným požadavkem je korespondující dataset obsahující množství vyrobených vadných kusů. Právě na základě počtů vadných kusů program extrahuje podezřelé intervaly. Jedná se o časové okolí doby vyrobení vadného kusu. Program nachází ty, co jsou si podobné a tedy opakovaně způsobují chybu. Nad určitou mírou výskytu pak tento interval označí jako kritický.

3.3.1 Načítání dat (skript `data_loading.m`)

Hlavní funkcí tohoto skriptu je načítání dat z tabulek ve formátu MS Excel (.xlsx). Data jsou uložena pod sebou ve sloupci, tak že právě jeden vzorek se nachází v jedné buňce. Skript používá funkci Matlabu `readmatrix`, která je dostupná od verze 2019a. Jako další se v této funkci nachází výše zmíněné vyskakovací okno specifikující vstupní data. Do okna se zadává název typu dat (např. teplota) a jednotky (stupně celsia). Načtená data se pak stávají základem generátorů dat.

3.3.2 Předzpracování dat (skript `data_preproces.m`)

Tato funkce má za úkol vstupní data ořezat tak, aby se zachovala pouze užitečná data. Jako první jsou oříznuta data tak, aby místa kde senzor nesnímal (nedodával informaci), byla odstraněna (polovina dat jsou hodnoty teploty a druhá polovina jsou pouze nuly a tak mohou být odstraněny). Jako další jsou vypočítány limity okolo střední hodnoty. Tyto limity jsou nastaveny na 25% (experimentální hodnota) nad a pod střední hodnotu všech dat. Pomocí těchto limit jsou vymazány hodnoty, které se od limit střední hodnoty liší o více než 200% (tzv. outliers neboli chyby snímače). Opět se jedná o experimentálně nastavenou hodnotu, která by se při testování mohla měnit.

Jako poslední je pro dané diskrétní vzorky vytvořena posloupnost dat. Tato posloupnost obsahuje čísla od jedna až po velikost vstupní matice a je použita pro zobrazování dat v grafech.

3.3.3 Generování dat rychlosti otáček laseru pro metodu regrese (skript data_gen_rpm.m)

Tento skript má za úkol vygenerovat posloupnost počtů vadných kusů korespondující s daty o rychlosti otáčení laseru. Tato data jsou následně použita pro rozdělení velkého vstupního datasetu na menší intervaly pro jednodušší a rychlejší zpracování velkého množství vzorků.

Tento skript je spuštěn pouze když uživatel zvolí možnost generování dat otáček laseru.

3.3.4 Generování dat teploty pro metodu regrese (skript data_gen_temp.m)

V tomto skriptu jsou generována data počtu vadných kusů odpovídající datům teploty. Stejně jako u rychlosti otáčení laseru, je hlavním účelem těchto dat následné rozdělení vstupního datasetu na intervaly pro snazší zpracování.

Tento skript je spuštěn pouze když uživatel zvolí možnost generování dat teploty.

3.3.5 Rozdělení vstupních dat na intervaly (skript data_parsing.m)

V tomto skriptu jsou vstupní data rozdělena na jednotlivé intervaly. Jako první je prohledána posloupnost dat obsahující počet vyrobených vadných kusů. Jsou uloženy indexy, na kterých došlo k inkrementaci množství vadných kusů.

Po nalezení tzv. „hotspotů“ (podezřelá hodnota), jsou určeny intervaly zkoumání (podle zvolené šířky intervalů). Tyto intervaly jsou uloženy do matice o počtu řádků podle zadané velikosti intervalů a počtu sloupců podle počtu nalezených „hotspotů“. Tím je dosaženo uložení všech podezřelých intervalů vedle sebe do jedné matice. K tomu je vytvořena pomocná matice obsahující posloupnosti od jedné po počet vzorků v jednotlivých intervalech, pro pozdější tisknutí do grafů.

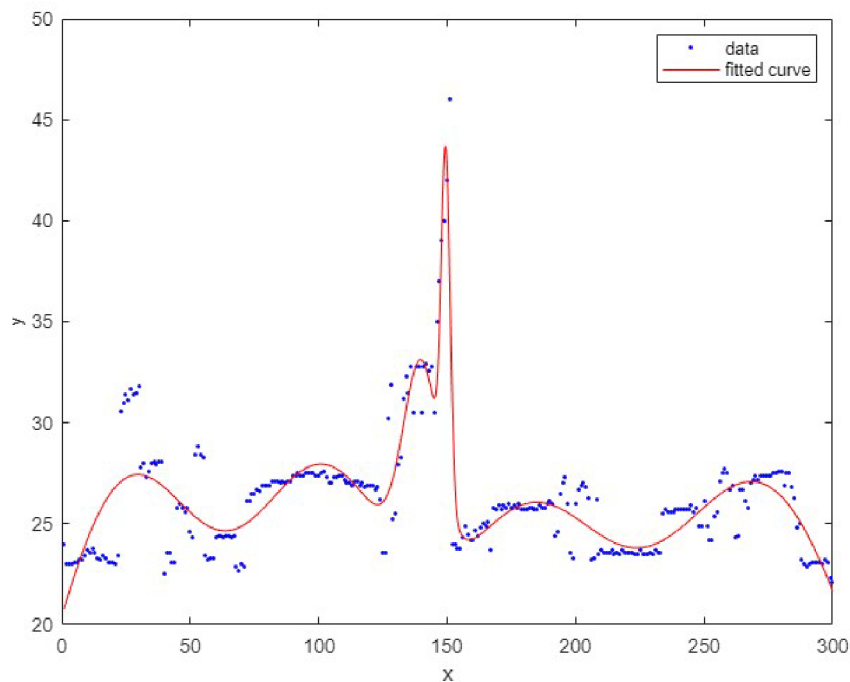
3.3.6 Metoda regresní analýzy (skript regression3.m)

Úkolem tohoto skriptu je nalezení kritických hodnot v předem rozdělených intervalech. Ukládány jsou pouze hodnoty dostatečně se opakující. Jsou hledány kritické hodnoty pěti různých ukazatelů. První z nich je maximální hodnota. Ta může mít vysokou výpovědní hodnotu například pokud se jedná o data teploty. Dalším je pak minimální hodnota, která může být důležitá v případě například tlaku vzduchu nebo vody, kde by pokles pod určitou hodnotu mohl způsobit chybu. Další dva ukazatele souvisí s gradientem hodnot. První z nich je maximální pozitivní gradient neboli maximální nárůst za nejkratší čas. Nárůst prakticky jakékoli hodnoty příliš

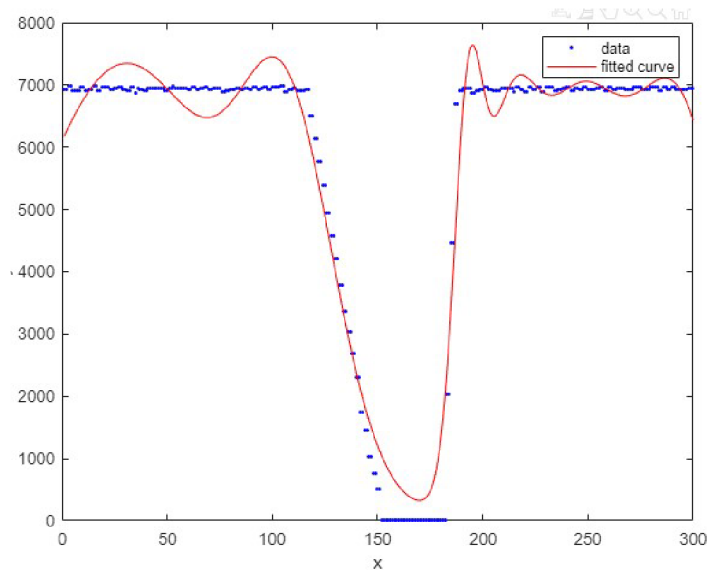
rychle může způsobit chybu ve výrobě. Druhým z nich je pak maximální záporný gradient. Opět tato hodnota má velkou výpovědní hodnotu u většiny dat. Například při obrábění (honování, broušení) může prudký pokles otáček znamenat problém. Posledním ukazatelem je pak průměrná odchylka intervalu. Tato hodnota nachází intervaly ležící mimo meze okolí střední hodnoty dat a zjišťuje, jestli se některé velikosti opakují. Odchylka je počítána ze všech hodnot intervalu vzhledem k průměrné hodnotě. Pokud se nějaká velikost intervalu opakuje, je spočítána její průměrná velikost odlišnosti od průměrné hodnoty, a je uložena.

Pokud dojde k nalezení více ukazatelů, které se dostatečně opakují, jsou uloženy všechny. Nastavení citlivosti může způsobit nalezení více nebo naopak méně kritických hodnot.

Výpočet všech kritických hodnot jednotlivých ukazatelů je založen na proložení intervalu gaussovou funkcí šestého řádu. Z této křivky jsou pak počítána maxima, minima a pomocí derivací i gradient. Zároveň je počítána kvalita aproximace dat, která by v budoucnu mohla být použita pro adaptivní aproximaci dat. Zatím není implementována, ale bylo by možné měnit řád gaussovy funkce podle kvality aproximace. V tomto skriptu je opakovaně volána funkce `regr.m` vracející matici obsahující všechny kritické hodnoty daného ukazatele. Funkce je vysvětlena v kapitole 3.3.7. Dva příklady proložení jsou vidět na obrázcích 3.7 a 3.8.



Obr. 3.7: Příklad proložení hodnot teploty gaussovou křivkou

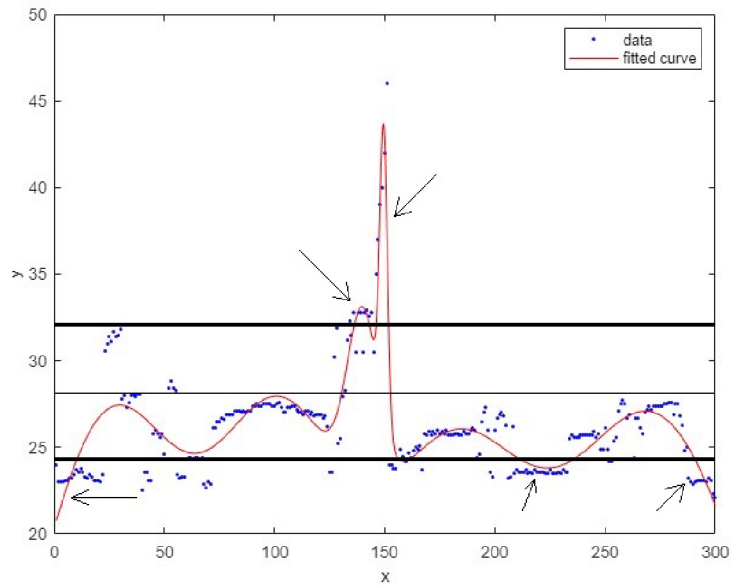


Obr. 3.8: Příklad proložení hodnot otáček gaussovou křivkou

Nacházení průměrné odchytky intervalů od střední hodnoty

Pro nalezení průměrné odchytky intervalu je použita jiná metoda. Postupně jsou procházeny všechny předem uložené intervaly. Pro každý interval jsou postupně nacházeny hodnoty ležící mimo meze okolí střední hodnoty. Po nalezení všech hodnot v daném intervalu ležících mimo tyto meze a jejich uložení do matice (ukládány jsou pouze indexy), dochází k jejich rozdělení. Prakticky se jedná o nacházení jednotlivých součástí intervalů podle toho, jestli indexy hodnot ležících mimo meze střední hodnoty, jdou ihned po sobě a nebo je mezi nimi skok. Pokud na sebe indexy přímo nenavazují, jsou všechny hodnoty dosud uložené (na sebe navazující) prohlášené jedním intervalem, který je pak dále analyzován. Jakmile jsou všechny odlišující se intervaly v daném analyzovaném bloku dat rozděleny, dochází k počítání odlišnosti každé hodnoty od průměrné hodnoty. Po spočítání všech odchylek je spočítána jejich průměrná hodnota.

Jsou uloženy pouze intervaly, jejichž průměrná odchytka od střední hodnoty, je větší než 30% (lze měnit v nastavení metody). Příklad těchto intervalů je vidět na obrázku 3.9. Černé šipky ukazují na hledané intervaly.



