

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE
Fakulta strojní



**Dopředné a rekurentní neuronové sítě pro
predikci teploty mezi řezy turbopropulvého
motoru**

Feedforward and Recurrent Neural Network for Temperature Prediction
Between Cuts of Turbo Propeller Engine

DIPLOMOVÁ PRÁCE

DIPLOMA THESIS

Bc. Josef Řehořík
AUTOR / AUTHOR

doc. Ing. Ivo Bukovský, Ph.D.
VEDOUCÍ PRÁCE / SUPERVISOR

Praha, 2019

I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Řehořík** Jméno: **Josef** Osobní číslo: **424670**
Fakulta/ústav: **Fakulta strojní**
Zadávací katedra/ústav: **Ústav mechaniky, biomechaniky a mechatroniky**
Studijní program: **Průmysl 4.0**
Studijní obor: **bez oboru**

II. ÚDAJE K DIPLOMOVÉ PRÁCI

Název diplomové práce:

Dopředné a rekurentní neuronové sítě pro predikci teploty mezi řezy turbovrtulového motoru

Název diplomové práce anglicky:

Feedforward and Recurrent Neural Networks for Temperature Prediction Between Cuts of Turbo-propeller Engine

Pokyny pro vypracování:

1. Proveďte rešerši problematiky analýzy závislostí (včetně predikce) a chování teplot mezi různými částmi malých leteckých turbomotorů. 2. Proveďte rešerši metod pro výpočet teplot v řezech motoru z měřených hodnot v jiných řezech motoru (dopředné a rekurentní neuronové sítě, např. HONU, MLP, ELM, RVFL, LSTM,...). 3. Analyzujte měřená data. 4. Navrhněte metodu výpočtu (predikce) alespoň jedné teploty v co nejvíce řezech motoru z minimálního počtu teplotních sensorů z jiných řezů motoru neuronovými sítěmi. 5. Proveďte experimenty na umělých i měřených datech a vyhodnoťte přesnost a robustnost učících algoritmů (i vůči šumu) pro jednotlivé teploty mezi různými řezy. 6. Metodu a experimenty zdokumentujte vyhodnoťte.

Seznam doporučené literatury:

[1] SUGANTHAN, Ponnuthurai Nagaratnam. Letter: On non-iterative learning algorithms with closed-form solution. Applied Soft Computing [online]. 2018, 70, 1078–1082. ISSN 15684946. Dostupné z: doi:10.1016/j.asoc.2018.07.013
[2] BUKOVSKY, I. a N. HOMMA. An Approach to Stable Gradient-Descent Adaptation of Higher Order Neural Units. IEEE Transactions on Neural Networks and Learning Systems [online]. 2017, 28(9), 2022–2034. ISSN 2162-237X. Dostupné z: doi:10.1109/TNNLS.2016.2572310

Jméno a pracoviště vedoucí(ho) diplomové práce:

doc. Ing. Ivo Bukovský, Ph.D., U12110.3

Jméno a pracoviště druhé(ho) vedoucí(ho) nebo konzultanta(ky) diplomové práce:

Datum zadání diplomové práce: **29.04.2019**

Termín odevzdání diplomové práce: **16.08.2019**

Platnost zadání diplomové práce: _____

doc. Ing. Ivo Bukovský, Ph.D.
podpis vedoucí(ho) práce

prof. Ing. Milan Růžička, CSc.
podpis vedoucí(ho) ústavu/katedry

prof. Ing. Michael Valášek, DrSc.
podpis děkana(ky)

III. PŘEVZETÍ ZADÁNÍ

Diplomant bere na vědomí, že je povinen vypracovat diplomovou práci samostatně, bez cizí pomoci, s výjimkou poskytnutých konzultací. Seznam použité literatury, jiných pramenů a jmen konzultantů je třeba uvést v diplomové práci.

Datum převzetí zadání

Podpis studenta

Čestné prohlášení

Prohlašuji, že jsem svou práci vypracoval samostatně, pouze pod odborným vedením pana doc. Ing. Iva Bukovského, Ph.D. Dále prohlašuji, že veškeré podklady a zdroje ze kterých jsem čerpal, jsou uvedeny v seznamu použité literatury.

V Praze, dne 12.7.2019

.....

Poděkování

Děkuji panu doc. Ing. Ivu Bukovskému, Ph.D. za jeho cenné rady, připomínky a čas strávený vedením této práce. Děkuji také své rodině, která mě vždy podporovala.

Abstrakt

Tato diplomová práce se věnuje predikci teploty, která je skalární termodynamickou stavovou veličinou. K predikci budou využity umělé neuronové sítě (*Artificial Neural Networks, ANN*), které jsou jedním z nástrojů umělé inteligence (*Artificial Intelligence, AI*). Tuto veličinu budu analyzovat a následně predikovat z dat naměřených v turbovrtulovém letadlovém motoru. Data, která budou v této práci zpracována a vyhodnocována pochází od blíže nespecifikovaného, avšak relevantního zdroje, který provedl měření teplot propulzní látky tohoto motoru. Část této práce se také věnuje termodynamickým dějům, na jejichž principech tyto stroje pracují. Pro zpracování velkého množství dat, která jsou v této práci použita, bude využita mimo jiné analýza hlavních komponent (*PCA*). Pro dosažení ideálního výsledku práce použiji různé kombinace modelů sítí a optimalizačních algoritmů.

Klíčová slova

Back Propagation, gradientní metody, neuron, neuronová síť, optimalizační algoritmus, predikce, propulzní látka, propulzní systém, somatická operace, synaptická operace, turbovrtulový letadlový motor

Abstract

This thesis deals with temperature prediction, which is a scalar thermodynamic state variable. Artificial Neural Networks (ANN), which are one of the tools of Artificial Intelligence (AI), will be used for prediction. I will analyze this quantity and then predict it from the data measured in a turboprop aircraft engine. The data that will be processed and evaluated in this work comes from an unspecified but relevant source, which performed the measurement of the temperature of the propulsion substance of this engine. Part of this work is also devoted to thermodynamic processes on which these machines work. The Principal Component Analysis (PCA) will be used to process large amounts of data used in this work besides other methods. I will use various combinations of neural network models and optimization algorithms to achieve ideal results.

Keywords

Back Propagation, gradient methods, neural network, neuron, optimization algorithm, prediction, propulsion substance, propulsion system, somatic operation, synaptic operation, Turboprop aircraft engine

Seznam obrázků

Obr. 2.1: Uspořádání neuronů do tří vrstev dopředné neuronové sítě [5]	3
Obr. 2.2: Model neuronu typu Perceptron [5].....	4
Obr. 2.3: Graf růstu počtu vah v závislosti na stupni polynomu HONU [10]	6
Obr. 2.4: Průběhy vybraných aktivačních funkcí a jejich derivace [10].....	8
Obr. 2.5: Minimalizace chybové funkce při adaptaci [8]	10
Obr. 2.6: Porovnání metod SGD a BGD [10]	13
Obr. 3.1: Nalevo: termoelektrický snímač teploty (a - tyčový, b - do jímky, c – plášťový kabel), napravo: plášťové termoelektrické články [14]	20
Obr. 3.2: Aerodynamická letová obálka – limity aerodynamického letu [17].....	21
Obr. 3.3: Oběhy v p-v a T-s diagramu [17]	23
Obr. 3.4: Řez 3D modelem turbovrtulového motoru [25]	24
Obr. 4.1: Komponenty turbovrtulového motoru v řezu [26].....	31
Obr. 4.2: Polohy jednotlivých senzorů v řezu TPL1.....	33
Obr. 4.3: Polohy jednotlivých senzorů v řezu TPL2.....	33
Obr. 4.4: Zastoupení jednotlivých senzorů ve všech komponentách řezu TPL1.....	35
Obr. 4.5: Důležitost jednotlivých komponent řezu TPL1	35
Obr. 4.6: Zastoupení jednotlivých senzorů ve všech komponentách řezu TPL2.....	36
Obr. 4.7: Důležitost jednotlivých komponent řezu TPL2	37
Obr. 4.8: Korelační matice vizualizovaná pomocí teplotní mapy.....	38
Obr. 5.1: Porovnání výsledků predikce senzorů sítí MLP	41
Obr. 5.2: Architektura ELM sítě	42
Obr. 5.3: Porovnání výsledků predikce senzorů sítí ELM.....	43
Obr. 5.4: Architektura RVFL sítě	44
Obr. 5.5: Porovnání výsledků predikce senzorů sítí RVFL	46
Obr. 5.6: Porovnání výsledků predikce senzorů sítí LNU	47
Obr. 5.7: Porovnání výsledků predikce senzorů sítí QNU.....	49
Obr. 6.1: Graf průběhu teploty reálné a predikované 34. senzoru	51
Obr. 6.2: Graf průběhu teploty reálné a predikované 7. senzoru	51
Obr. 6.3: Graf průběhu teploty reálné a predikované 33. senzoru	52
Obr. 6.4: Graf průběhu teploty reálné a predikované 26. senzoru	52
Obr. 6.5: Graf průběhu teploty reálné a predikované 17. senzoru	53

Obr. 6.6: Graf průběhu teploty reálné a predikované 33. senzoru	53
Obr. 6.7: Graf průběhu teploty reálné a predikované 26. senzoru	54
Obr. 6.8: Graf průběhu teploty reálné a predikované 26. senzoru	54
Obr. 6.9: Graf průběhu teploty reálné a predikované 26. senzoru	55
Obr. 6.10: Graf průběhu teploty reálné a predikované 26. senzoru	55
Obr. 6.11: Graf průběhu teploty reálné a predikované 17. senzoru	56
Obr. 6.12: Graf průběhu teploty reálné a predikované 33. senzoru	56
Obr. 6.13: Graf průběhu teploty reálné a predikované 25. senzoru	57
Obr. 6.14: Graf průběhu teploty reálné a predikované 18. senzoru	57
Obr. 6.15: Graf průběhu teploty reálné a predikované 33. senzoru	58