Obr. 3.9: Příklad intervalu ležících mimo okolí střední hodnoty

3.3.7 Metoda regresní analýzy (skript `regr.m`)

Hlavním úkolem této funkce je určit, jak nalezené podezřelé hodnoty rozdělit na validní a náhodné. Funkce postupně prochází vstupní pole všech kritických hodnot a hledá ty, co jsou si navzájem podobné. Na začátku cyklu je uložena první hodnota ze vstupního pole a následně je procházen jeho zbytek. Pokud je nalezena hodnota ležící v intervalu specifikovaném parametrem citlivosti, je uložena do pole navzájem si podobných hodnot. Po nalezení každé další hodnoty je přepočítán průměr ze všech podobných si hodnot a pomocí parametru citlivosti jsou nastaveny nové meze pro hledání dalších podobných hodnot. Zároveň jsou také uloženy indexy všech uložených podobných hodnot a po každém prohledání celého vstupního pole jsou buňky na těchto indexech vymazány, protože již jednou byli zpracováni a mohli by zavádět chybu v dalších iteracích. Když je celé vstupní pole prohledáno, je-li zjištěna procentuální hodnota výskytu vyšší než je nastavený práh (podle parametru citlivosti), je uložen průměr všech těchto podobných hodnot do výstupního pole. Takto je vstupní pole procházeno dokud není prázdné. Po vyprázdnění vstupu funkce vrátí matici obsahující všechny kritické hodnoty daného ukazatele vyskytující se opakovaně.

Na obrázku 3.10 je vidět příklad zavolání funkce `regr`. Jako první parametr je funkci předáno vstupní pole kritických hodnot, poté název daného ukazatele v uvozovkách (např. `maximum`) a nakonec parametr citlivosti. Funkce vrátí pole o dvou sloupcích a jednom a více řádcích podle počtu nalezených dostatečně se opakujících kritických hodnot.

```

>> regr([98;97;1510;96;1550;96;1515;2;3;99;1501],"ukazatel",sensitivity)

ans =

2x2 cell array

{"ukazatel"}    {[97.2000]}
{"ukazatel"}    {[ 1519]}

```

Obr. 3.10: Příklad zavolání funkce regr

Postupně je volána tato funkce na matice obsahující různé druhy ukazatelů a následně jsou výstupy ukládány do poslední finální matice. Tato matice obsahuje všechny kritické hodnoty všech ukazatelů.

Funkce regr je univerzální a v budoucnu může být do metody regresní analýzy přidáno více ukazatelů.

3.3.8 Tisk výsledku metody (skript data_print.m)

Úkolem této funkce je přehledně zobrazit výsledky metody regresní analýzy. Na obrázku 3.11 je vidět zobrazení všech nalezených ukazatelů. U maxima a minima jsou jednotky ve stupních Celsia. V případě gradientů se jedná o pokles ve stupních Celsia za vzorek (v případě vzorkovací periody 1s je to pokles za jednu vteřinu). Odlišnost intervalu je na konci uvedena v procentech.

```

>> data_print
Maximals = 43.3703°C
Minimals = 20.7673°C
Positive Gradient = +4.2952°C/sample
Negative Gradient = -5.7106°C/sample
Average interval difference percentage = 46.6033%

```

Obr. 3.11: Ukázka tisku dat

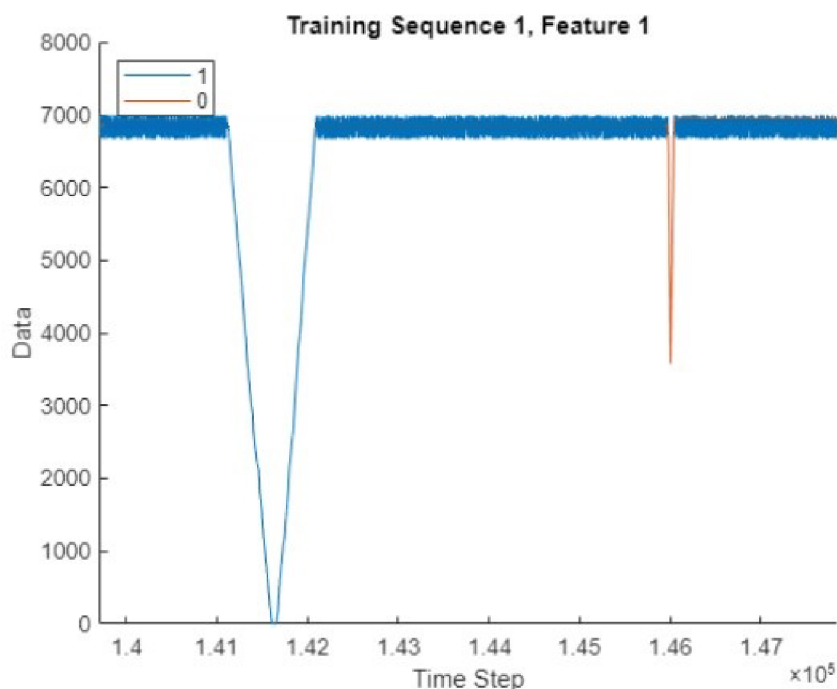
Skript pro tisk hodnot je opět napsán univerzálně, takže po přidání dalších ukazatelů je skript schopný vytisknout všechny.

3.4 Neuronová síť

Pro tuto práci byla vybrána síť typu LSTM (1.5.6). Hlavním cílem práce je nalezení kritických intervalů v datech. Díky tomu se tato síť perfektně hodí, jelikož dokáže zpracovávat sekvence dat a dokáže odhalit opakovaně se vyskytující intervaly v těchto sekvencích. Hlavní předností je pak zapamatování si důležitých intervalů a na jejich základě klasifikace vstupních dat. Vlastně se jedná o zapamatování si krátkých závislostí v širokém časovém měřítku.

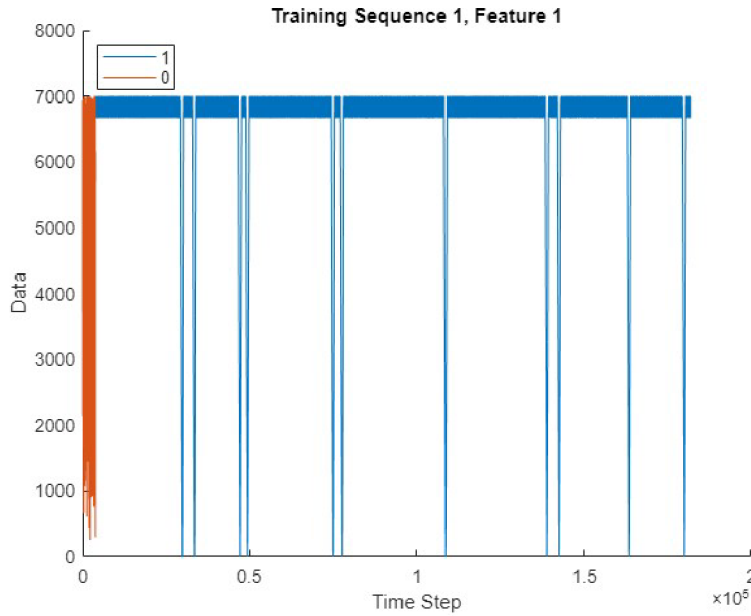
3.4.1 Skript pro tvorbu datasetů (DNN_data.m)

V této práci se nachází generátor datových řad simulující reálná data ze strojů ve Škoda Auto a.s. Generátor lze z části považovat pouze za program na tvorbu testovacích a trénovacích dat. Program vytváří oba datasety opakovaným voláním dvou funkcí na generování dat. V tomto programu by bylo možné použít reálná data místo volání generátorů. Tento skript vytváří data x , kterým odpovídají data y . V datech x se nachází skutečné hodnoty ze snímačů (např. teplota nebo otáčky). V datech y se pak nachází kategorické proměnné specifikující zda odpovídající data v x jsou OK nebo NOK. Příklad takto vytvořených dat je vidět na obrázku 3.12, kde oranžovou barvou (0) jsou označena NOK data a modrou barvou (1) jsou označena OK data. Generátory OK a NOK intervalů jsou probírány v kapitolách 3.4.2 a 3.4.4.

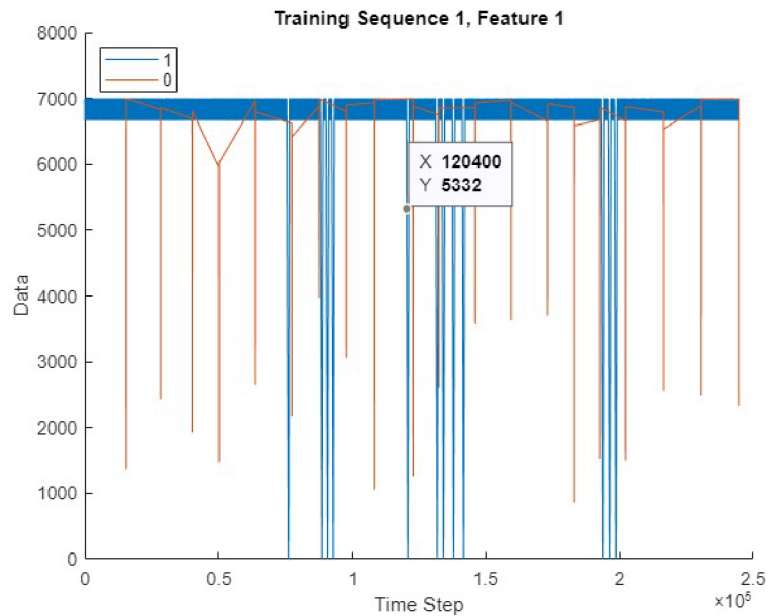


Obr. 3.12: LSTM krok 4[35]

Tento skript vytváří dva různé datasety pro neuronovou síť. Jeden z nich je trénovací a druhý je testovací. Trénovací dataset je poskládán tak, že nejdříve jsou vkládány vygenerované NOK intervaly a po nich jsou vkládány OK intervaly. Tato data jsou vidět na obrázku a jsou použita k trénování neuronové sítě.



Obr. 3.13: Ukázka složení trénovacích dat



Obr. 3.14: Ukázka složení testovacích dat

Ve skriptu se nachází několik cyklů, ve kterých dochází k volání generátorů intervalů a jejich následnému skládání do finálních datasetů.

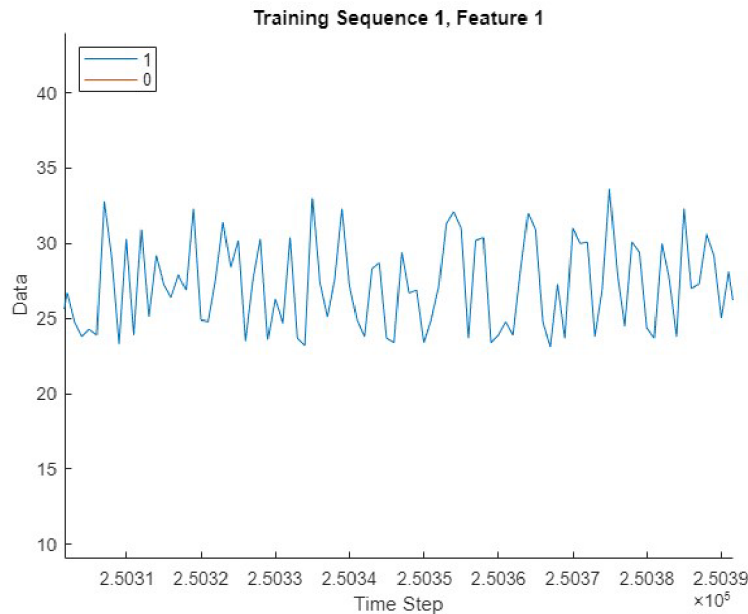
Ve vnitřním cyklu dochází nejprve k vymazání dočasných proměnných a následném zavolání dvou ze čtyř generátorů, podle vybraného typu dat a uložení výstupu generátoru do dočasných proměnných. Podle uživatelem nastavené proměnné `nof` (počet prvků v sekvenci), poté probíhá celý proces. Vnitřní cyklus probíhá právě tolikrát, kolik je prvků v sekvenci. Vždy při generování prvního intervalu dojde k vygenerování pomocných proměnných, které určují složení generovaných intervalů z OK a NOK intervalů. V jednotlivých prvcích sekvence musí být stejné složení OK a NOK prvků (číselně se mohou lišit, ale pozice musí být stejná), protože pro všechny prvky v datové proměnné existuje pouze jedna kategoričká proměnná. Prakticky dojde k proběhnutí vnitřního cyklu podle požadovaného počtu prvků a při skládání prvního intervalu program náhodně rozhodne o velikosti generovaných OK a NOK intervalů a o jejich složení. U testovacích dat existují tři možnosti, je vložen dvakrát NOK interval, dvakrát OK interval nebo jeden OK a jeden NOK interval. U trénovacích dat jsou vytvářeny dvě velké proměnné OK a NOK intervalů, které jsou nakonec spojeny do jedné. K tomu je vytvořena i korespondující kategoričká proměnná pro oba typy dat. Po každém složení intervalů je ověřena jejich velikost a případně větší z intervalů oříznut pro dodržení velikosti jednotlivých prvků (řádků v matici). Opakovaným průběhem těchto cyklů je postupně skládána celá sekvence. Po dokončení celé sekvence dochází k jejímu uložení do buňky výstupní proměnné (`YTrain`, `XTrain`, `YTest`, `XTest`). V těchto proměnných se nachází data pro trénování a testování neuronové sítě. Výstupní proměnné pak obsahují určitý počet sekvencí dat (podle uživatelského nastavení). Tyto sekvence jsou pak složeny z jednoho až tří prvků. Každý prvek je vlastně posloupnost dat simulující nepřetržitý výstup ze stroje. Každý prvek je pak složen z intervalů, které simulují buď bezproblémový průběh výroby (OK interval) nebo chybný (NOK interval). Detailní popis dat a jejich generátorů se nachází v kapitolách 3.4.2 a 3.4.4.