Seznam tabulek

Tab. 4.1: Seznam senzorů z řezu TPL1	32
Tab. 4.2: Seznam senzorů z řezu TPL2	32
Tab. 5.1: Kvalita predikce sítí MLP podle maximální chyby	40
Tab. 5.2: Kvality predikce sítí MLP podle MSE	40
Tab. 5.3: Kvality predikce sítí MLP podle směrodatné odchylky	41
Tab. 5.4: Kvalita predikce sítí ELM podle maximální chyby	43
Tab. 5.5: Kvalita predikce sítí ELM podle MSE	43
Tab. 5.6: Kvalita predikce sítí ELM podle směrodatné odchylky	43
Tab. 5.7: Kvalita predikce sítí RVFL podle maximální chyby	45
Tab. 5.8: Kvalita predikce sítí RVFL podle MSE.....	45
Tab. 5.9: Kvalita predikce sítí RVFL podle směrodatné odchylky	45
Tab. 5.10: Kvalita predikce sítí LNU podle maximální chyby	47
Tab. 5.11: Kvalita predikce sítí LNU podle MSE.....	47
Tab. 5.12: Kvalita predikce sítí LNU podle směrodatné odchylky	47
Tab. 5.13: Kvalita predikce sítí QNU podle maximální chyby	48
Tab. 5.14: Kvalita predikce sítí QNU podle MSE	48
Tab. 5.15: Kvalita predikce sítí QNU podle směrodatné odchylky	48
Tab. 6.1: Dosažené průměrné hodnoty aplikovaných modelů sítí	58

Nomenklatura

AI	Artificial Intelligence (umělá inteligence)
ANN	Artificial Neural Network (umělá neuronová síť)
BP	Back Propagation (zpětné šíření chyby)
CNU	Cubic Neural Unite (třetí stupeň polynomu neuronové jednotky)
ELM	Extreme Learning Machine
GA	Genetic Algorithm (genetický algoritmus)
GD	Gradient Descent
GNA	Gauss - Newtonova metoda
HONU	Higher Order Neural Unite (neuronová jednotka vyššího stupně)
LMA	Levenberg - Marquardtova metoda
LNU	Linear Neural Unite (první stupeň polynomu neuronové jednotky)
MSE	Mean Squared Error (střední kvadratická chyba)
ML	Machine Learning (strojové učení)
MLP	Multi Layer Perceptron (vícevrstvý perceptron)
PCA	Principal Component Analysis (analýza hlavních komponent)
QNU	Quadratic Neural Unite (druhý stupeň polynomu neuronové jednotky)
ReLU	Rectified Linear Unite
Turboprop	Turbo Propeller (turbovrtulový letadlový motor)
$\sigma(\xi)$	Aktivační funkce (somatická operace)
ξ	Vnitřní potenciál neuronu
Θ	Bias (prahová hodnota)
μ	Learning Rate (parametr učení)

Obsah

1	Úvod	1
2	Neuronové sítě	3
2.1	<i>Neuron</i>	3
2.2	<i>Topologie, architektura a modely neuronových sítí</i>	6
2.3	<i>Aktivační funkce</i>	7
2.4	<i>Optimalizační algoritmy</i>	9
2.4.1	<i>Dávkové metody</i>	11
2.4.1.1	<i>Adam optimizer</i>	11
2.4.1.2	<i>Batch Gradient Descent (BGD)</i>	11
2.4.1.3	<i>Stochastic Gradient Descent (SGD)</i>	12
2.4.1.4	<i>Levenberg-Marquardtova metoda (LMA)</i>	13
2.4.2	<i>Krokové metody</i>	14
2.4.2.1	<i>Gradient Descent</i>	14
2.5	<i>Algoritmus zpětného šíření chyby</i>	14
3	Rešerše termodynamických dějů v letadlových motorech a jejich predikce	18
3.1	<i>Měření teploty v technické praxi</i>	18
3.1.1	<i>Teplota a teplota</i>	18
3.1.2	<i>Chyby a nejistoty měření</i>	18
3.1.3	<i>Plášťové termoelektrické články</i>	19
3.1.4	<i>Seebeckův jev</i>	20
3.2	<i>Letadlové motory</i>	21
3.2.1	<i>Teorie tepelných motorů</i>	22
3.2.2	<i>Turbovrtulový motor</i>	23
3.3	<i>Rešerše využití a aplikací pro predikci neuronovými sítěmi</i>	25
3.3.1	<i>Řešené problematiky v oblasti lopatkových strojů prostřednictvím ANN</i>	25
3.3.2	<i>Shrnutí poznatků z výzkumů a studií v oblasti lopatkových motorů s využitím ANN</i>	30
4	Datová analýza, příprava a zpracování dat z motoru Turboprop	31
4.1	<i>Analýza hlavních komponent – PCA</i>	34
4.2	<i>Korelace senzorů měření</i>	37
5	Experimentální analýza a návrh neuronových sítí pro predikci teplot	39
5.1	<i>LSTM (Long-Short Term Memory)</i>	39
5.2	<i>MLP (Multi Layer Perceptron)</i>	40
5.3	<i>ELM (Extreme Learning Machine)</i>	41
5.4	<i>RVFL (Random Vector Function Link)</i>	44
5.5	<i>HONU (Higher Order Neural Unite)</i>	46
5.5.1	<i>LNU</i>	46
5.5.2	<i>QNU</i>	48
6	Aplikace a zhodnocení navržených modelů sítí	50
6.1	<i>Vyhodnocení modelu MLP</i>	50

6.2	<i>Vyhodnocení modelu ELM</i>	52
6.3	<i>Vyhodnocení modelu RVFL</i>	53
6.4	<i>Vyhodnocení modelu LNU</i>	55
6.5	<i>Vyhodnocení modelu QNU</i>	56
6.6	<i>Zhodnocení použitých neuronových sítí</i>	58
7	Závěr	60
	Seznam použité literatury	61
	Přílohy	65

1 Úvod

V roce 1900 David Hilbert zformuloval otázku rozhodovacího problému, kde se ptal, zdali existuje postup, který by jednoznačně zaručil platnost matematických axiomů. Toto své přesvědčení dál rozvíjel, a tak se později pokusil sestavit soubor axiomů, kterými by dokázal všechny matematické věty. Tuto domněnku však v roce 1930 vyvrátil Kurt Gödel, když dokázal, že takový algoritmus je nedosažitelný. Argumentem jeho důkazu bylo, že neexistuje systém obsahující elementární aritmetiku, který by byl úplný a současně bezesporný. Tedy takový, který by bylo možné jednoznačně dokázat axiomy. Stále však zůstávala naděje, že pro ty axiomy, které dokázat lze, existuje způsob, jak je mechanicky odvodit. V roce 1937 však Alan Turing přišel se svým matematickým strojem, který tuto teorii vyvrátil. Názorovým vyústěním tedy tohoto období bylo, že neexistuje způsob jak logicky, matematicky a komplexně popsat svět, tak jak jej známe.

Když v roce 1943 Warren McCulloch a Walter Pitts vytvořili první umělý neuron, začala se psát éra nového oboru, tedy umělé inteligence (*Artificial Intelligence, AI*). To inspirovalo Donalda Hebba k sepsání knihy „*The Organization of Behaviour*“, ve které zformuloval učící pravidlo mezi dvěma neurony (*Hebbův princip*) popsaný vztahem:

$$\Delta w_{i,j} = \eta \cdot x_i \cdot y_j, \quad (1)$$

kde $\Delta w_{i,j}$ je změnou váhy mezi neurony i a j , x_i je vstupem neuronu, y_j je výstupem neuronu a η je koeficient učení (*Learning Rate*)

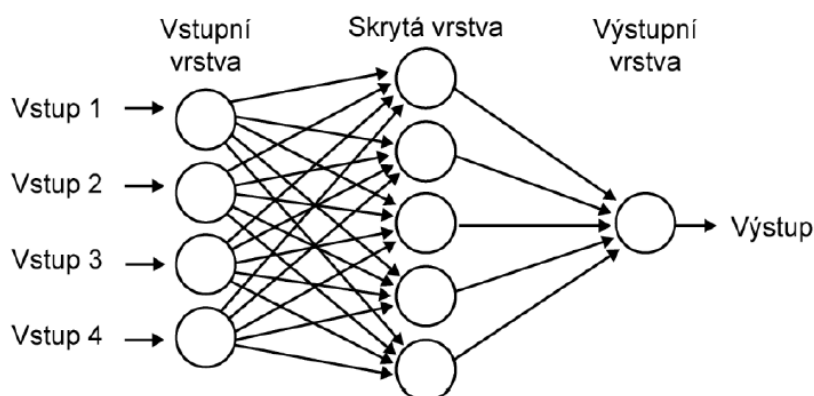
Během 40. a 50. let však k žádnému výraznému pokroku nedošlo. Sice se v roce 1951 podařilo Marvinu Minskému sestrojít první neuropočítač Snark, jenž uměl adaptovat váhy, k řešení praktického problému ale aplikován nebyl. Snark však v roce 1957 inspiroval Franka Rosenblatta, který vynalezl zobecnění McCullochova a Pittsova modelu neuronu, tedy perceptron. Navrhl pro něj učící algoritmus, který po konečném počtu pokusů nalezne adekvátní váhový vektor nezávisle na jeho počátečním stavu. Roku 1958 Rosenblatt s Charlesem Wightmanem a dalšími sestrojili konečně první úspěšný neuropočítač „*Mark I Perceptron*“.

Dalším typem neuronového počítače byl „ADALINE“ (*ADaptive Linear NEuron*). Ten roku 1960 sestrojil Bernard Widrow se svými studenty. Funkčnost tohoto zařízení byla demonstrována na jednoduchých typových příkladech. I to bylo příčinou, že na přelomu 50. a 60 let došlo k úspěšnému rozvoji neurovýpočtů, k návrhu nových neuronových sítí a jejich implementaci. V následujícím období však následovala doba, kdy se začalo jevit, že obor neuronových sítí je již vyčerpán a další vývoj v této oblasti by vyžadoval nový přístup. Následkem toho se stalo, že nejlepší kapacity v oboru se začali zaměřovat na další obory příbuzné umělé inteligenci. Mimo jiné i proto, že jednoduchá logická funkce, tzv. vylučovací disjunkce (*XOR*), jelikož není lineárně separabilní, nemůže být vypočtena jedním perceptronem. Šlo by sice tento problém vyřešit dvouvrstvou sítí se třemi neurony, v té době ale nebyl znám učící algoritmus pro vícevrstvý perceptron. Zvrat pak nastal v roce 1983, kdy americká grantová agentura DARPA začala finančně podporovat výzkum neuronových sítí, stejně tak jako později i další organizace podporující základní a aplikovaný výzkum. O své se zasloužil i John Hopfield. Roku 1982 a 1984 publikoval závěry své práce, jež vedly k přitáhnutí mnoha kvalifikovaných vědců, matematiků a technologů zpět do oblasti neuronových sítí. Návazně na to v roce 1986 vyšla publikace výsledků práce, jejíž podstatou bylo vyřešení problému nepřekonatelné překážky hlubšího využití a dalšího rozvoje neuronových sítí, kterou v 60. letech definovali Minský s Pappertem. Řešením byl učící algoritmus zpětného šíření chyby (*Back Propagation, BP*).

2 Neuronové sítě

Stejně jako lidský mozek, tak i umělé neuronové sítě mají schopnost učit se. To je do jisté míry dáno i tím, že inspirací jejich podoby byly právě ty, které tvoří centrální nervovou soustavu lidského těla. Tyto sítě se tedy snaží modelovat právě ty biologické, proto je nazýváme taktéž neuronovými sítěmi. Jsou to sítě sestavené z mnoha jednoduchých procesorů, které označujeme jako neuronové jednotky, jež jsou vzájemně propojeny. Graf propojení těchto výpočetních jednotek se často označuje jako topologie sítě. [7].

Můžeme tedy říci, že umělou neuronovou síť tvoří matematické neurony uspořádané do orientované struktury pomocí ohodnocených spojů, kde každý vstup zpracovává vstupní signály ohodnocené dle vlastních vah, které jsou generovány na výstup. Každá neuronová síť je mimo jiné charakterizována typem neuronů, jejich topologickým uspořádáním a optimalizačním algoritmem [5].



Obr. 2.1: Uspořádání neuronů do tří vrstev dopředné neuronové sítě [5]

2.1 Neuron

Neuron je základní stavební jednotka neuronových sítí v podobě matematického modelu. Formálně má neuron n vstupů (*axony*) x_1, x_2, \dots, x_n , které jsou ohodnoceny (*synaptickými*) váhami w_1, w_2, \dots, w_n , jež určují jejich propustnost. Neuron má jediný výstup, který může být přiveden na vstup mnoha dalších procesorů, nebo vyveden do vnějšího okolí. Vnitřním potenciálem neuronu ξ (*synaptickou operací*) tedy v případě Perceptronu rozumíme hodnotu vyjádřenou vztahem:

$$\xi = \sum_{i=1}^n (w_i \cdot x_i) - v, \quad (2)$$

kde v je prahová hodnota (*bias*), x je vektor vstupů ve tvaru $[x_1, x_2, \dots, x_n]^T$ a w je vektor vah ve tvaru $[w_1, w_2, \dots, w_n]$. V některých případech se vyplatí považovat práh za speciální případ váhy spoje vedoucího do neuronu, jehož výstup je trvale -1 . Pak platí:

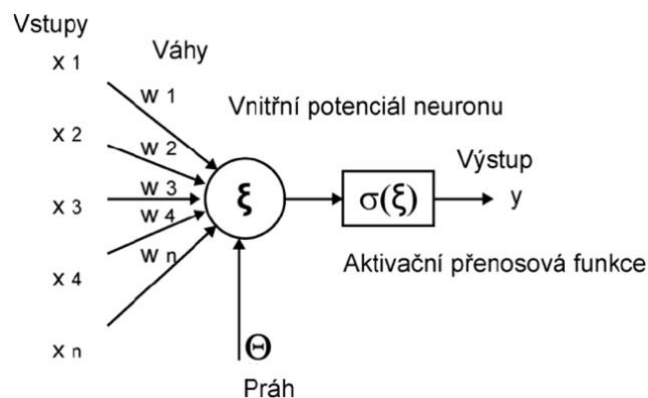
$$\theta = w_0 \text{ a } x_0 = -1. \quad (3)$$

Výstup se často počítá pomocí aktivační přenosové funkce (*somatická operace*) $\sigma(\xi)$, které budou dále popsány v kapitole 1.3, v případě sigmoidy pak ve tvaru:

$$\sigma(\xi) = \frac{1}{1 + e^{-\xi}} = y, \quad (4)$$

kde e je Eulerovo číslo, a y je hodnota výstupu neuronu, tedy výsledek somatické operace. Typů aktivačních funkcí je k dispozici více, jak bude popsáno v následující části této práce.

Nejstarším a nejzákladnějším modelem neuronu je tzv. Perceptron. Model neuronové sítě tvořený tímto neuronem je jednovrstvý, kdy jde o paralelně pracující procesory. Jinými slovy každý z těchto neuronů provádí transformaci vstupního vektoru na výstupní hodnotu, a to nezávisle na neuronech ostatních [8].



Obr. 2.2: Model neuronu typu Perceptron [5]

Dalším typem neuronu může být např. neuronová jednotka vyššího stupně (*Higher Order Neural Unit, HONU*), kdy jejich výstup vypočítáme vztahem:

$$y = \sigma \left(\theta_0 + \sum_{i_1}^n w_{i_1} x_{i_1} + \sum_{i_1, i_2}^n w_{i_1, i_2} x_{i_1} x_{i_2} + \dots + \sum_{i_1, \dots, i_N}^n w_{i_1, \dots, i_N} x_{i_1} \dots x_{i_N} \right), \quad (5)$$

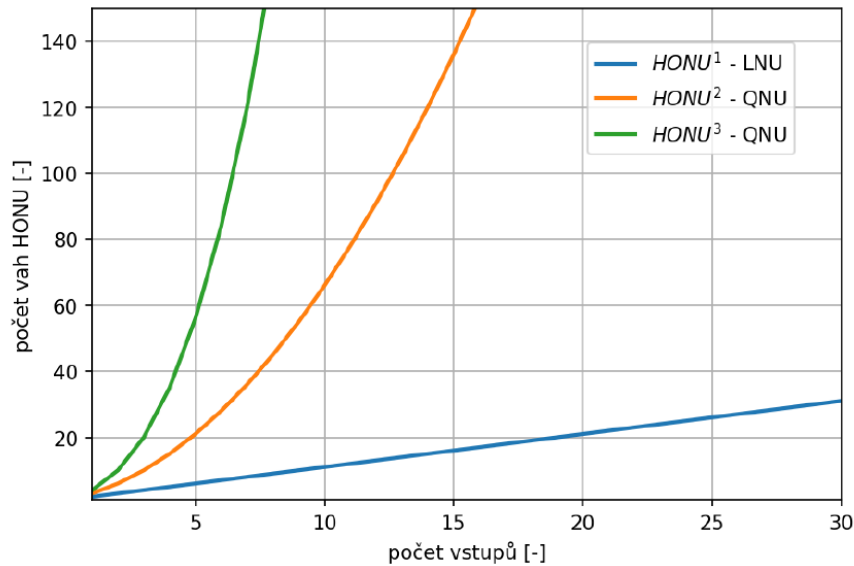
kde $x = [x_1, x_2, \dots, x_n]^T$ je vstupní vektor, $\sigma(\cdot)$ je aktivační funkce a y je výstup neuronu. V závislosti na počtu polynomů potom určujeme, zda je HONU 1., 2., 3., nebo vyššího stupně. 1. stupeň se pak nazývá LNU (*Linear Neural Unite*), 2. QNU (*Quadratic Neural Unite*) a 3. CNU (*Cubic Neural Unite*) a platí tedy následující:

$$LNU(x) = \sigma \left(\theta_0 + \sum_{i_1}^n w_{i_1} x_{i_1} \right), \quad (6)$$

$$QNU(x) = \sigma \left(\theta_0 + \sum_{i_1}^n \sum_{i_1 < i_2}^n w_{i_1, i_2} x_{i_1} x_{i_2} \right), \quad (7)$$

$$CNU(x) = \sigma \left(\theta_0 + \sum_{i_1}^n \sum_{i_1 < i_2}^n \sum_{i_2 < i_3}^n w_{i_1, i_2, i_3} x_{i_1} x_{i_2} x_{i_3} \right). \quad (8)$$

Stupeň polynomu pak dle této analogie může růst do libovolné velikosti. Problémem je však nárůst počtu vah, jak můžeme vidět na obr. 3, z čehož plyne i náročnost výpočtu.