3.4.2 Generátor OK intervalů

OK data teploty (skript `OK_generator_temp.m`)

Generátor dat teploty je založen na reálných datech ze snímače z výroby. Bohužel se jedná pouze o OK data, protože jsou z doby kdy neprobíhala výroba ale pouze ruční sběr dat. Tato data jsou načtena ze souboru `shuf.xlsx`. Pomocí zavolání funkce na zamíchání dat (vysvětleno v kapitole 3.4.3) jsou reálná data zamíchána. K tomu dochází z důvodu potřeby použít data opakovaně. Potom co funkce na míchání dat vrátí potřebné intervaly podle nastavených hodnot (počet a velikost intervalů), dochází k jejich skládání a ukládání do výstupní proměnné. Hlavní program si pak

náhodně vybírá jeden z takto složených intervalů. Původně tato funkce generovala rovnou i NOK intervaly, ale z důvodu potřeby generování testovacích a trénovacích dat o jiném složení a zároveň variabilitě intervalů, byl NOK generátor přesunut do samostatného skriptu. Ukázka OK dat se nachází na obrázku 3.15.



Obr. 3.15: Příklad vygenerovaných OK dat teploty

OK data rychlosti otáček (skript `OK_generator_rpm.m`)

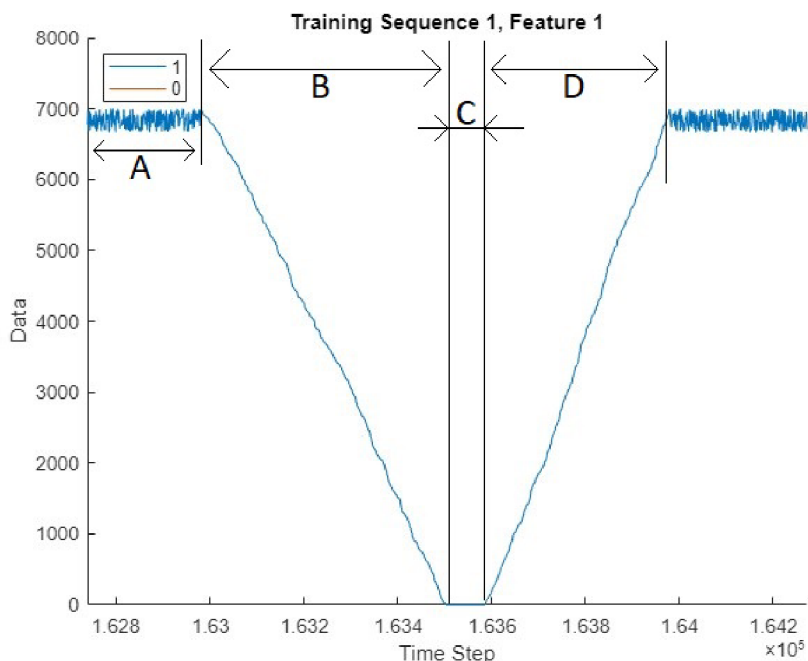
Generátor rychlosti otáček vrtáku není založen na reálných datech. Jedná se o náhodně vygenerované intervaly složené ze čtyř částí. Tyto části jsou vidět na obrázku 3.16. Na začátku je vygenerována velikost intervalu B+C+D a následně jsou generovány velikosti jednotlivých intervalů. Klesající i rostoucí interval je vygenerován pomocí vytvoření pole o dané velikosti a potřebných náhodných hodnotách, jež jsou následně seřazeny vzestupně nebo sestupně.

V části A se nachází hodnoty mezi 6666 a 7000 rpm, které simulují aktivní část obrábění a rychlost otáčení při vrtání. Z těchto hodnot se skládá převážná část dat.

V části B je pak vidět simulování situace kdy je vrták neaktivní a postupně se snižují otáčky vypnutím motoru. Jedná se o zpomalování otáček setrvačností. Pokud by se otáčky zpomalily příliš rychle, mohlo by se jednat o chybu (podrobněji popsáno v kapitole 3.4.4). Data jsou generována náhodně, ale při snižování rychlosti je vygenerován vždy stejný počet vzorků, čímž je zaručena stejná strmota klesání.

Třetí část je pak doba vypnutí vrtáku a nulových otáček. Jedná se o interval C. Délka této doby je náhodně generovaná.

Poslední část je pak interval D. Jedná se o znovu naběhnutí otáček vrtáku na maximální hodnotu. Strmost stoupání této části se může mírně měnit podle velikosti celého intervalu vypnutí a zapnutí pohonu. Interval D nemá generovanou velikost ale je doplňkem na celkovou velikost intervalu poklesu. Hodnoty pak pokračují opět intervalem A.



Obr. 3.16: Příklad vygenerovaných NOK dat teploty

3.4.3 Zamíchání dat (skript `data_shuffler.m`)

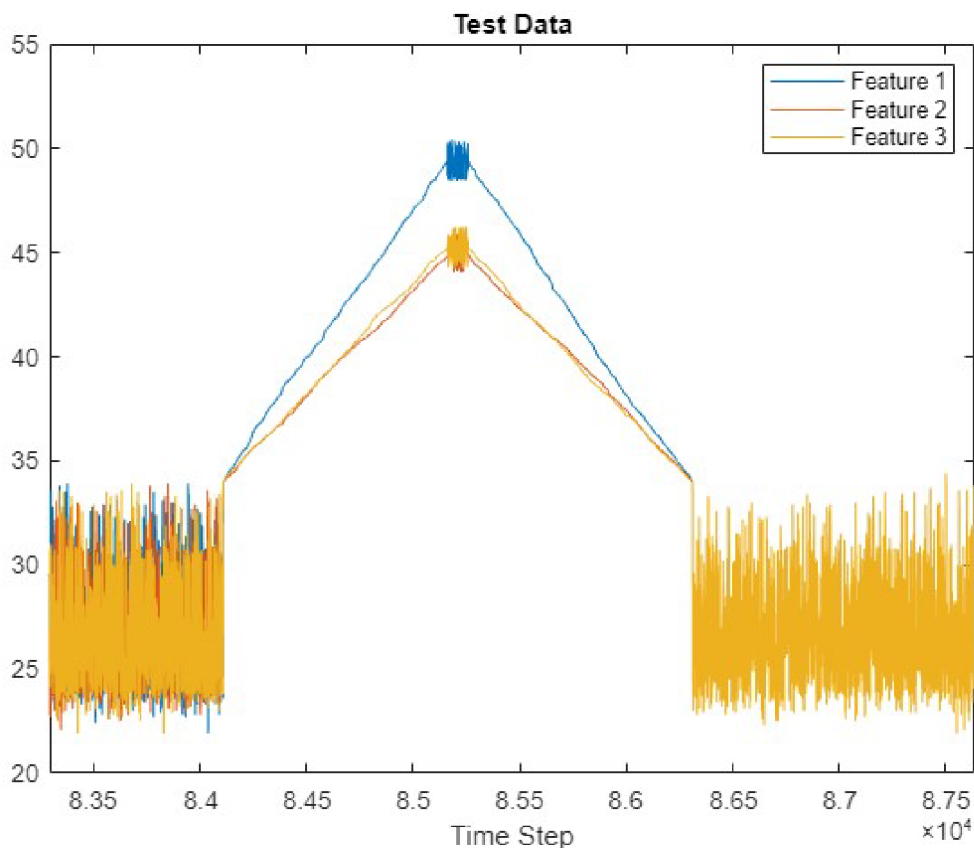
Úkolem této funkce je zamíchání vstupních dat a z nich následné vytvoření pole obsahujícího tolik OK intervalů, kolik bylo zadáno. Vstupem je přibližně 70000 vzorků z reálného snímače. Pro opakované použití dat je potřeba je vždy zamíchat a vytvořit unikátní interval. Ze vstupních dat je náhodně vybrán interval dat a pomocí náhodných permutací jsou hodnoty zamíchány. Po vytvoření všech požadovaných intervalů jsou tato data poslána na výstup.

3.4.4 Generátor NOK intervalů

NOK data teploty (skript `OK_generator_temp.m`)

Tato funkce má za úkol vygenerování NOK intervalu dat teploty podle zadaného parametru velikosti intervalu. Na obrázku 3.17 jsou vidět tři náhodně vygenerované intervaly (intervaly z nichž je složen jeden prvek jedné sekvence). Každý interval je

složený ze tří částí. Jejich velikost určuje hlavně náhodně vygenerovaná část B. V této části se nachází simulace fluktuace teploty kolem maximální hodnoty. Velikosti částí A a C jsou pak nastaveny jako poloviny zbývající velikosti celého generovaného intervalu. Pokud nárůst teploty považujeme za poruchu klimatizace nebo těsnění, teplota by se měla logicky vracet na původní hodnotu stejně rychle. Při stoupání pak může být strmost náhodná. Na obrázku 3.17 interval A značí poruchu a nárůst teploty a interval C její opětovné vrácení k normálu. Intervaly jsou opět generovány jako náhodné matice hodnot a následně jsou seřazeny vzestupně nebo sestupně. Na konci funkce jsou všechny tři intervaly spojeny do jednoho a poslány na výstup.



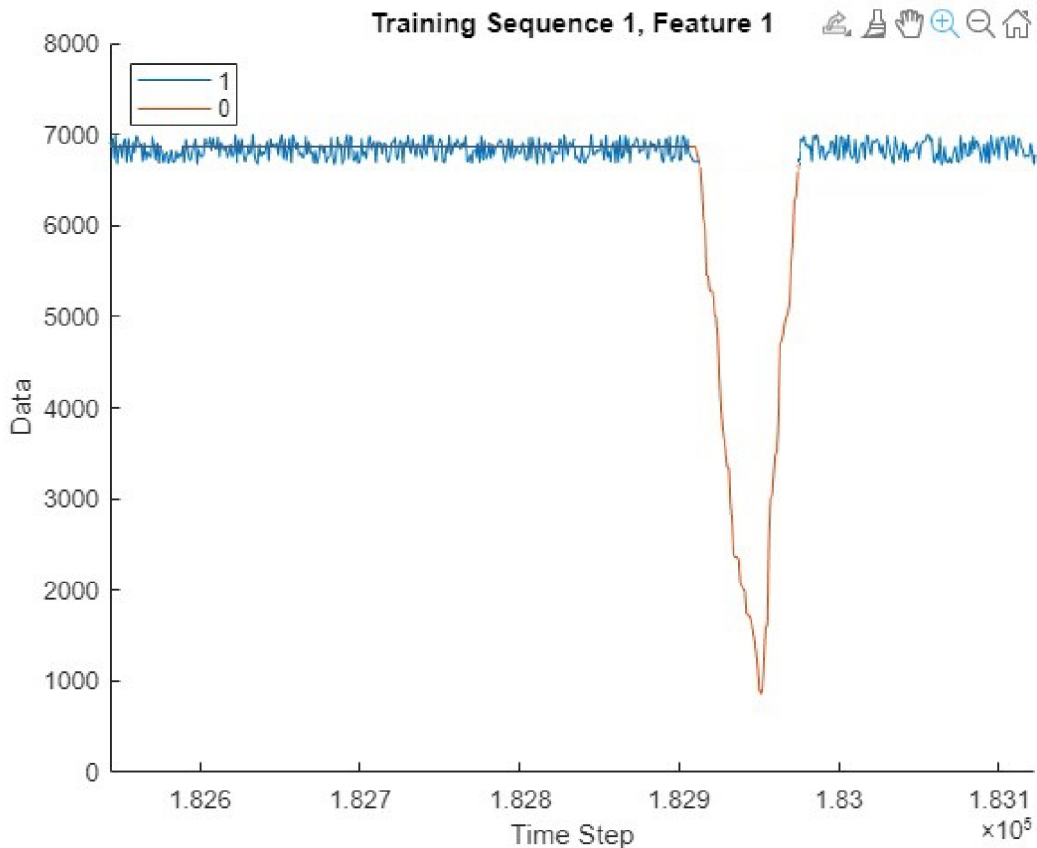
Obr. 3.17: Příklad vygenerovaných NOK dat otáček

NOK data rychlosti otáček (skript `OK_generator_rpm.m`)

Výstupem tohoto skriptu jsou NOK intervaly rychlosti otáček. Příklad tohoto intervalu je na obrázku 3.18. K výskytu takového intervalu může dojít například při vrtání příliš hluboko, kde se vrták zasekne, při špatně připraveném materiálu (například příliš vysoká nebo nízká teplota) nebo při poklesu napětí na napájení pohonu vrtáku a tudíž při nedostatečném výkonu. NOK intervaly jsou generovány postupně,

složením ze dvou intervalů. Nejdříve je vygenerováno minimum mezi 0 a 4000, což značí jak moc poklesnou otáčky neboli jak moc se vrták zbrzdí v materiálu.

Poté jsou vygenerovány dva náhodné intervaly poklesu a nárůstu otáček. Tento interval je úplně náhodný, protože i v reálné situaci může dojít k náhodnému poklesu otáček. Zároveň pak jejich nárůst může být ovlivněn stavem vrtáku (například při zaseknutí), tudíž je i nárůst náhodný. Nakonec je výsledný vygenerovaný interval poslán na výstup funkce.



Obr. 3.18: Příklad vygenerovaných NOK dat otáček

3.4.5 Parametry neuronové sítě (skript NN.m)

V této kapitole se nachází popis nastavených parametrů neuronové sítě navržené autorem práce. Tyto parametry byly nastaveny experimentálně převážně z předchozích zkušeností. Při prvních pokusech naučit neuronovou síť bylo zjištěno, že data musí být správně označena kategoričkou proměnnou. Na začátku byla data ve formátu tak, že byl vytvořen OK interval a do něj vložen NOK interval, a jako NOK byl označen celý interval což zavádělo velkou chybu. Později byla jako NOK označena

už jen skutečná odchylka od normálu jako je znázorněno na obrázku 3.18, vyznačeno oranžovou barvou.

Na obrázku 3.19 jsou pak vidět jednotlivé parametry učení neuronové sítě. První parametr `nof` specifikuje počet prvků v každé sekvenci. Druhý parametr `numHiddenUnits` pak určuje počet neuronů ve skryté vrstvě. Poslední parametr `numClasses` pak specifikuje kolik tříd má daná síť klasifikovat. Další parametry (`layers` a `options`) specifikují nastavení jednotlivých vrstev a parametrů učení neuronové sítě a jsou probírány v dalších kapitolách.

```
numFeatures = nof;
numHiddenUnits = 150;
numClasses = 2;

layers = [ ...
    sequenceInputLayer(numFeatures)
    lstmLayer(numHiddenUnits,'OutputMode','sequence')
    fullyConnectedLayer(numClasses)
    softmaxLayer
    classificationLayer];
%parametry uceni
options = trainingOptions('adam', ...
    'InitialLearnRate',0.1, ...
    'LearnRateSchedule','piecewise', ...
    'LearnRateDropFactor',0.5, ...
    'LearnRateDropPeriod',25, ...
    'MaxEpochs',200, ...
    'GradientThreshold',1, ...
    'Verbose',0, ...
    'MiniBatchSize',128, ...
    'Plots','training-progress',...
    'ExecutionEnvironment','gpu');

net = trainNetwork(XTrain,YTrain,layers,options);
```

Obr. 3.19: Parametry učení neuronové sítě

Vrstvy NS

- `sequenceInputLayer` - Tato vrstva vytvoří vstupní sekvenci podle velikost vstupního parametru.
- `lstmLayer` - Vytváří LSTM vrstvu (1.5.7), která se učí dlouhodobé závislosti mezi časovými kroky v časové řadě a sekvencemi dat. Vrstva provádí aditivní interakce, což může pomoci zlepšit dlouhodobý tok gradientu v průběhu učení.
- `fullyConnectedLayer` - Specifikuje plně propojenou vrstvu násobící vstup vektorovou maticí a přidává vektor biasů. Specifikuje velikost výstupu.

- `softmaxLayer` - Vytváří softmax vrstvu. Softmax v matematice znamená normalizovanou exponenciální funkci, která bere jako vstup vektor reálných čísel a normalizuje jej do rozdělení pravděpodobnosti sestávajícího z počtu pravděpodobností, podle počtu prvků vstupního vektoru, úměrných exponenciále všech vstupních hodnot.
- `classificationLayer` - Klasifikační vrstva počítá ztrátu křížené entropie (cross entropy loss) kvůli problému s vícetřídovou klasifikací a navzájem se vylučujícími třídami.

Parametry NS

- `'adam'` - Tento parametr specifikuje systém řešení pomocí adam optimalizátoru¹
- `'InitialLearnRate'` - Počáteční rychlost učení použitá pro trénink. Pokud by hodnota byla příliš nízká, učení by trvalo příliš dlouho. Pokud by byla moc vysoká, mohlo by dojít k dosažení suboptimálního výsledku nebo neoptimálního výsledku.
- `'LearnRateSchedule'` - Možnost snížení rychlosti učení během učení. Pomocí volby `'piecewise'` dojde po uběhnutí nastaveného počtu epoch k vynásobení rychlosti učení konstantou.
- `'LearnRateDropFactor'` - Konstanta pro násobení rychlosti učení.
- `'LearnRateDropPeriod'` - Počet epoch mezi snižováním rychlosti učení.
- `'MaxEpochs'` - Maximální počet epoch při učení.
- `'GradientThreshold'` - Hodnota prahu po jejímž překročení dojde k oříznutí gradientu na specifikovanou hodnotu.
- `'Verbose'` - Indikátor povolující zobrazování informací o průběhu učení v příkazovém okně.
- `'MiniBatchSize'` - Velikost mini-šarže použité pro každou iteraci trénování. Prakticky se jedná o maximální počet hodnot, které si proces pamatuje v každém kroku. Ty jsou následně použity pro aktualizaci gradientu ztrátové funkce (loss function) a vah. Tato hodnota je ovlivňována pamětí použité grafické karty. Například při učení sítě pro otáčky vrtáku byla tato hodnota nastavena na 2, protože jakákoli vyšší hodnota způsobila chybu učení.
- `'Plots'` - Grafy, které se mají zobrazit během učení. Možnost `'training-progress'` nastavuje zobrazení průběhu učení přesnosti sítě a ztrátové funkce.

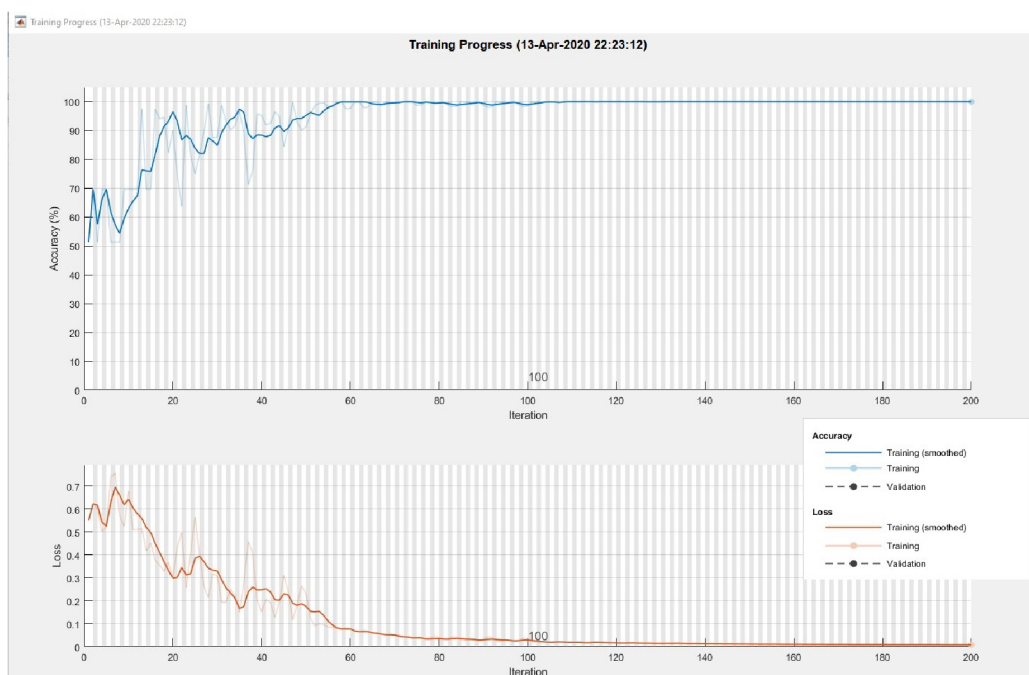
Posledním příkazem je pak příkaz `trainNetwork`, který spustí učení sítě s nastavenými parametry a vstupními trénovacími daty.

¹Adam_optimalizátor

3.4.6 Průběh učení

Neuronová síť pro teplotu

Na obrázku 3.20 je vidět detailní průběh učení neuronové sítě pro sledování průběhu teploty. Na dalším obrázku 3.21 jsou pak vidět detailní parametry učení jako například datum trénování, doba trvání trénování, počet iterací, hardware a rychlost učení v poslední iteraci. V horním grafu je vidět průběh přesnosti neuronové sítě a ve spodním pak průběh ztrátové funkce (jak moc se optimalizace odlišuje od globálního minima).



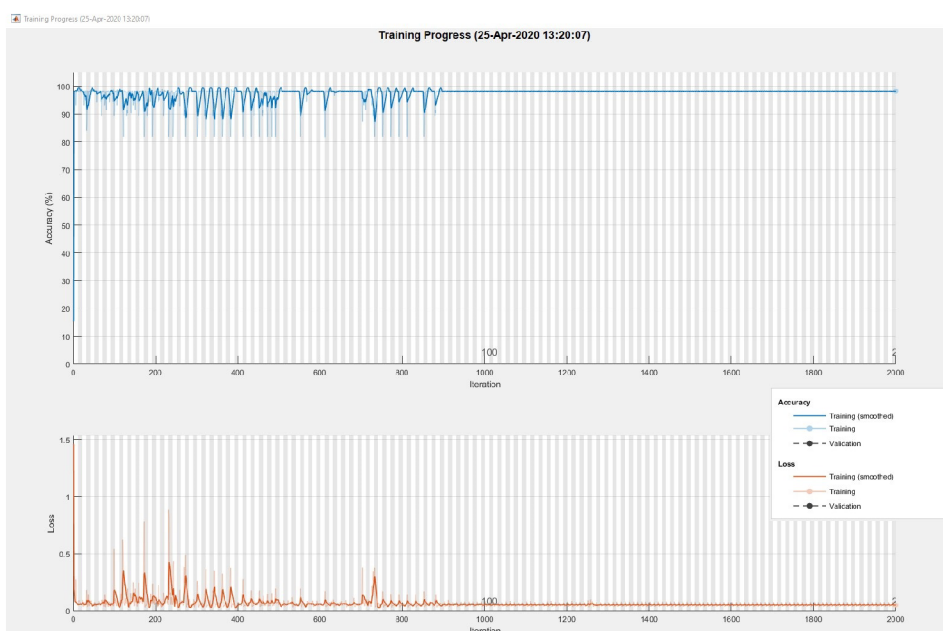
Obr. 3.20: Průběh učení neuronové sítě pro teplotu

Results	
Validation accuracy:	N/A
Training finished:	Reached final iteration
Training Time	
Start time:	13-Apr-2020 22:23:12
Elapsed time:	29 min 17 sec
Training Cycle	
Epoch:	200 of 200
Iteration:	200 of 200
Iterations per epoch:	1
Maximum iterations:	200
Validation	
Frequency:	N/A
Patience:	N/A
Other Information	
Hardware resource:	Single GPU
Learning rate schedule:	Piecewise
Learning rate:	0.00078125

Obr. 3.21: Průběh učení neuronové sítě pro teplotu

Neuronová síť pro rychlost otáček

Na obrázku 3.22 je možné vidět učení neuronové sítě pro sledování průběhu rychlosti otáček. Na dalším obrázku 3.23 jsou uvedeny detailní parametry učení jako například datum trénování, doba trvání trénování, počet iterací, hardware a rychlost učení v poslední iteraci. V horním grafu se nachází průběh přesnosti neuronové sítě a ve spodním pak průběh ztrátové funkce. Průběh se může zdát extrémně strmý a tím se nabízí možnost přeučení sítě. Při učení byla finální přesnost sítě mezi 99% a 100% a následně na testovacích datech se nadále přesnost drží nad 99%, tudíž se o přeučení sítě nejedná. Graf je mírně zkreslen pouze velkým množstvím iterací.



Obr. 3.22: Průběh učení neuronové sítě pro rychlost otáček

Results	
Validation accuracy:	N/A
Training finished:	Reached final iteration
Training Time	
Start time:	25-Apr-2020 13:20:07
Elapsed time:	333 min 19 sec
Training Cycle	
Epoch:	200 of 200
Iteration:	2000 of 2000
Iterations per epoch:	10
Maximum iterations:	2000
Validation	
Frequency:	N/A
Other Information	
Hardware resource:	Single GPU
Learning rate schedule:	Piecewise
Learning rate:	0.00078125

Obr. 3.23: Průběh učení neuronové sítě pro rychlost otáček

3.4.7 Výsledky metody neuronové sítě (skript NN_results.m)

Úkolem tohoto skriptu je vypočítání parametrů neuronové sítě po provedení predikce na testovacím datasetu. V tomto skriptu jsou vypočteny parametry nákladové matice (TP,TN,FP,FN). Parametry TP a TN označují správně klasifikované pozitivní i negativní vzorky. Další dva parametry FP a FN pak označují špatně klasifikované vzorky. Pomocí těchto parametrů jsou pak počítány další parametry vyjadřující kvalitu klasifikace neuronové sítě. Teoretický rozbor počítaných parametrů se nachází v kapitole 1.5.9

Finální vypočtené parametry pro obě sítě jsou vidět v tabulkách 3.1 a 3.2. Na posledním řádku se pak nachází výpis množství všech správně a špatně klasifikovaných vzorků.