Obr. 2.3: Graf růstu počtu vah v závislosti na stupni polynomu HONU [10]

2.2 Topologie, architektura a modely neuronových sítí

Neuronová síť je tedy tvořena jednotlivými neurony (výpočetními jednotkami), jež tvoří její topologii. První vrstva se nazývá vstupní (*input layer*), poslední výstupní (*output layer*) a vrstvy mezi těmito dvěma se označují jako skryté (*hidden layers*). Spojení mezi neurony v sousedních vrstvách se pak vypočte jako suma hodnot jednotlivých výstupů z n neuronů z vrstvy nižší, vstupujících do neuronu ve vrstvě vyšší, přičemž je vždy tato hodnota vstupních neuronů vyšší o prahovou hodnotu Θ (*bias*) [3]. Její velikost se vypočte podle vztahu:

$$\theta = w_0 x_0, \quad (9)$$

kde $x_0 = 1$ a $w_0 = -\theta$. Pak tedy vztah mezi jednotlivými vrstvami $l-1$ a l pro vstup do neuronu k -tého ve vrstvě l lze vyjádřit dle rovnice:

$$\xi_k^l = \sum_{i=1}^n w_i^l y_i^{l-1} - \theta_k^l. \quad (10)$$

Modelem neuronových sítí rozumíme typy neuronů, které pro sestavení dané sítě použijeme. Síť můžeme z hlediska jejich topologie, tedy jejich architektury, dělit na jednovrstvé, vícevrstvé, dle typu učení na učení s učitelem a bez učitele a na statické (*dopředné - feedward*), kdy se vstup neuronu šíří pouze na jeho výstup a po té do vrstvy následující a dynamické (*rekurentní*), kdy je přiveden vstup neuronu dané vrstvy po jeho zpracování zpět do neuronu ve vrstvě předcházející té aktuální [4]. Jelikož se u rekurentních sítí vyhodnocují nejen aktuální vstupy, ale i ty minulé a tento model obsahuje cykly, nazýváme je zpětnovazební.

Příklady modelů neuronů pro sestavení neuronových sítí mohou být následující:

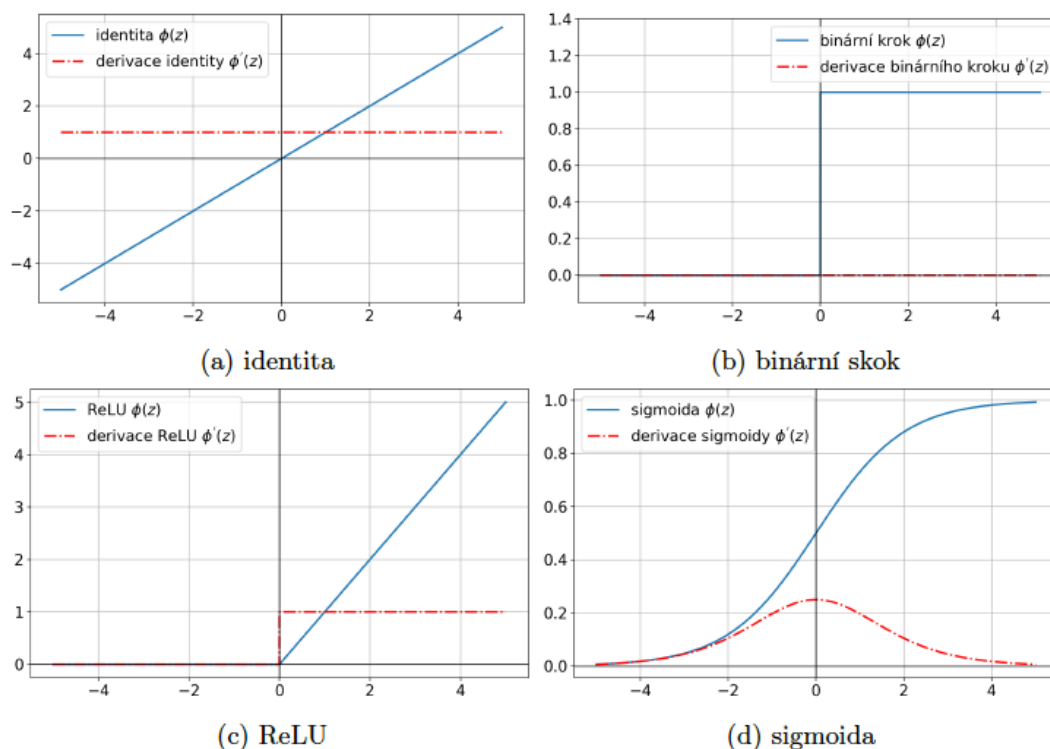
- Více vrstvý Perceptron (*Multi Layer Perceptron, MLP*)
- Neuronové sítě vyšších řádů (*Higher Order Neural Networks, HONN*)
- Extreme Learning Machine (*ELM*)
- Random Vector Functional Link (*RVFL*)
- Long Short Term Memory (*LSTM*), je speciální druh zpětnovazebních neuronových sítí
- Gated Recurrent Unit (*GRU*)
- Radial Basis Function (*RBF*), je síť s funkcí s radiální bází
- Hopfieldova síť je sítí vzájemně propojených neuronů, používaná jako model sémantické paměti
- Kohonenova síť, jedna z nejlepších sítí pro proces samoorganizace, jde o učení bez učitele

V kapitole 5 uvedu principy těch modelů neuronových sítí, které k predikci teplot v této práci nakonec použiji.

2.3 Aktivační funkce

Aktivační funkci neboli somatickou operaci můžeme chápat jako matematickou bránu mezi výstupem neuronu a vstupem do další vrstvy, nebo jako konečný výstup ze sítě. Každá jednotlivá neuronová jednotka může mít vlastní aktivační funkci, obvykle se však používají stejné pro celé vrstvy neuronových sítí [10].

Nejjednoduššími typy aktivační funkce jsou tzv. ostrá nelinearita (*unipolární binární*), kterou reprezentuje skoková funkce, nebo logistická sigmoida (*unipolární spojitá*). Aktivačními funkcemi dále mohou být hyperbolický tangens (*bipolární spojitá*), ReLU (*Rectified Linear Unit*) - rektifikovaná jednotka (*spojitá lineární*) a její modifikace, která je po částech lineární, nebo RBF (*Radial Basis Function*) - radiální báze a další. Průběh některých z nich můžeme vidět na obr. 4.



Obr. 2.4: Průběhy vybraných aktivačních funkcí a jejich derivace [10]

Jako aktivační funkce skoková ani lineární funkce však vhodné nejsou. Hlavním požadavkem na aktivační funkci je schopnost transformovat vstupní reálnou hodnotu do intervalu $\langle 0, 1 \rangle$, případně $\langle -1, 1 \rangle$ tak, aby v blízkosti nuly velmi prudce stoupala, zatímco u vysokých hodnot stoupala (klesala) jen nepatrně. Tyto požadavky splňuje logistická sigmoida nebo hyperbolický tangens. Tyto funkce si jsou velmi podobné, liší se především oborem hodnot, kdy logistická sigmoida nabývá hodnot v intervalu $\langle 0, 1 \rangle$, hyperbolický tangens $\langle -1, 1 \rangle$. Další výhodou obou zmíněných funkcí je snadné vyjádření jejich derivací [6].

2.4 Optimalizační algoritmy

Optimalizační algoritmy, označovány také jako algoritmy učení, nebo adaptační algoritmy neuronových sítí jsou předpisy, jakými se budou u těchto sítí upravovat a nastavovat, tedy adaptovat hodnoty vah w jednotlivých vstupů, prahové hodnoty neboli biasy Θ pro každou vrstvu, učící parametr (*learning rate*) a další parametry, které v důsledku stanoví, jakým způsobem se bude chybovost neuronové sítě minimalizovat. Nejčastější způsob minimalizace této chybovosti se realizuje prostřednictvím statistického nástroje výpočtu průměrné čtvercové chyby neboli střední kvadratické odchylky (*Mean Square Error, MSE*) a vypočte se dle vztahu:

$$MSE = \frac{1}{2} \sum_{i=1}^n (y_i - \hat{y}_i)^2. \quad (11)$$

Pakliže tento vztah přeformulujeme do podoby pro výpočet chybovosti neuronové sítě, píšeme:

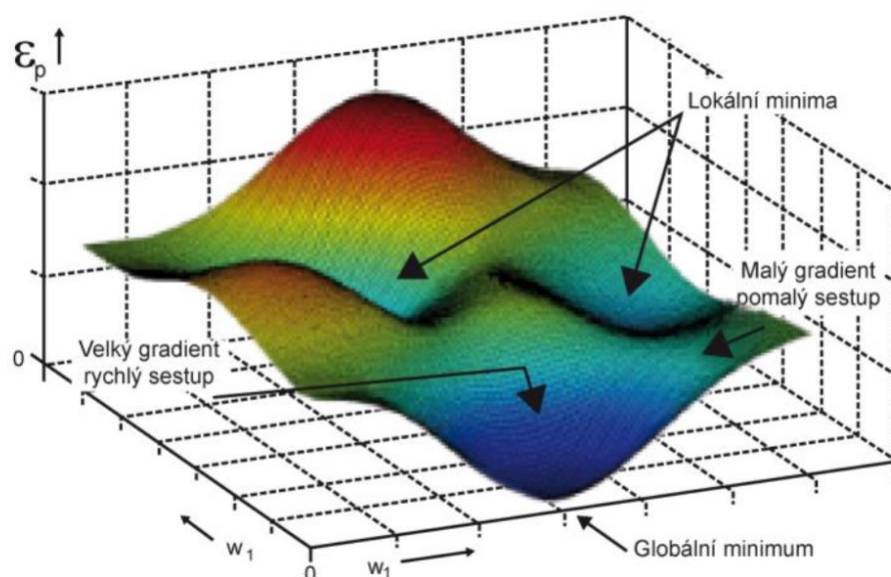
$$e^2 = (y_r - y_u)^2, \quad (12)$$

kde y_r je požadovaný výstup naší sítě a y_u je výstupem, který sít' skutečně spočítá. MSE pak můžeme uvažovat jako kvalitu odhadu.