Výsledné parametry neuronové sítě pro sledování rychlosti otáček	
Accuracy	99.06%
Precision	98.97%
Recall	100%
F1 score	99.48%
Wrong classified/Total samples	3785/362622

Tab. 3.1: Tabulka NS rychlosti otáček

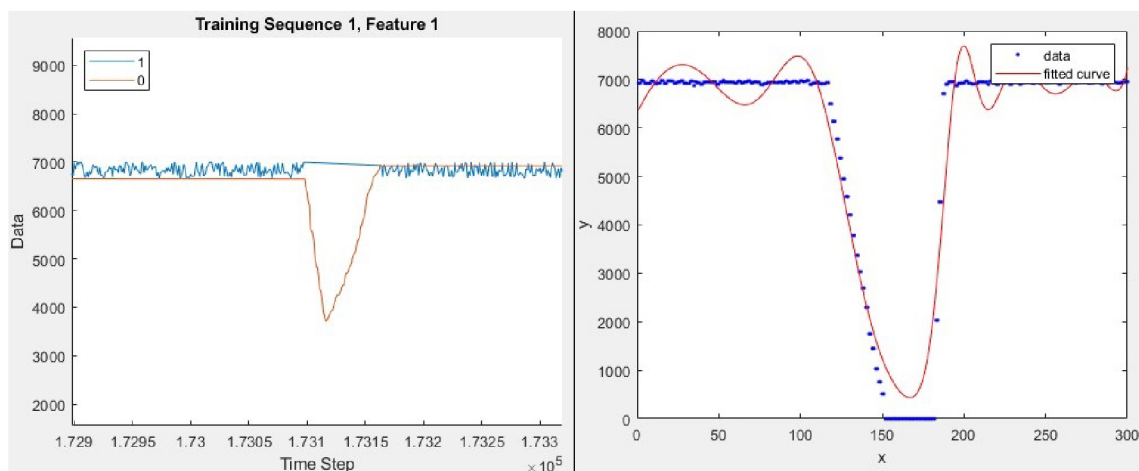
Výsledné parametry neuronové sítě pro sledování teploty	
Accuracy	99.55%
Precision	100%
Recall	99.78%
F1 score	99.89%
Wrong classified/Total samples	645/289995

Tab. 3.2: Tabulka NS teploty

3.5 Porovnání výsledků obou metod

V této práci se nachází dvě různé metody pro potřeby prediktivní analýzy. Tyto metody se od sebe navzájem velmi liší. Metoda regresní analýzy je poměrně výpočetně nenáročná a rychlá (desítky sekund). Naopak metoda neuronové sítě je výpočetně náročnější. Výsledkem regresní analýzy jsou pouze kritické hodnoty určitých veličin, které je potřeba následně v reálném čase sledovat a při jejich překročení spustit alarm. Toto sledování nevyžaduje mnoho výpočetního výkonu, prakticky se jedná o porovnávání dvou hodnot pro každou veličinu. U metody neuronové sítě je pak tento proces mnohem náročnější. Když pomíneme dobu učení a generování dat (při reálných datech tato generování odpadá) tak nastává problém s množstvím sledovaných veličin. Pro každou veličinu je potřeba jedna neuronová síť. Tím s každou další sledovanou veličinou roste výpočetní náročnost. Záleží také na zadavateli, jaké veličiny by chtěl sledovat a jak velký výpočetní výkon je schopen poskytnout.

Při naučení neuronové sítě a provedení regresní analýzy na stejné veličině, bude neuronová síť vždy přesnější. Dokáže zachytit i NOK intervaly, které nejsou tak výrazně chybné. Příklad tohoto jevu je na obrázku 3.24. Vlevo je vidět kritický interval zachycený neuronovou sítí a vpravo je interval nalezený metodou regresní analýzy.



Obr. 3.24: Porovnání výsledků metod

Neuronová síť správně označila interval jako NOK, ale na stejných datech metoda regresní analýzy označila kritický gradient jako -266 RPM/vzorek. Na obrázku vlevo je průměrný gradient klesání -119 RPM/vzorek. Proto by se mohlo stát, že nebude metodou regresní analýzy označen jako kritický, i když ve skutečnosti kritickým je. Při detailním zkoumání bylo zjištěno, že při vypsání jednotlivých gradientů, v několika případech klesají otáčky o více než 266 RPM/vzorek (kritická hodnota označená regresní analýzou). Několik gradientů kritického intervalu je vidět na obrázku 3.25.

-175.0000 -133.5000 -303.0000 -264.0000 -37.0000 -115.0000 -258.0000 -252.0000

Obr. 3.25: Jednotlivé gradienty kritického intervalu

V třetím vzorku pokles otáček dosahuje hodnoty -303 RPM/vzorek a díky tomu by tento interval byl označen jako kritický, protože metoda porovnává každý gradient dvou po sobě jdoucích hodnot. Například, kdyby hodnota byla -260 RPM/vzorek, nepřekročila by práh kritičnosti, proto by došlo k neoznačení tohoto intervalu, neupozornění obsluhy stroje na problém a tím selhání metody. Obě neuronové sítě s přesností přes 99% dokáží většinu těchto intervalů odhalit. V tomto případě se jedná o přesnost predikce každého vzorku. Cílem této práce je odhalit chybné intervaly způsobující chyby ve výrobě. Tyto intervaly jsou složeny z desítek i stovek takových vzorků, proto se dá prohlásit, že tato metoda neodhalí 100% chybných vzorků, ale s vysokou pravděpodobností dokáže odhalit 100% chybných intervalů.

Časová náročnost zpracování hodnot a vytvoření predikce se také liší. U metody regresní analýzy je časová náročnost přibližně lineární a roste s každou tisícovkou dat o sekundu (100000 vzorků se bude zpracovávat 100 sekund). Časová náročnost tedy není tak velká jako při učení neuronové sítě. Při rostoucím množství dat může dojít k problémům s výpočetním výkonem. Doba učení tedy může být několik hodin. Například u neuronové sítě pro hledání anomálií v datech rychlosti otáček trvalo učení necelých sedm hodin. Do této doby není započítaná doba generování dat.

Aplikace obou metod jsou sice ukázány pouze na dvou druzích dat, ale jsou koncipovány jako univerzální. Tedy po odpadnutí generátoru dat a aplikaci reálných výrobních dat je možné obě metody použít téměř na libovolná data z výroby. Program vyžaduje pro změnu vstupu data přímo z datového souboru místo z generátoru. Metody jsou aplikovatelné převážně na data, která mají stálou střední hodnotu s občasnými výchyly. Proto jejich použití není možné například na datech klesajících od maximální hodnoty k nule, jako je stárnutí nástroje a podobně.

Závěr

V první části práce po stanovení cílů projektu bylo přistoupeno k provedení základního průzkumu různých metod prediktivní analýzy. Popis a rozbor těchto metod se nachází v první kapitole této práce. Tato rešerše obsahuje popis pouze nejvýznamnějších metod pro tuto práci.

Druhým krokem byla analýza datových bloků z linky. Původně měli být k dispozici dva zdroje dat. Tato data měla být použita pro základní analýzu, jejímž úkolem mělo být nalezení nejproblémovějších operací respektive strojů na lince plazmy. Zároveň by pak bylo možné analyzovat, které parametry nejvíce ovlivňují výslednou kvalitu kusu. V tomto kroku se nachází jak sběr dat (z několika dostupných zdrojů), tak datová analýza (prohlídka a nalezení užitečných dat).

Při vypracovávání práce došlo bohužel z důvodu situace ve světě k uzavření výrobního závodu Škoda Auto a.s. Mladá Boleslav a tudíž k znemožnění dokončení výrobní linky. Z tohoto důvodu nebyla při vypracovávání k dispozici data z průběhu výroby. K dispozici bylo jen omezené množství dat z testování přenosu ze snímačů z doby kdy neprobíhala výroba. Tato data byla použita pro inspiraci při generování dat. Další data je nutné generovat z důvodu příliš malého množství nasnímaných dat a také z důvodu absence intervalů způsobujících vyrobení vadných kusů. Tyto intervaly je tedy nutno generovat.

V práci se nachází celkem čtyři generátory dat. Pro obě metody jsou k dispozici dva druhy dat a tedy jsou potřeba čtyři generátory. Každý z nich potom generuje různé sekvence dat obsahující OK a NOK intervaly založené na skutečných datech. Velikost a specifická nastavení jsou ovlivněna uživatelským nastavením v hlavní funkci.

Hlavní část práce je popsána ve třetí kapitole a je jí návrh a realizace dvou metod prediktivní analýzy. Byly vybrány dvě metody, které se od sebe liší v mnoha ohledech.

První je metoda regresní analýzy, která je rychlejší, méně komplikovaná, ale zároveň může v některých případech dávat špatné výsledky. Problém s touto metodou je podrobně probrán v samotné práci (3.5). Výstupem této metody jsou kritické hodnoty jednotlivých ukazatelů. Tyto ukazatele je možné použít pro sledování veličin ze strojů a při překročení mezí varovat obsluhu stroje nebo zastavit výrobu. Přesně vyčíslit přesnost této metody je obtížné. Proto její procentuální přesnost byla odhadnuta z opakovaného testování na simulovaných datech na 85-90%.

Druhou metodou je rekurentní neuronová síť typu LSTM. Učení sítě je jak časově, tak i výpočetně náročné. Také u této metody je potřeba pro každou veličinu vlastní neuronová síť, proto tato metoda vyžaduje více výpočetního výkonu i v průběhu nasazení modelu do výroby. Hlavní výhodou této metody je její přesnost, která

je přes 99% pro jednotlivé vzorky a 100% co se týče nalezení celých chybných intervalů na simulovaných datech. Pro reálná data by bylo třeba provést znovu kroky prediktivní analýzy.

Vzhledem k odlišnosti obou metod jsou tedy i výsledky odlišné. Je k dispozici metoda jednodušší s nízkou výpočetní náročností a druhá s vyšší výpočetní náročností, ale s vysokou spolehlivostí. Výběr jedné z těchto metod pro finální nasazení do výroby již záleží na zadavateli zakázky a samozřejmě také hardwarových možnostech firmy. Pokud má firma dostatečné možnosti, je možné nasadit obě metody, nechat metody sbírat data a následně podle výsledků a kvalit predikce upravit původní model. Po dostatečně dlouhé době je možné se na základě statistiky výsledků rozhodnout pro vhodnou metodu a následně ji implementovat do skutečné výroby. Bez aplikace na skutečné hodnoty z výroby a bez důkladného dlouhodobého testování se nedá rozhodnout, která metoda je vhodnější.

Do budoucna se nabízí několik možností zdokonalení obou metod. Pro obě dvě metody je bezpochyby nutné získat reálná data a použít je pro prediktivní analýzu. Pro obě metody s rostoucím množstvím dat roste i množství informací z nich získaných a tím zároveň kvalita predikce. Dalším krokem je také možnost aplikování metody na další dostupné veličiny ze strojů z výrobní linky.

Literatura

- [1] *Predictive Analytics White Paper* [online]. American Institute for Chartered Property Casualty Underwriters/Insurance Institute of America: 2007, Dostupné z URL:
<<https://www.the-digital-insurer.com/wp-content/uploads/2013/12/78-Predictive-Modeling-White-Paper.pdf>>.
- [2] *To Explain or to Predict?* [online]. Project Euclid: 4 January 2011. Dostupné z URL:
<<https://projecteuclid.org/euclid.ss/1294167961#info/>>.
- [3] *Time Series Analysis and Forecasting* [online]. Muhammad Imdad Ullah: December 27, 2013. Dostupné z URL:
<<http://itfeature.com/time-series-analysis-and-forecasting/time-series-analysis-forecasting/>>.
- [4] *Linear Regression Analysis* [online]. From: Methods in Enzymology: 2002. Dostupné z URL:
<<https://www.sciencedirect.com/topics/medicine-and-dentistry/linear-regression-analysis/>>.
- [5] *Logistic Regression* [online]. JAMA: August 2, 2016. Dostupné z URL:
<<https://jamanetwork.com/journals/jama/article-abstract/2540383/>>.
- [6] *A Complete Tutorial on Time Series Modeling in R/* [online]. TAVISH SRIVASTAVA: DECEMBER 16, 2015. Dostupné z URL:
<<https://www.analyticsvidhya.com/blog/2015/12/complete-tutorial-time-series-modeling/>>.
- [7] Miller, Rupert G. *Survival analysis, John Wiley & Sons*. ISBN 0-471-25218-2
- [8] *What is Survival Analysis?/* [online]. Simona Despa. Dostupné z URL:
<<https://www.cscu.cornell.edu/news/statnews/stnews78.pdf>>.
- [9] *Life Table: Meaning, Types and Importance/* [online]. Divisha S. Dostupné z URL:
<<http://www.sociologydiscussion.com/demography/population-growth/life-table-meaning-types-and-importance/3030/>>.
- [10] *Understanding survival analysis: Kaplan-Meier estimate/* [online]. Int J Ayurveda Res.: 2010 Oct-Dec; Dostupné z URL:
<<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3059453/>>.