Před vlastním použitím učících algoritmů je nutné analyzovat vlastnosti řešeného problému. K tomu nám do značné míry pomůže odpovědět si na následující otázky:

- Kolik je vstupních proměnných (jaká je dimenze úlohy)?
- Jsou všechny proměnné stejného typu a rozsahu?
- Jsou proměnné diskrétní nebo spojité?
- Jsou známy meze hodnot vstupních proměnných?
- Mohou být parametry optimalizovány nezávisle?
- Je kritériální funkce multimodální (má více minim)?
- Je optimální hodnota časově nezávislá (statická úloha)?
- Objevuje se v úloze šum (liší se výsledky pro stejné vstupní hodnoty)?[12]

Adaptační algoritmus tedy provádí iterativním procesem minimalizaci chybové funkce, kde hledá gradientní metodou minimum na chybové hyperploše. Na každou váhu sítě můžeme prohlížet jako na jednu dimenzi z n-dimenzionálního chybového prostoru. Na obr. 5 je znázorněna hyperplocha, kde na vodorovných osách jsou hodnoty vah a svislá osa reprezentuje chybovou funkci, která je funkcí prvků trénovací množiny. Cílem optimalizačního algoritmu je v ideálním případě nalézt globální minimum na této hyperploše, vždy ale platí riziko uváznutí v minimu lokálním. Tomuto problému se však dá předejít vhodnou volbou počátečních parametrů, trénovací množiny a modifikací tohoto algoritmu [8].



Obr. 2.5: Minimalizace chybové funkce při adaptaci [8]

Učení neuronových sítí rozdělujeme do dvou základních skupin. První z nich je učení bez učitele, pro které je typické, že adaptační algoritmus nemá k dispozici způsob, jak verifikovat správnost své činnosti. Tento případ je používán především u samoorganizujících se map. Druhou je učení s učitelem, kdy adaptační algoritmus má k dispozici množinu prvků, které představují vstupy a jim odpovídající výstupy, což je případ této práce, jelikož pro dané vstupy známe správné výstupy. Pro učení s učitelem rozlišujeme dvě metody, a to dávkové metody a metody vzorek po vzorku.

2.4.1 Dávkové metody

Dávkovými metodami označujeme metody, kdy v jedné iteraci učení (trénování) bereme v úvahu celou dávku. Často se kvůli hardwarovým omezením celá trénovací data rozdělí do několika dávek, které se střídají v procesu učení. Výhodou těchto metod je, že jsou účinnější (počítají se všemi daty najednou). Nevýhodou je výpočetní náročnost a že často neumožňují nasazení v online režimu.

2.4.1.1 Adam optimizer

Algoritmus Adam (*Adaptive Moment Estimation*) je velmi oblíbený optimizér. Vychází z algoritmů Adadelta a RMSprop. Často se používá pro optimalizaci rekurentních sítí. Kromě ukládání předchozích hodnot gradientu také používá moment vycházející z průměru předchozích gradientů. Existují i jeho modifikace, konkrétně algoritmus Adamax a Nadam [36].

2.4.1.2 Batch Gradient Descent (BGD)

Tato metoda je numerický iterativní algoritmus pro hledání funkčního minima.

Nechť Θ je vektor parametrů našeho systému, matice X je maticí vstupů a y je vektor výstupů. Dále předpokládejme, že funkce $h_0(x)$ je funkcí, jejíž parametry chceme optimalizovat:

$$h_0(x) = \sum_{j=0}^{n-1} \Theta_j x_j. \quad (13)$$

Naše chybové kritérium bude vypadat následovně:

$$J_{trénovací}(\Theta) = \frac{1}{M} \sum_{i=1}^M \frac{1}{2} (h_0(x^i) - y^i)^2. \quad (14)$$

Aktualizační pravidlo:

$$\Theta^{(k+1)} = \Theta^k - \alpha \frac{\partial}{\partial \Theta} J_{\text{trénovací}}(\Theta^k), \quad (15)$$

$$\Theta^{(k+1)} = \Theta^k - \alpha \frac{1}{M} \sum_{i=1}^M (h_0(x^i) - y^i) x^i, \quad (16)$$

kde k značí iteraci, M je počet řádků matice X a α je parametr, který ovlivňuje délku kroku [10].

2.4.1.3 Stochastic Gradient Descent (SGD)

Při implementaci algoritmu BGD se však potýkáme s několika problémy, nejzásadnějším je výpočetní náročnost chybového kritéria. Se zvyšováním složitosti našeho parametrického systému totiž stoupá i výpočetní náročnost. V každém kroku totiž musíme sečíst jednotlivé chybové funkce všech řádků matice X .

Stochastický Gradient Descent v daném kroku počítá pouze s jedním řádkem matice X , který náhodně vybere, což je fundamentalismem tohoto algoritmu.

Naše funkce, jejíž parametry budeme optimalizovat, bude ve stejném tvaru jako u algoritmu BGD v předcházejí kapitole, tedy:

$$h_0(x) = \sum_{j=0}^{n-1} \Theta_j x_j. \quad (17)$$

Dále zavedeme funkci *cost*:

$$\text{cost}(\Theta, (x^i, y^i)) = \frac{1}{2} [h_0(x^i) - y^i]^2 \quad (18)$$

a naše chybová funkce bude ve tvaru:

$$J_{\Theta} = \frac{1}{M} \sum_{i=1}^M \text{cost}(\Theta, (x^i, y^i)), \quad (19)$$

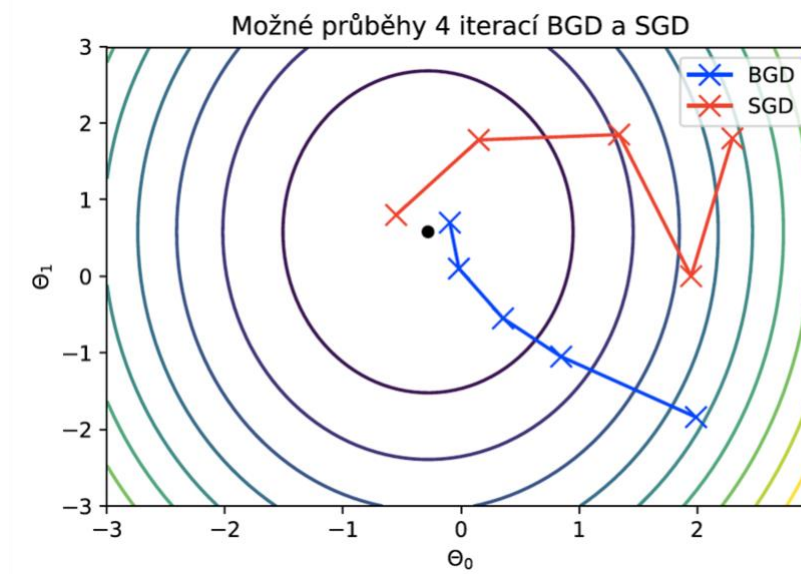
potom aktualizací pravidlo bude vypadat následovně:

$$\Theta^{(k+1)} = \Theta^k - \alpha \frac{\partial}{\partial \Theta} \text{cost}(\Theta, (x^k, y^k)), \quad (20)$$

$$\Theta^{(k+1)} = \Theta^k - \alpha (h_0(x^k) - y^k) x^k, \quad (21)$$

kde k je opět počet iterací a parametr α ovlivňuje délku kroku. Nejdůležitější součástí této metody je proces náhodného výběru před každou iterací.

Rozdíl obou průběhů můžeme vidět na obrázku 6, kde na osách x a y jsou parametry Θ_0 a Θ_1 . Barevné vrstevnice pak značí chybovou funkci J_Θ a černý bod je lokálním minimem [10].



Obr. 2.6: Porovnání metod SGD a BGD [10]

2.4.1.4 Levenberg-Marquardtova metoda (LMA)

LMA je numerický iterativní algoritmus pro hledání minima funkce, která kombinuje Gauss-Newtonovu metodu a metodu Gradient Descent. Tato metoda obsahuje klíčové vlastnosti, jako je robustnost, efektivnost, konvergenci a numerickou stabilitu.

Aktualizační pravidlo při použití tohoto algoritmu pro neurony HONU bude vypadat následovně:

$$\Theta^{l+1} = \Theta^l + [X^T X + \lambda I]^{-1} X e, \quad (22)$$

kde λ je tlumící faktor (*dumping factor*) a určuje poměr metod, které LMA kombinuje, X je matice vstupů, I je jednotková matice a e je vektor chyb našeho modelu podle:

$$e = X\Theta - y, \quad (23)$$

kde y je vektor výstupů [4], [10].

2.4.2 Krokové metody

Neboli metoda typu vzorek po vzorku jsou taktéž numerickými iterativními algoritmy. Oproti metodám dávkovým ale v dané iteraci pracují jen s jedním vzorkem (párem typu vstup-výstup). Jinými slovy je proces mnohem rychlejší a je vhodný k učení za běhu, častěji se však objevují problémy s konvergencí.