- [11] *Understanding survival analysis: Kaplan Generalized Linear Models*/ [online]. Germán Rodríguez: 2019; Dostupné z URL:
<<https://data.princeton.edu/wws509/notes/c7s1/>>.
- [12] *The logrank test*/ [online]. J Martin Bland, Douglas G Altman: 2004 May 1; Dostupné z URL:
<<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC403858/>>.
- [13] *Process or Product Monitoring and Control*/ [online]. Engineering Statistic Handbook; Dostupné z URL:
<<https://www.itl.nist.gov/div898/handbook/pmc/pmc.htm>>.
- [14] *Box-Jenkins modelling*/ [online]. Rob J Hyndman: 25 May 2001; Dostupné z URL:
<<https://robjhyndman.com/papers/BoxJenkins.pdf>>.
- [15] *Cox Proportional-Hazards Model*/ [online]. Statistical tools for high-throughput data analysis; Dostupné z URL:
<<http://www.sthda.com/english/wiki/cox-proportional-hazards-model#infos/>>.
- [16] *Induction of Decision Trees* / [online]. Kluwer Academic Publishers, Boston - Manufactured in The Netherland: 1986; Dostupné z URL:
<<https://link.springer.com/content/pdf/10.1007/BF00116251.pdf>>.
- [17] *Decision Trees in Machine Learning* / [online]. Prashant Gupta: May 17, 2017; Dostupné z URL:
<<https://towardsdatascience.com/decision-trees-in-machine-learning-641b9c4e8052/>>.
- [18] *Decision Tree a simple way to visualize a decision* / [online]. Rajesh S. Brid: Oct 26, 2018; Dostupné z URL:
<<https://medium.com/greyatom/decision-trees-a-simple-way-to-visualize-a-decision-dc506a403aeb/>>.
- [19] *Selected Algorithms of Machine Learning from Examples*/ [online]. Jerzy W. GRZYMALA-BUSSE Department of Computer Science, University of Kansas Lawrence: 1993; Dostupné z URL:
<<https://people.eecs.ku.edu/~jerzygb/j24-sel.pdf>>.
- [20] *The Elements of Statistical Learning Data Mining, Inference, and Prediction*/ [online]. Trevor Hastie, Robert Tibshirani, Jerome Friedman: 12th printing -

- Jan, 13 2017; Dostupné z URL:
<<https://web.stanford.edu/~hastie/ElemStatLearn/>>.
- [21] *machine learning (ML)*/ [online]. Margaret Rouse: last updated in May 2018; Dostupné z URL:
<<https://searchenterpriseai.techtarget.com/definition/machine-learning-ML/>>.
- [22] *NEURONOVÉ SÍT Ě 1* / [online]. Eva Volná: Ostrava 2002; Dostupné z URL:
<https://www.vutbr.cz/www_base/priloha.php?dpid=144935/>.
- [23] *A Beginner's Guide to Multilayer Perceptrons (MLP)*/ [online]. Chris Nicholson, skymind; Dostupné z URL:
<<https://skymind.ai/wiki/multilayer-perceptron/>>.
- [24] *Naive Bayes Classifier*/ [online]. Rohith Gandhi: May 5, 2018; Dostupné z URL:
<<https://towardsdatascience.com/naive-bayes-classifier-81d512f50a7c/>>.
- [25] *Introduction to k-Nearest Neighbors: A powerful Machine Learning Algorithm*/ [online]. TAVISH SRIVASTAVA: MARCH 26, 2018; Dostupné z URL:
<<https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/>>.
- [26] *The STURM Group and MAG IAS Group Present Engine Technology of the Future*/ [online]. Engine Builder: APR 22, 2015; Dostupné z URL:
<<https://www.enginebuildermag.com/2015/04/the-sturm-group-and-mag-ias-group-present-engine-technology-of-the-future/>>.
- [27] *Synergies of Pre & Finish Machining on SPECHT HMC Machines Coating Technology Industry 4.0 & MAG*/ [online]. Bernd Kornrumpf: 25.08.2015; Dostupné z URL:
<<https://file.vogel.com.cn/2015/1105/1533561493.pdf>>.
- [28] *Predictive analytics*/ [online]. Wikipedia: 2011; Dostupné z URL:
<https://www.wikiwand.com/en/Predictive_analytics/>.
- [29] *Acute Cardiac Unloading and Recovery - Autumn 2017*/ [online]. Basir et al: 2017; Dostupné z URL:
<<https://www.radcliffecardiology.com/image-gallery/figure-2-kaplan-meier-curves-freedom-death-30-days-device-implanted-prepost-pci/>>.

- [30] *Weight-Selected Attribute Bagging for Credit Scoring - Scientific Figure on ResearchGate/* [online]. Accessed 31 Dec, 2019; Dostupné z URL: [<https://www.researchgate.net/figure/A-linear-SVM-in-2-dimensional-space-with-a-maximummargin-hyperplane_fig15_258392869/>](https://www.researchgate.net/figure/A-linear-SVM-in-2-dimensional-space-with-a-maximummargin-hyperplane_fig15_258392869/).
- [31] *Notes on Artificial Intelligence, Machine Learning and Deep Learning for curious people/* [online]. Özgür Genç: Jan 26; Dostupné z URL: [<https://towardsdatascience.com/notes-on-artificial-intelligence-ai-machine-learning-ml-and-deep-learning-dl-for-56e51a2071c2/>](https://towardsdatascience.com/notes-on-artificial-intelligence-ai-machine-learning-ml-and-deep-learning-dl-for-56e51a2071c2/).
- [32] *Deep Learning/* [online]. MARSHALL HARGRAVE: Apr 30, 2019; Dostupné z URL: [<https://www.investopedia.com/terms/d/deep-learning.asp/>](https://www.investopedia.com/terms/d/deep-learning.asp/).
- [33] *What is Deep Learning?/* [online]. Jason Brownlee: August 16, 2019; Dostupné z URL: [<https://machinelearningmastery.com/what-is-deep-learning/>](https://machinelearningmastery.com/what-is-deep-learning/).
- [34] *Introducing Convolutional Neural Networks in Deep Learning/* [online]. Cyrille Kone: Nov 6, 2019; Dostupné z URL: <https://towardsdatascience.com/introducing-convolutional-neural-networks-in-deep-learning-400f9c3ad5e9>.
- [35] *Understanding LSTM Networks/* [online]. August 27, 2015; Dostupné z URL: <https://colah.github.io/posts/2015-08-Understanding-LSTMs>.
- [36] *Using different recurrent neural network architectures for classifying sequential inputs such as one to many, many to one and sequence to sequence with Long Short Term Memory (LSTM)/* [online]. Jack Dermody: 25 September, 2018; Dostupné z URL: http://www.jackdermody.net/brightwire/article/Sequence_to_Sequence_with_LSTM.
- [37] *Precision and recall/* [online]. By Walber - Own work, CC BY-SA 4.0: 22 November 2014; Dostupné z URL: <https://commons.wikimedia.org/w/index.php?curid=36926283>.
- [38] *Adam: A Method for Stochastic Optimization/* [online]. Diederik P. Kingma, Jimmy Ba: Mon, 30 Jan 2017; Dostupné z URL: <https://arxiv.org/abs/1412.6980>.

Seznam symbolů, veličin a zkratek

f_{vz}	vzorkovací kmitočet
LSTM	Long short term memory
RFID	Radio Frequency Identification
MARS	Multivariate adaptive regression spline
KNS	Konvoluční neuronová síť
a.s	akciová společnost
NOK	Nesouhlasné prvky
OK	Souhlasné prvky
RPM	Rotation per minute
DNN	Deep neural network
TP	True positive
TN	True negative
NN	Neural network
RNN	Recurrent neural network
nos	number of samples
nof	number of features
int	interval
OLS	Ordinary Least Squares
ZKG	Zylinderkurbelgehäuse

Seznam příloh

A První příloha	101
A.1 Globální data	101
A.2 Poslední operace	101
A.3 Lokální data	102
B Druhá příloha	111
B.1 Operace AF110 2x Stroj GROB	111
B.2 Operace AF130 1x Stroj WZ-Bau-Braunschweig - Laseranlage	112
B.3 Operace AF140 2x Stroj WZ-Bau-Braunschweig - Beschichtungsan- lage (Plazmování - potahování)	113
C Program	115
C.1 main.m	115
C.2 data_loading.m	115
C.3 data_preproces.m	115
C.4 data_parsing.m	115
C.5 regression3.m	115
C.6 regr.m	115
C.7 data_gen_rpm.m	115
C.8 data_gen_temp.m	115
C.9 data_shuffler.m	116
C.10 data_print.m	116
C.11 NOK_int2.xlsx	116
C.12 OK_int.xlsx	116
C.13 shuf.xlsx	116
C.14 temp.xlsx	116
C.15 random_stuff.m	116
C.16 DNN_data.m	116
C.17 NOK_generator_RPM	116
C.18 NOK_generator_temp.m	117
C.19 OK_generator_RPM	117
C.20 OK_generator_temp.m	117
C.21 NN.m	117
C.22 NN_results.m	117
C.23 Pregenerated_data_RPM.mat	117
C.24 Pregenerated_data_temp.mat	117
C.25 Pretrained_NN_rpm.m	117

C.26 Pretrained_NN_temp.m	117
D Obsah CD	119

A První příloha

Zde se nachází seznam dat sbíraných pro každý vyráběný kus ukládaná na RFID čip.

A.1 Globální data

1. Datum zadání výrobku
2. Identifikátor továrny
3. Datum
4. Číslo sekvence
5. Závod
6. -
7. Doplnující informace
8. Typové číslo
 - 1= R3 1.0L TSI EVO APS
 - 2= R4 1.5L TSI EVO APS
 - 3= R4 1.4L TSI PHEV
 - 4=
9. Rezerva

A.2 Poslední operace

1. Status kusu (INT)
 - Žádný kus
 - OK kus
 - NOK kus
 - Část měření - Ruční měření s následným zpracováním
 - Část měření - Ruční měření bez dalšího zpracování
 - Část měření - Interval měření
 - Část měření - Výměna nástroje
 - Vyhrazeno pro zobrazení
2. Zvláštní identifikace (WORD)
 - Bit 0.7 DMC vadný
 - Bit 1.3 Připravena zpráva čištění
3. Poslední aplikovaná pracovní sekvence (INT)
4. Stav zpracovávaného dílu (INT)
5. Stav zpracovávaného dílu (INT)
6. Doba výroby (BCD)

7. Protahování (INT)
8. Číslo stroje (INT)
9. Identifikace výrobní linky (INT)
10. Rezerva (ASCII)

A.3 Lokální data

Zde se nachází data z jednotlivých operací, strojů a měřidel.