2.4.2.1 Gradient Descent

V minulé části učících metod byly popsány metody *Batch Gradient Descent* a *Stochastic Gradient Descent* jakožto dávkové. K dispozici je však i kroková metoda tohoto iterativního postupu nazvaná jednoduše *Gradient Descent*. U této metody učení vzorek po vzorku jde prakticky o užití BGD, nebo SGD, ale jen s jedním učícím vzorkem. Aktualizační pravidlo pak v tomto případě píšeme ve tvaru:

$$\Theta^{l+1} = \Theta^l - \alpha(h_0(x) - y)x, \quad (24)$$

kde x, y jsou učícím párem (vstup, výstup).

2.5 Algoritmus zpětného šíření chyby

Algoritmus zpětného šíření chyby (*Back Propagation, BP*) je základním algoritmem učení vícevrstvých neuronových sítí. Jak bylo uvedeno již v úvodu této práce, prvotní

Perceptron nedokázal ze své podstaty řešit problém s lineární separovatelností množin, tedy logickou operaci (*exkluzivní disjunkce, XOR*). Zlom nastal v roce 1974, když Paul J. Werbos představil svůj algoritmus zpětného šíření chyby.

Základem algoritmu je přepočítání jednotlivých vah w neuronové sítě. Individuálně se jejich velikosti korigují tak, aby zvolená chybová funkce J postupně snižovala svou hodnotu. Přepočítání synaptických vah se provádí pouze v případě, že je odchylka dosažené odezvy od skutečné větší než požadovaná tolerance učení. Při adaptaci se informace o chybě šíří směrem od výstupní vrstvy ke vstupní [9].

Postup algoritmu BP:

1. Inicializace: nastaví všechny váhy v síti na náhodné hodnoty
2. Předloží vzor: z trénovací množiny vybere vzor a vloží jej na vstupy sítě a dále se po vrstvách směrem od vstupní k výstupní spočítají výstupy jednotlivých neuronů dle vztahu:

$$y = \sigma \sum_{i=1}^n x_i w_i \quad (25)$$

kde $\sigma(\xi)$ je aktivační funkce.

3. Srovnání: vypočte se chyba sítě dle vztahu:

$$J = \frac{1}{2} \sum_{i=1}^n u_i - d_i \quad (26)$$

kde n je počet neuronů výstupní vrstvy, u_i je skutečná odezva *i-tého* neuronu ve výstupní vrstvě na vstupní vektor a d_i je očekávaný výstup *i-tého* neuronu ve výstupní vrstvě. Tato chyba se použije jako přírůstek celkové energie spočítané přes všechny vzory. Dále se vypočítá chyba pro vstupní vrstvu jako:

$$\delta_i^u = (d_i - u_i^u)(1 - u_i^u)u_i^u \quad (27)$$

4. Zpětné šíření chyby: pro všechny neurony ve vrstvě l se vypočtou váhy dle vztahů:

$$\Delta w_{i,j}^l = \eta \delta_i^l(t) u_j^{l-1}(t) + \mu \Delta w_{i,j}^l(t-1), \quad (28)$$

$$\Delta \theta_i^l = \eta \delta_i^l(t) + \mu \Delta \theta_i^l(t-1), \quad (29)$$

kde η je koeficient učení a μ je koeficient vlivu změny vah z předchozího kroku. Oba koeficienty nabývají hodnot z intervalu $\langle 0, 1 \rangle$. Pokud je nastavení $\mu = 0$, pak předešlé nastavení váhy nemá vliv na nové nastavení a hodnota změny synaptické váhy je dána pouze aktuálním krokem. Čím vyšší koeficient učení bude nastaven, tím se projeví větší v neuronové síti a naopak. Pak se podle vztahu:

$$\delta_i^{h-1} = y(1 - u_i^{h-1}) \sum_{k=1}^n w_{k,i}^h \delta_k^h \quad (30)$$

chyba šíří zpětně do vrstvy, která je nejbližší vstupům. Na konec tohoto bodu se modifikují váhy:

$$w_{i,j}^l(t+1) = w_{i,j}^l(t) + \Delta w_{i,j}^l(t) \quad (31)$$

$$\theta_i^l(t+1) = \theta_i^l(t) + \Delta \theta_i^l(t) \quad (32)$$

Krok 4 se opakuje pro všechny vrstvy sítě. Začíná se vrstvou výstupní a pak následují skryté vrstvy. V případě, že se zpracovává skrytá vrstva, která je nejbližší výstupní vrstvě, nahradí se u_i^{l-1} ve vztahu (10) odpovídající vstupní hodnotou, tedy x_j . Pak platí:

$$\Delta w_{i,j}^l = \eta \delta_i^l(t) x_j(t) + \mu \Delta w_{i,j}^l(t-1) \quad (33)$$

5. Konec výběru vzorků z trénovací množiny: jestliže jsme předložili síti všechny vzorky z trénovací množiny, pak se pokračuje krokem 6, jinak se algoritmus navrácí zpět ke kroku 2.
6. Konec učení: jestliže byla chyba sítě po posledním výpočtu menší než požadované kritérium, proces se ukončí, jinak pokračuje krokem 2 [6].

3 Rešerše termodynamických dějů v letadlových motorech a jejich predikce

V našem případě byla měřena teplota plynu proudícího motorem až po jeho záběhu, teploty již tedy byly v ustáleném stavu a jejich změna se dále měnila jen v závislosti na termodynamických a mechanických dějích uvnitř motoru, případně na ohřevu jeho okolí a opotřebení měřicích senzorů. Sensory pro odečítání teploty byly typu plášťových hřebenových termočlánků. Více se o nich zmíním v kapitole 3.1.3.

3.1 Měření teploty v technické praxi

3.1.1 Teplo a teplota

Teplota je termodynamická stavová veličina, která je klíčová pro popis všech tepelných dějů. Z makroskopického hlediska je definice teploty velmi komplexní, z mikroskopického hlediska je teplota úměrná kinetické energii částic látky [13].

3.1.2 Chyby a nejistoty měření

Při měření fyzikálních veličin může dojít k celé řadě chyb, které je nutné eliminovat nebo alespoň minimalizovat. Tyto chyby rozlišujeme podle původu a podle povahy. Podle své povahy mohou být měřené fyzikální veličiny:

- statické: v čase neproměnné, konstantní,
- kvazistatické: čase proměnné tak pomalu, že je během měření chápeme jako statické,
- dynamické: v čase proměnné a to periodicky nebo neperiodicky,

podle místa původu pak rozlišujeme chyby metody, chyby přístroje a chyby člověka. Příčinou vzniku chyb metody je nevhodný výběr metody vzhledem k účelu, vlivu prostředí, nebo nevhodného umístění snímačů, chyby přístroje lze zahrnout do vyhodnocení měření,

kdy podle principu a konstrukce přístroje může být chyba statická, nebo dynamická. Nejzásadnějším původem chyby pak bývá lidský faktor, kde vznikají chyby nesprávnou obsluhou měřicího zařízení.

Podle charakteru a velikosti rozlišujeme:

- chyby hrubé
- chyby systematické
- chyby náhodné [13]

Přesnosti měření prostřednictvím relativních chyb senzorů a absolutních chyb měření bývají reprezentovány nejistotami měření. Nejistota měření je parametrem přiřazeným k výsledku měření, charakterizující rozptyl hodnot, které lze racionálně a s určitou pravděpodobností přiřazovat k měřené veličině.

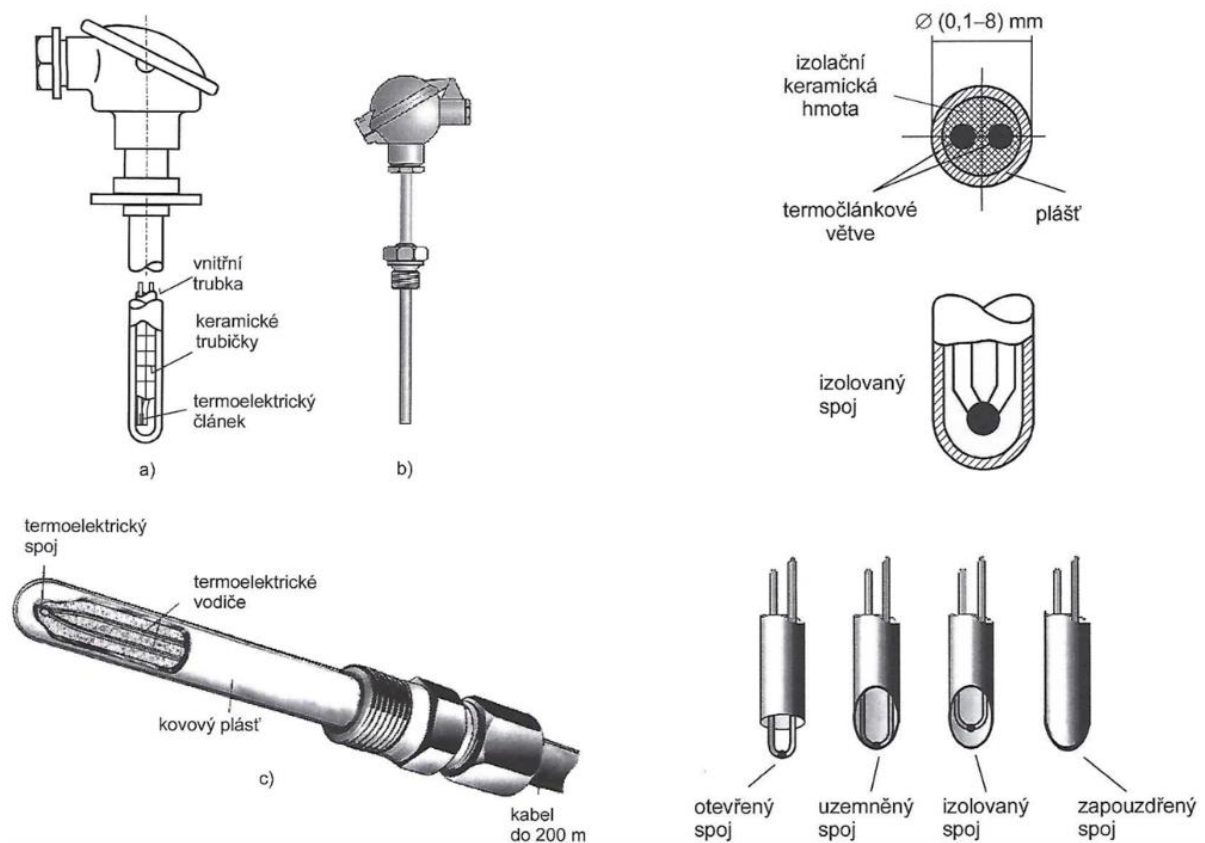
Dle způsobu vyhodnocení se nejistoty měření člení do těchto skupin:

- standardní nejistota typu A, značena u_A
- standardní nejistota typu B, značena u_B
- kombinovaná standardní nejistota, značena u_C
- rozšířená nejistota, značena $U_{(x)}$ [14]

3.1.3 Plášťové termoelektrické články

Teploměry lze členit do dvou základních kategorií, na dotykové a bezdotykové. V řešeném případě se pak jedná o použití termočlánků dotykových, konkrétně o plášťové termoelektrické články.

Tyto snímače teploty mají termoelektrické vodiče uloženy v kovovém plášti ve tvaru trubky a jsou uloženy v plášti izolovány zhutnělou práškovou minerální izolací z MgO nebo Al₂O₃, jak je vidět na obr. 8. Plášť je pak z korozivzdorných, nebo žáruvzdorných ocelí, z chromniklových slitin (Inconel, Incoloy, Hastelloy, Nicrobell), nebo ze slitiny PtRh10 pro vysoké teploty [14].



Obr. 3.1: Nalevo: termoelektrický snímač teploty (a - tyčový, b - do jámky, c – plášťový kabel), napravo: plášťové termoelektrické články [14]

3.1.4 Seebeckův jev

Termoelektrické články jsou založeny na jevu, jehož podstatou je převod tepelné energie na elektrickou. Tento jev v roce 1821 popsal německý fyzik T. J. Seebeck. Termočlánek vzniká spojením různých kovových vodičů, kdy jejich spojením na obou koncích vznikne termoelektrický obvod, kterým protéká stejnosměrný proud. Elektrické napětí v obvodu je pak funkcí rozdílu teplot spojených konců a materiálu vodičů. Platí, že pro velmi malé změny teplot je napětí lineárně závislé na teplotě podle vztahu:

$$\Delta u_{AB} = \alpha \Delta T, \quad (34)$$

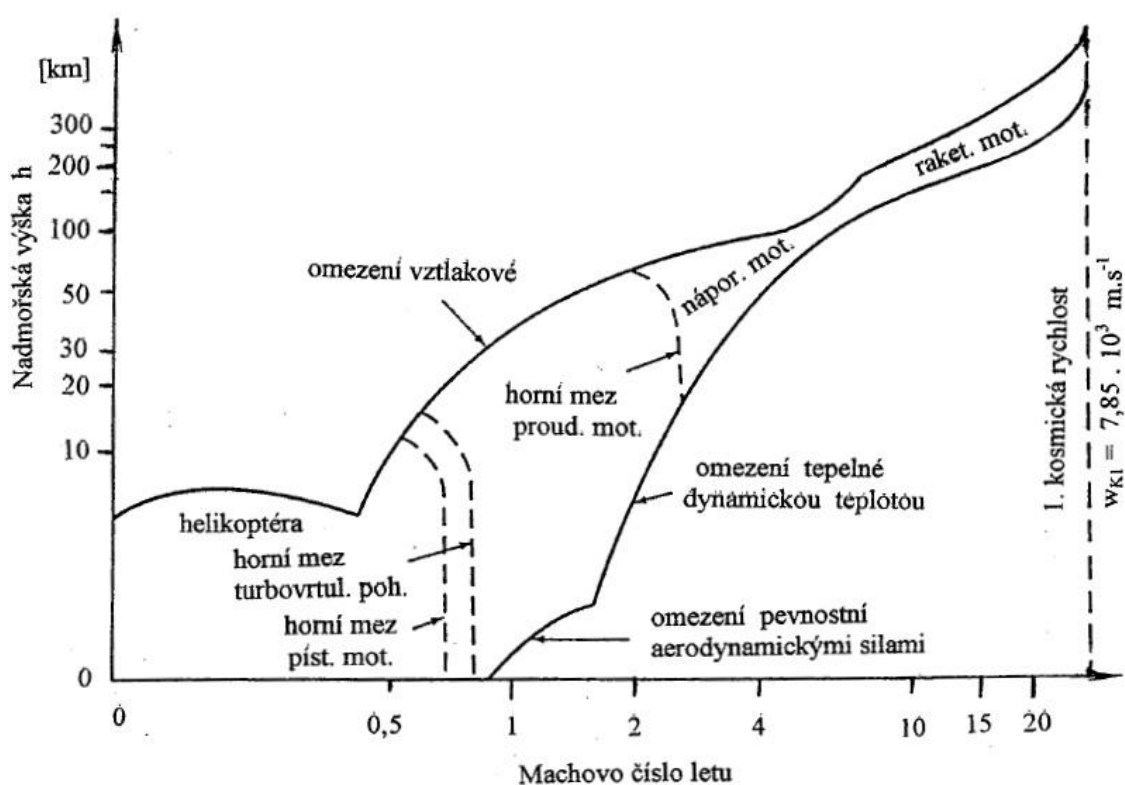
kde Δu_{AB} je Seebeckovo napětí a α je Seebeckův koeficient [14].

3.2 Letadlové motory

Při všech režimech letu, s výjimkou letu klouzavého, musí na letadlo působit tah F_T . Tohoto tahu je docíleno transformací chemické energie na tepelnou a následně transformací tepelné energie na energii mechanickou prostřednictvím tepelného oběhu. Pro tuto transformaci platí zákon o zachování hmotnosti a zákon o zachování energie (I. věta termodynamická), stejně tak jako i II. věta termodynamická. Tato transformace probíhá v soustavě pohonu letadla, kterou označujeme jako propulzní soustavu letadla, propulzní látkou je pak pracovní látka, jejíž součástí je i nasávaný vzduch [18].

Pohony letadel členíme do 3 základních skupin, a to:

- vrtulový
- proudový
- raketový



Obr. 3.2: Aerodynamická letová obálka – limity aerodynamického letu [17]

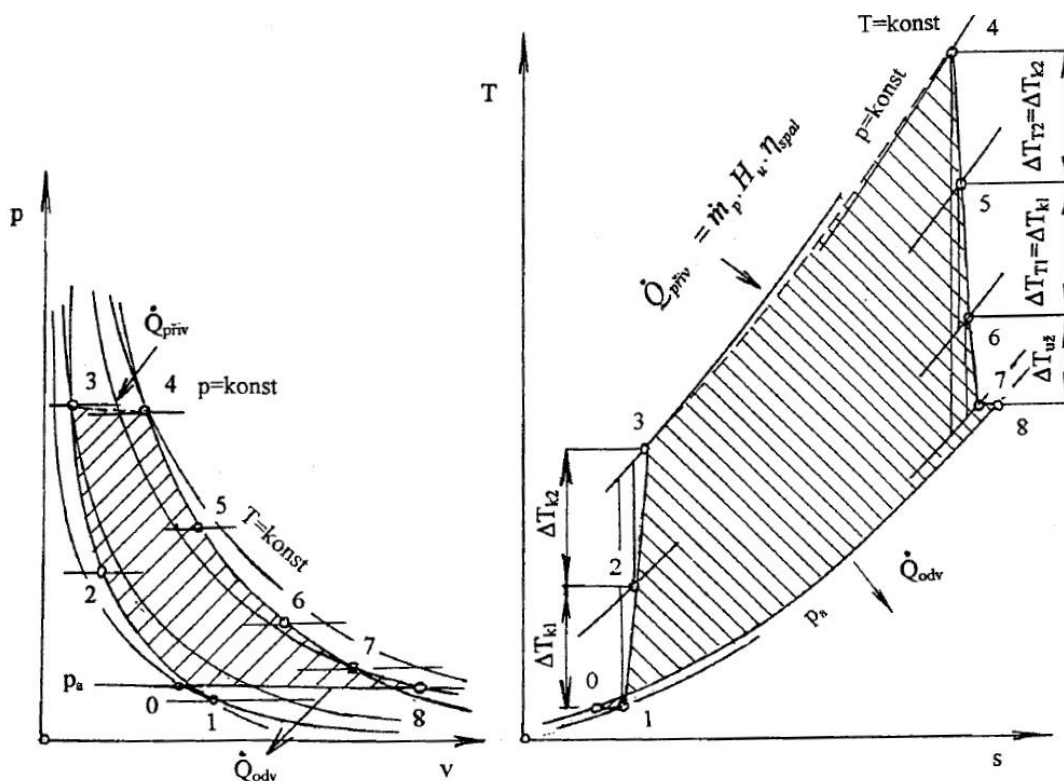
3.2.1 Teorie tepelných motorů

Stlačitelnost pracovní látky je klíčovou vlastností pro transformaci tepelné energie na energii mechanickou a tím na mechanickou práci, která vzniká působením tlaku pracovní látky na pohyblivé pracovní části motoru. Tyto části jsou spojené s výstupním hřídelem, z něhož je mechanická práce odváděna. Kinetická energie plynů je další formou mechanické práce. Tato energie je transformována z tlakové a tepelné energie plynu.