1. Operace AF100
 - 1 Status kusu (INT)
 - Žádný kus
 - OK kus
 - NOK kus
 - Část měření - Ruční měření s následným zpracováním
 - Část měření - Ruční měření bez dalšího zpracování
 - Část měření - Interval měření
 - Část měření - Výměna nástroje
 - Vyhrazeno pro zobrazení
 - 2 Rezerva (INT)
 - 3 Poslední aplikovaná pracovní sekvence (INT)
 - 4 Stav zpracovávaného dílu (INT)
 - 5 Stav zpracovávaného dílu (INT)
 - 6 Doba výroby (BCD)
 - 7 Upínací proces (INT)
 - 8 Číslo stroje (INT)
 - 9 Identifikace výrobní linky (INT)
 - 10 Rezerva pro údaje specifické pro zákazníka (ASCII)
2. Operace AF110 (Stroj GROB GM4989/01)
 - 1 Status kusu (INT)
 - Žádný kus
 - OK kus
 - NOK kus
 - Část měření - Ruční měření s následným zpracováním
 - Část měření - Ruční měření bez dalšího zpracování
 - Část měření - Interval měření
 - Část měření - Výměna nástroje
 - Vyhrazeno pro zobrazení
 - 2 Rezerva (INT)
 - 3 Poslední aplikovaná pracovní sekvence (INT)

- 4 Stav zpracovávaného dílu (INT)
- 5 Stav zpracovávaného dílu (INT)
- 6 Doba výroby (BCD)
- 7 Protahování (INT)
- 8 Číslo stroje (INT)
- 9 Identifikace výrobní linky (INT)
- 10 Rezerva pro údaje specifické pro zákazníka (ASCII)
- 3. Operace AF120 (Stroj EcoClean - Myčka)
 - 1 AF (INT)
 - 2 Identifikace stroje (INT)
 - 3 Doba trvání zpracování (ASCII)
 - 4
 - 5 Stav součástí kusu (INT)
 - 6 Počet poruch ZKG (INT)
 - 7 Parametr 1 (Teplota praní) (DINT/°C)
 - 8 Doba zapnutí trysek (DINT) v ms
 - 9 Rezerva 4 (DINT)
 - 10 Rezerva 3 (DINT)
 - 11 Rezerva 2 (DINT)
 - 12 Rezerva 1 (DINT)
 - 13 Čas praní (DINT)
 - 14 Počet mycích cyklů (DINT)
 - 15 Čas průtoku (DINT)
- 4. Operace AF130 (Stroj WZ-Bau-Braunschweig - Laserový systém)
 - 1 Stroj / Vřeteno / upínač
 - 2 Doba zpracování
 - 3 Cyklus / Vřeteno
 - 4 Stav zpracovávaného dílu
 - 5 Měření stavu kusu
 - 6 Pořadové číslo pro ZKG
 - 7 Stroj byl v poruše
 - 8 Doba běhu laseru 1
 - 9 Doba běhu rotoru 1
 - 10 Doba běhu zrcadla 1?
 - 11 Celková doba běhu laseru L1 od posledního velkého čištění
 - 12 Celková doba běhu laseru L1 od posledního malého čištění
 - 13 Doba běhu laseru 2
 - 14 Doba běhu rotoru 2
 - 15 Doba běhu zrcadla 2?

- 16 Celková doba běhu laseru 2 od posledního velkého čištění
- 17 Celková doba běhu laseru 2 od posledního malého čištění
- 18 Doba běhu / Parametr 7
- 19 Doba běhu / Parametr 8
- 20 Doba běhu / Parametr 9
- 21 Doba běhu / Parametr 10
- 22 Doba běhu / Parametr 11
- 23 Doba běhu / Parametr 12
- 24 Doba běhu / Parametr 13
- 25 Doba běhu / Parametr 14
- 26 Doba běhu / Parametr 15
- 27 Doba běhu / Parametr 16
- 28 Teplota rotačního pohonu IBK IST
- 29 Mezní hodnota předběžného varování překročení teploty rotačního pohonu IBK
- 30 Varovný limit překročení teploty rotačního pohonu IBK
- 31 Výkon laseru L1 norma
- 32 Výkon laseru L1 minimální
- 33 Výkon laseru L1 maximální
- 34 Rychlost klesání vzduchu válec 1
- 35 Rychlost klesání vzduchu válec 2
- 36 Rychlost klesání vzduchu válec 3
- 37 Rychlost klesání vzduchu válec 4
- 38 Zaostření laseru L1 norma
- 39 Zaostření laseru L1 minimální
- 40 Zaostření laseru L1 maximální
- 41 Proplachovací plyn L1 norma
- 42 Proplachovací plyn L1 skutečný
- 43 Posuv laseru L1 norma (norma == referenční hodnota)
- 44 Posuv laseru L1 skutečný
- 45 Úhel laseru
- 46 Rychlost posuvu laseru L1 referenční
- 47 Rychlost posuvu laseru L1 minimální
- 48 Rychlost posuvu laseru L1 maximální
- 49 Rotační rychlost posuvu laseru L1 referenční
- 50 Rotační rychlost posuvu laseru L1 minimální
- 51 Rotační rychlost posuvu laseru L1 maximální
- 52 Linky na mm
- 53 Povrchová rychlost

- 54 Sesterský stroj
 - 55 Sesterský stroj
 - 56 Počet cyklů od posledního měření výkonu
 - 57 Počet cyklů od posledního měření zaostření
 - 58 Doba trvání cyklu
 - 59 Rezerva
 - 60 Rezerva
 - 61 Rezerva
5. Operace AF 135 (Stroj Strum - Inspekce válců)
- 1 Identifikátor stroje
 - 2 Doba zpracovávání
 - 3 Pracovní cyklus / vřeteno
 - 4 Stav zpracovávaného dílu
 - 5 Teplota kusu
 - 6 Teplota kalibračního disku

Následují hodnoty měření obrobeneho kusu:

- 1 Průměr válce 1, ME-2, hloubka 24mm
 - 2 Průměr válce 1, ME-4, hloubka 70mm
 - 3 Průměr válce 1, ME-5, hloubka 110mm
 - 4 Průměr válce 2, ME-2, hloubka 24mm
 - 5 Průměr válce 2, ME-4, hloubka 70mm
 - 6 Průměr válce 2, ME-5, hloubka 110mm
 - 7 Průměr válce 3, ME-2, hloubka 24mm
 - 8 Průměr válce 3, ME-4, hloubka 70mm
 - 9 Průměr válce 3, ME-5, hloubka 110mm
 - 10 Průměr válce 4, ME-2, hloubka 24mm
 - 11 Průměr válce 4, ME-4, hloubka 70mm
 - 12 Průměr válce 4, ME-5, hloubka 110mm
 - 13 Parametr Rz válce 1 - 0°
 - 14 Parametr Rz válce 2 - 0°
 - 15 Parametr Rz válce 3 - 0°
 - 16 Parametr Rz válce 4 - 0°
 - 17 Doba trvání cyklu měření
 - 18 Rezerva
 - 19 Rezerva
6. Operace AF140 (Stroj WZ-Bau-Braunschweig - lakování)
- 1 Stroj / vřeteno / upínač
 - 2 Dopravník prášku
 - 3 Doba zpracovávání

- 4 Kus / Vřeteno za měřicí linkou
- 5 Pracovní cyklus ZKG / Vřeteno na
- 6 Stav kusu
- 7 Měření stavu kusu
- 8 Stroj byl v poruše
- 9 Doba zapnutí ionizátoru
- 10 Doba zapnutí hořáku 1
- 11 Doba zapnutí Rotaplazma 1
- 12 Doba zapnutí dopravníku prášku 1
- 13 Doba zapnutí dodávkových trubek
- 14 Doba běhu hořáku po velkém čištění
- 15 Doba běhu hořáku po malém čištění
- 16 Doba běhu masky
- 17 Číslo masky
- 18 Počet zapálení hořáku 1
- 19 Teplota zařízení
- 20 Napětí v trubici válce 1
- 21 Proud v trubici válce 1
- 22 Výkon v trubici válce 1
- 23 Čistá energie v trubici válce 1
- 24 Rychlost otáčení hořáku
- 25 Spotřeba hořáku
- 26 Počet vrstev
- 27 Úhel rozstříku v trubici válce 1
- 28 Tlakové chlazení hořáku v trubici válce 1
- 29 Rychlost klesání vzduchu norma
- 30 Rychlost klesání vzduchu minimální (skutečný)
- 31 Rychlost klesání vzduchu maximální (skutečný)
- 32 Tok argonu v trubici válce 1 maximální
- 33 Parametr 1 (tlak / průtok) maximální (skutečný)
- 34 Parametr 1 (tlak / průtok) minimální (skutečný)
- 35 Tok nosného plynu v trubici válce 1
- 36 Parametr 2 (tlak / průtok) maximální (skutečný)
- 37 Parametr 2 (tlak / průtok) minimální (skutečný)
- 38 Tok ochranného plynu 1 v trubici válce 1
- 39 Parametr 3 (tlak / průtok) maximální (skutečný)
- 40 Parametr 3 (tlak / průtok) minimální (skutečný)
- 41 Tok ochranného plynu 2 v trubici válce 1
- 42 Parametr 4 (tlak / průtok) maximální (skutečný)

- 43 Parametr 4 (tlak / průtok) minimální (skutečný)
 - 44 Tok vodíku v trubici válce 1
 - 45 Parametr 5 (tlak / průtok) maximální (skutečný)
 - 46 Parametr 5 (tlak / průtok) minimální (skutečný)
 - 47 Tok chladicí kapaliny v trubici válce 1
 - 48 Parametr 6 (tlak / průtok) maximální (skutečný)
 - 49 Parametr 6 (tlak / průtok) minimální (skutečný)
 - 50 Parametr 7
 - 51 Parametr 8
 - 52 Parametr 9
 - 53 Parametr 10
 - 54 Typ prášku (recept)
 - 55 Dávka prášku
 - 56 Rychlost dodávání prášku do trubice válce 1
 - 57 Tlak prášku v trubici válce 1
 - 58 Vlhkost prášku
 - 59 Indikátor množství na dopravníku prášku
 - 60 Teplota chladicí kapaliny v trubici válce 1 (přítok)
 - 61 Teplota chladicí kapaliny v trubici válce 1 (odtok)
 - 62 Rychlost desky v trubici válce 1
 - 63 Elipsa vlevo těžiště x v trubici válce 1
 - 64 Elipsa vlevo těžiště y v trubici válce 1
 - 65 Elipsa vlevo délka poloosy a v trubici válce 1
 - 66 Elipsa vlevo délka poloosy b v trubici válce 1
 - 67 Elipsa vlevo velikost úhlu alfa v trubici válce 1
 - 68 Elipsa vpravo těžiště x v trubici válce 1
 - 69 Elipsa vpravo těžiště y v trubici válce 1
 - 70 Elipsa vpravo délka poloosy a v trubici válce 1
 - 71 Elipsa vpravo délka poloosy b v trubici válce 1
 - 72 Elipsa vpravo velikost úhlu alfa v trubici válce 1
 - 73 Elipsa vpravo i vlevo, úhel vstříkování v trubici válce 1
 - 74 Doba trvání cyklu
 - 75 Rezerva
 - 76 Rezerva
 - 77 Rezerva
 - 78 Rezerva
 - 79 Rezerva
7. Operace AF145 (Stroj Sturm - Inspekce válců)
- 1 Identifikátor stroje

- 2 Doba zpracování
- 3 Pracovní cyklus / kamera
- 4 Stav zpracovávaného dílu
- 5 Teplota kusu
- 6 Teplota kalibračního disku

Následují hodnoty měření obrobeneho kusu:

- 1 Průměr válce 1, ME-2
 - 2 Průměr válce 1, ME-4
 - 3 Průměr válce 1, ME-5
 - 4 Průměr válce 2, ME-2
 - 5 Průměr válce 2, ME-4
 - 6 Průměr válce 2, ME-5
 - 7 Průměr válce 3, ME-2
 - 8 Průměr válce 3, ME-4
 - 9 Průměr válce 3, ME-5
 - 10 Průměr válce 4, ME-2
 - 11 Průměr válce 4, ME-4
 - 12 Průměr válce 4, ME-5
 - 13 Doba trvání cyklu měření
 - 14 Rezerva
 - 15 Rezerva
 - 16 Rezerva
 - 17 Rezerva
 - 18 Rezerva
 - 19 Rezerva
 - 20 Rezerva
8. Operace AF150 (Stroj EcoClean - Myčka)
- 1 AF (INT)
 - 2 Identifikace stroje (INT)
 - 3 Doba trvání zpracování (ASCII)
 - 4 Stav součástí kusu (INT)
 - 5 Počet poruch ZKG (INT)
 - 6 Parametr 1 (Teplota praní) (DINT/°C)
 - 7 Doba zapnutí trysek 1 (DINT) v ms
 - 8 Doba zapnutí vysokotlakých trysek 2 (DINT) v ms
 - 9 Rezerva 4 (DINT)
 - 10 Rezerva 3 (DINT)
 - 11 Rezerva 2 (DINT)
 - 12 Rezerva 1 (DINT)

- 13 Čas praní (DINT)
 - 14 Počet mycích cyklů (DINT)
 - 15 Čas průtoku (DINT)
9. Operace AF160 (Stroj GROB GM4989/02)
- 1 Status kusu (INT)
 - Žádný kus
 - OK kus
 - NOK kus
 - Část měření - Ruční měření s následným zpracováním
 - Část měření - Ruční měření bez dalšího zpracování
 - Část měření - Interval měření
 - Část měření - Výměna nástroje
 - Vyhrazeno pro zobrazení
 - 2 Rezerva (INT)
 - 3 Poslední aplikovaná pracovní sekvence (INT)
 - 4 Stav zpracovávaného dílu (INT)
 - 5 Stav zpracovávaného dílu (INT)
 - 6 Doba výroby (BCD)
 - 7 Upínací proces (INT)
 - 8 Číslo stroje (INT)
 - 9 Identifikace výrobní linky (INT)
 - 10 Rezerva pro údaje specifické pro zákazníka (ASCII)
10. Operace AF170 (Stroj Gehring - Honování)
- 1 Pracovní sekvence
 - 2 Doba výroby
 - 3 Stav kusu
 - 4 Status vrtání (broušení) válce
 - 5 Bit 0.0 = Válcový otvor 1B201 IO
 - 6 Bit 0.1 = Válcový otvor 1B202 IO
 - 7 Bit 0.2 = Válcový otvor 1B203 IO
 - 8 Bit 0.3 = Válcový otvor 1B204 IO
 - 9 Vrtání válce Č - měřicí rovina 9, DM-1 (osa x)
 - 10 Vrtání válce Č - měřicí rovina 9, DM-2 (osa y)
 - 11 Vrtání válce Č - měřicí rovina 24, DM-1 (osa x)
 - 12 Vrtání válce Č - měřicí rovina 24, DM-2 (osa y)
 - 13 Vrtání válce Č - měřicí rovina 39, DM-1 (osa x)
 - 14 Vrtání válce Č - měřicí rovina 39, DM-2 (osa y)
 - 15 Vrtání válce Č - měřicí rovina 70, DM-1 (osa x)
 - 16 Vrtání válce Č - měřicí rovina 70, DM-2 (osa y)

- 17 Vrtání válce Č - měřící rovina 120, DM-1 (osa x)
 - 18 Vrtání válce Č - měřící rovina 120, DM-2 (osa y)
 - 19 Rezerva
11. Operace AF180 (Stroj Ecoclean - Pračka)
- 1 AF (INT)
 - 2 Identifikace stroje (INT)
 - 3 Doba trvání zpracování (ASCII)
 - 4 Stav součástí kusu (INT)
 - 5 Počet poruch ZKG (INT)
 - 6 Parametr 1 (Teplota praní) (DINT/°C)
 - 7 Parametr 2 (tlak)
 - 8 Parametr 3 (tlak)
 - 9 Vstupní tok (DINT)
 - 10 Doba zapnutí vysokotlakých trysek 3 (Nástroj 1) v ms
 - 11 Doba zapnutí vysokotlakých trysek 4 (Nástroj 2/4) v ms
 - 12 Rezerva 4 (DINT)
 - 13 Rezerva 3 (DINT)
 - 14 Rezerva 2 (DINT)
 - 15 Rezerva 1 (DINT)
 - 16 Čas praní (DINT)
 - 17 Počet mycích cyklů (DINT)
 - 18 Čas průtoku (DINT)
12. Operace AF190 (Stroj Braunschweig - Výstupní kontrola)
- 1 Identifikace stroje
 - 2 Místo konečné kvality
 - 3 Doba zpracování kusu
 - 4 Stav součástí kusu
 - 5 Rezerva
 - 6 Rezerva
13. Operace AF190 (Stroj NEDKONCENE - Laserování a Rozmontování)
- 1 Identifikace stroje
 - 2 Označení kvality (jakosti)
 - 3 Doba zpracování kusu
 - 4 Stav součástí kusu
 - 5 Rezerva
 - 6 Rezerva
14. Frei

B Druhá příloha

V následující tabulce se nachází jednotlivé parametry, které nejvíce ovlivňují kvalitu finálního kusu. Označení DB v tabulkách označuje identifikátory v databázové struktuře, kde se data ukládají.

B.1 Operace AF110 2x Stroj GROB

Parametr	Stav / Zdroj dat
Stroj - Vypnuto	Bez napájení, PLC zapnuto
Stroj - zapnuto	Napájení zapnuto, žádná chyba, bez výroby
Stroj - automat	d
Prázdňý vstup - automat	Kus není k dispozici, žádná chyba, automatický režim
Plný výstup	Plně obsazený výstup, žádná chyba, automatický režim
Stroj - seřízení	Nastavení stavu stroje / systému
Total stop	Total stop
Stroj - Porucha	Automatický provoz není možný, porucha
Počet OK kusy	Hodnota
Počet NOK kusy	Hodnota
DMC	RFID, DMS, identifikace obrobku
Takt	Hodnota
Plánovaný takt	Hodnota(pokud nepochází z plánovacího systému)
Takt	Hodnota
Nástroj č.	Hodnota (jako řetězec)
Životnost nástroj	Hodnota
Počet vyrobených kusů	Hodnota
Otáčky (RPM)	Hodnota
Posuv	Hodnota
	Teploty pohonů

B.2 Operace AF130 1x Stroj WZ-Bau-Braunschweig - Laseranlage

Parametr	Stav / Zdroj dat	
Čas obrábění	viz. RFID	
DMC kód (32 znaků)	viz. RFID	
Provozní stavy stroje		
	Laser 1	Laser 2
Životnost optiky	DB770.DBx10	DB770.DBx140
Odsávání	DB750.DBx0	DB750.DBx4
Výkon laseru	DB220.DBx2020	DB220.DBx2022
Otáčky laseru	DB716.DBx104	DB716.DBx304
Posuv		

B.3 Operace AF140 2x Stroj WZ-Bau-Braunschweig - Beschichtungsanlage (Plazmování - potahování)

Parametr	Stav / Zdroj dat	
DMC kód (32 znaků)	viz. RFID	
Provozní stavy stroje		
	Hořák 1	Hořák 2
Stroj / Vřeteno / upínač	viz RFID	viz RFID
Zásobník prášku	viz RFID	viz RFID
Čas operace	viz RFID	viz RFID
Díl/Vřeteno - po měření	viz RFID	viz RFID
Cyklus	viz RFID	viz RFID
Status dílu	viz RFID	viz RFID
Měřený díl	viz RFID	viz RFID
Stav stroje (automat vs ma- nual)	viz RFID	viz RFID
Životnost Ionizéru	DB771.DBD4090	DB771.DBD4090
Životnost hořáku 1	DB771.DBD4094	DB771.DBD4094
Životnost rotaplasmy 1	DB771.DBD4098	DB771.DBD4098
Životnost zásobníku prášku 1	DB771.DBD4102	DB771.DBD4102
Životnost přívodní trubice	DB771.DBD4106	DB771.DBD4106
Životnost B1 po velkém čiš- tění	DB771.DBD4110	DB771.DBD4110
Životnost B2 po malém čiš- tění	DB771.DBD4114	DB771.DBD4114
Životnost maskování	DB771.DBD4114	DB771.DBD4114
Počet zapálení hořáku 1	DB721.DBD308	DB721.DBD808
Upínač	DB721.DBD16	DB721.DBD516
Intenzita proudu	DB721.DBD0	DB721.DBD500
Výkon 1	DB721.DBD32	DB721.DBD532
Čistá energie 1	DB721.DBD312	DB721.DBD812
Otáčky hořáku 1	DB751.DBD24	DB751.DBD24
Posuv hořáku 1	DB751.DBD36	DB751.DBD36
Počet cyklů	DB751.DBD12	DB751.DBD12
Úhel nástřiku	DB649.DBD3520	DB650.DBD3520
Chladicí vzduch v hořáku (AirJet)	DB721.DBD96	DB721.DBD596

Odsávání nominál	DB750.DBD126	DB750.DBD140
Odsávání L1 min realita	DB900.DBD22	DB900.DBD42
Odsávání L1max realita	DB900.DBD62	DB900.DBD82
Argon (Procesní plyn)	DB721.DBD48	DB721.DBD548
Nosný plyn	DB721.DBD160	DB721.DBD660
Plášťový plyn	DB721.DBD64	DB721.DBD564
Vodík	DB721.DBD344	DB721.DBD844
Chladicí voda	DB721.DBD144	DB721.DBD644
Recept	DB717.DBW0	DB717.DBW2
Dopravník prášku (rychlost)	DB721.DBD176	DB721.DBD676
Dopravník prášku (tlak)	DB721.DBD240	DB721.DBD740
Otáčky disku zásobníku prášku	DB721.DBD192	DB721.DBD692
Dopravník prášku (hladina)	DB721.DBD224	DB721.DBD724
Teplota chladicí vody na vstupu	DB721.DBD360	DB721.DBD860
Teplota chladicí vody na výstupu	DB721.DBD128	DB721.DBD628
Rychlost odsávání na talíři	DB721.DBD208	DB721.DBD708
Elipsa - levá - těžiště x	DB649.DBD3460	DB650.DBD3460
Elipsa - levá - těžiště y	DB649.DBD3466	DB650.DBD3466
Elipsa - levá - délka poloosy a	DB649.DBD3472	DB650.DBD3472
Elipsa - levá - délka poloosy b	DB649.DBD3478	DB650.DBD3478
Elipsa - levá - úhel alfa	DB649.DBD3484	DB650.DBD3484
Elipsa - pravá - těžiště x	DB649.DBD3490	DB650.DBD3490
Elipsa - pravá - těžiště y	DB649.DBD3496	DB650.DBD3496
Elipsa - pravá - délka poloosy a	DB649.DBD3502	DB650.DBD3502
Elipsa - pravá - délka poloosy b	DB649.DBD3508	DB650.DBD3508
Elipsa - pravá - úhel alfa	DB649.DBD3514	DB650.DBD3514
Elipsa - úhel nástřiku - levá - pravá	DB649.DBD3520	DB650.DBD3520
Čas cyklu	viz. RFID	viz. RFID
Ostatní parametry Oerlikon	viz. Oerlikon	viz. Oerlikon

C Program

V této kapitole se nachází seznam jednotlivých příložených souborů k práci s odkazem na jejich detailní popis. Jedná se o rozcestník pro jednodušší orientaci v práci.

C.1 main.m

Hlavní skript programu 3.2.

C.2 data_loading.m

Skript realizující načítání dat 3.3.1.

C.3 data_preproces.m

Skript pro předzpracování dat 3.3.2.

C.4 data_parsing.m

Skript rozdělující data na intervaly 3.3.5.

C.5 regression3.m

Skript realizující metodu regresní analýzy 3.3.6.

C.6 regr.m

Pomocná funkce pro metodu regresní analýzy 3.3.7.

C.7 data_gen_rpm.m

Skript generující data rychlosti otáček pro metodu regresní analýzy 3.3.3.

C.8 data_gen_temp.m

Skript generující data teploty pro metodu regresní analýzy 3.3.4.

C.9 data_shuffler.m

Pomocný skript pro generování dat pro metodu regresní analýzy 3.4.3.

C.10 data_print.m

Skript realizující zobrazení výsledků metody 3.3.8.

C.11 NOK_int2.xlsx

Soubor excelu obsahující NOK intervaly.

C.12 OK_int.xlsx

Soubor excelu obsahující OK intervaly.

C.13 shuf.xlsx

Soubor excelu pouze OK data.

C.14 temp.xlsx

Soubor excelu obsahující NOK i OK data.

C.15 random_stuff.m

Skript obsahující pomocné funkce pro zobrazení grafů a výsledků bez nutnosti spuštění hlavního programu.

C.16 DNN_data.m

Skript tvořící datasety pro neuronovou síť 3.4.1.

C.17 NOK_generator_RPM

Skript generující NOK data rychlosti otáček 3.4.4.

C.18 NOK_generator_temp.m

Skript generující NOK data teploty 3.4.4.

C.19 OK_generator_RPM

Skript generující OK data rychlosti otáček 3.4.2.

C.20 OK_generator_temp.m

Skript generující OK data teploty 3.4.2.

C.21 NN.m

Skript obsahující parametry potřebné pro trénování neuronové sítě 3.4.5.

C.22 NN_results.m

Skript zobrazující výsledky predikce neuronové sítě 3.4.7.

C.23 Pregenerated_data_RPM.mat

Předgenerovaná data rychlosti otáček pro metodu neuronové sítě.

C.24 Pregenerated_data_temp.mat

Předgenerovaná data teploty pro metodu neuronové sítě.

C.25 Pretrained_NN_rpm.m

Předtrénovaná neuronová síť pro predikci rychlosti otáček.

C.26 Pretrained_NN_temp.m

Předtrénovaná neuronová síť pro predikci teploty.

D Obsah CD

Kód na CD byl spouštěn pouze v MATLABu verze 2020a. Žádná jiná verze nebyla testována.

/	Hlavní složka CD
├── Program.....	Složka obsahující program
│ ├── data_gen_rpm.m	
│ ├── data_gen_temp.m	
│ ├── data_loading.m	
│ ├── data_parsing.m	
│ ├── data_preproces.m	
│ ├── data_print.m	
│ ├── data_shuffler.m	
│ ├── DNN_data.m	
│ ├── main.m	
│ ├── NN.m	
│ ├── NN_results.m	
│ ├── NOK_generator_RPM.m	
│ ├── NOK_generator_temp.m	
│ ├── NOK_int2.xlsx	
│ ├── OK_generator_RPM.m	
│ ├── OK_generator_temp.m	
│ ├── OK_int.xlsx	
│ ├── Pregenerated_data_RPM.mat	
│ ├── Pregenerated_data_temp.mat	
│ ├── Pretrained_NN_rpm.mat	
│ ├── Pretrained_NN_temp.mat	
│ ├── random_stuff.m	
│ ├── regr.m	
│ ├── regression3.m	
│ ├── shuf.xlsx	
│ └── temp.xlsx	
├── Data	Složka obsahující data
│ ├── data_plazma_M6	Data ze serveru
│ └── GM4989_Balluff_Datenträgerbelegung_V1.3.....	Data z RFID čipu
└── Vasko_PredictiveAnalysis.pdf.....	Diplomová práce