Aby bylo možné docílit potřebného tlaku plynu, je třeba zajistit tepelný oběh. Ten je tvořen z uzavřené posloupnosti změn tepelného stavu pracovního plynu, která obsahuje především stlačení plynu, přívod tepla do stlačeného plynu a následně jeho expanzi. Ta působí na pracovní prvky stroje a předá jim tak mechanickou energii. Tato posloupnost změn stavu plynu se nazývá tepelným oběhem (cyklem) a musí probíhat v uzavřené smyčce. Je tedy nezbytné docílit kontinuálního cyklu komprese, přívodu tepla a expanze, s následnou transformací na mechanickou energii.

Projevem druhého termodynamického zákona však po expanzi horkého plynu jeho teplota neklesne na hodnotu před kompresí. Aby přešel tepelný stav plynu po jeho expanzi opět na svou původní hodnotu a tepelný oběh se tak uzavřel, je nutno z plynu odvézt jisté množství tepla při konstantním tlaku do okolí, což je realizováno výfukem spalin do volné atmosféry a nasátím čerstvého vzduchu. Tepelný oběh se označuje jako otevřený, jelikož dochází k výměně propulzní látky motoru [16].

Jak je na obr. 9 vidět, tepelný oběh spalovací turbíny lze vyjádřit pomocí $p - v$ a $T - s$ diagramu:



Obr. 3.3: Oběhy v p - v a T - s diagramu [17]

Při průtoku pracovní látky spalovací turbínou se mění plynule termodynamický stav od vstupu (bod 0) do výstupu (bod 8), přičemž ve vstupu a výstupu z turbínového motoru je tlak roven tlaku atmosférickému p_a . Tím se oběh pracovní látky uzavírá. Z diagramu $p - v$ je patrné, že při oběhu uvedeného cyklu v posloupnosti bodů 0, 1, ..., 8, 0 je v důsledku proběhlých stavových změn produkována pracovní látkou užitečná mechanická práce [17].

3.2.2 Turbovrtulový motor

Turbovrtulový motor sestává z hlavních komponent, jako jsou kompresor, spalovací komora, plynová turbína a výkonová turbína.

Vrtule je lopátkový stroj sestávající z rotoru o dvou nebo více vrtulových listech uložených ve vrtulové hlavě (náboji) po obvodu rovnoměrně rozložených a nastavených tak, aby při otáčení vrtule byly aerodynamické profily listů účinně obtékány vzduchem. Přivedený výkon vrtulí je definován:

$$N_{přiv} = M_V \alpha, \quad (35)$$

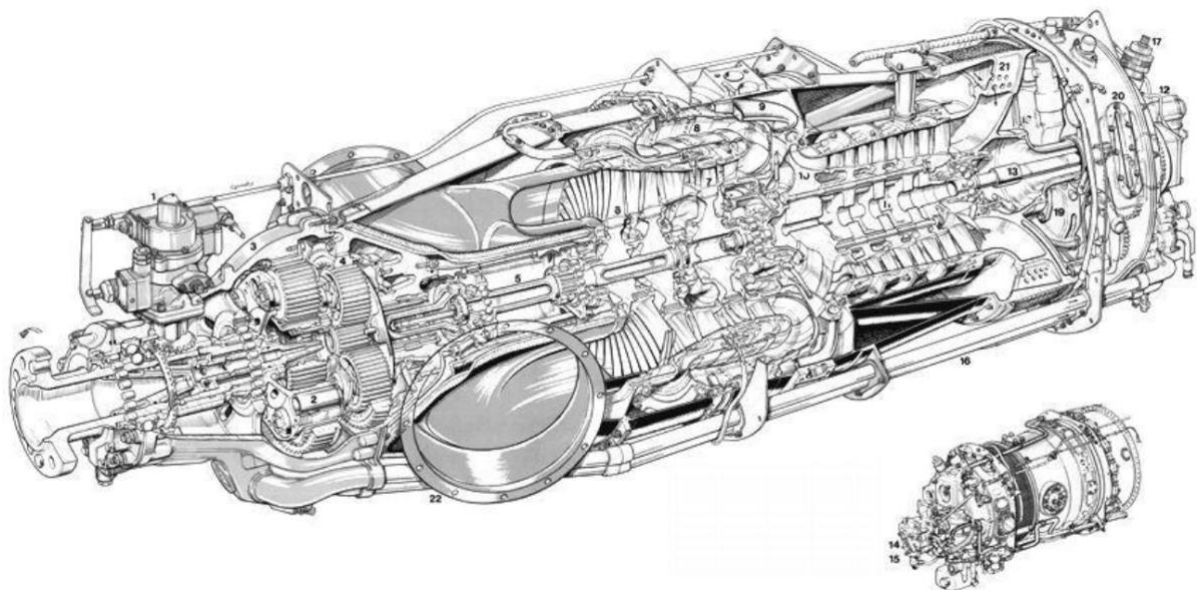
kde M_V je odporový točivý moment vrtule a α je úhlová rychlost otáčení vrtule, propulsní výkon vrtule je:

$$N_T = F_{Tv} w_L, \quad (36)$$

kde F_{Tv} je celková tahová síla vrtule a w_L je rychlost přitékajícího vzduchu k listu vrtule a účinnost je pak dána vztahem:

$$\eta_V = \frac{N_T}{N_{přiv}} = \frac{F_{Tv} w_L}{M_V \alpha}. \quad (37)$$

Turbovrtulový motor je turbínový motor, u něhož je užitečný mechanický výkon vyveden z motoru na hřídel, kterým je přes reduktor poháněna vrtule. Tah motoru je vyvozován vrtulí, přičemž spaliny vystupující z motoru relativně nízkou rychlostí z výstupní trysky motoru působí přidavným tahem. Na rozdíl od proudového motoru, kde jsou hnací plyny vystupující z turbíny dále v hnací trysce urychlovány, jsou vystupující spaliny z turbíny turbovrtulového motoru zpomalovány ve výstupním difuzoru na rychlost mírně vyšší, než je cestovní rychlost letu [16].



Obr. 3.4: Řez 3D modelem turbovrtulového motoru [25]

3.3 Rešerše využití a aplikací pro predikci neuronovými sítěmi

Problematika vyhodnocování chování a stavů motorů (v našem případě turbínových strojů) a jejich částí je velmi rozšířené téma na mnoha vědeckých a výzkumných pracovištích, či institucích. Prostřednictvím neuronových sítí se často řeší problematika teploty výfukových plynů motorů (*Exhaust Gas Temperature, EGT*) nebo analýza dráhy plynu (*Gas Path Analysis, GPA*) v letadlových motorech za účelem jejich diagnostiky. Podle [27] byla taková první studie, jež je popsána v [28], publikována roku 1997.

Zde uvedu problematiku oblasti, které byly v této oblasti řešeny a jsou si svou podstatou řešení k predikci teplot plynu v turbovrtulových motorech nejbliže.

3.3.1 Řešené problematiky v oblasti lopatkových strojů prostřednictvím ANN

Práce [24] se zabývá vývojem ANN pro studium průtoku vzduchu propulzním systémem letadla a jeho vlivu na parametry tohoto systému. Dále je v této práci také docíleno odvození algoritmu, který je nástrojem pro volbu parametrů ANN sloužících pro simulaci chování průtokového systému motoru pracujícího jak se závadami, tak i bez nich. Díky tomu je možné provést diagnostiku a stanovit tak vhodné řešení závad tohoto propulzního systému.

Práce se nejdříve zabývá dvouprúdovým turbodmychadlovým motorem (*Turbofan*). Poznatky pak byly aplikovány pro studium průtokové soustavy turbovrtulového motoru 600 SHP, jak bude dále popsáno. Tato studie po celou dobu uvažuje neustálou nutnost nejhodnější volby modifikace vyvíjených neuronových sítí pro toto modelování a simulace toku propulzní látky, včetně jejich změn ve vztahu k charakteristikám tohoto motoru. Obsahem práce jsou také volby architektur, přenosových funkcí, parametrů trénování a ověřování přesnosti i robustnosti neuronových sítí.

Vstupem této ANN jsou hodnoty objemu průtoku propulzní látky, nadmořské výšky a Machovo číslo letu. Výstupními informacemi byly hodnoty celkového hrubého specifického tahu, tahu ventilátoru, tahu spalín, poměru tlaku motoru (EPR), poměru průtoku paliva k hmotnostnímu průtoku na vstupu motoru, měrné spotřeby paliva (SFC), poměru výstupních spalín k hmotnostnímu průtoku na vstupu, poměru vzduchu z výstupní

oblasti k hmotnostnímu průtoku na vstupu a teploty výfukových plynů (EGT). Nezbytné údaje pro naučení ANN poskytuje program pro simulaci pracovního cyklu motoru. Výpočet byl proveden náhodným změněním hodnot průtoku vzduchu, nadmořské výšky a Machova čísla. Výpočty cyklu motoru poskytují 10 000 vzorků. Pro trénování sítě bylo použito 7 000 vzorků, zatímco zbývajících 3 000 bylo použito pro testování a její validaci.

První pokus sestával z klasické třívrstvé sítě. Vstupní vrstva měla 11 neuronů + bias, který byl stejně jako všechny neurony přiřazen ke každému neuronu skryté vrstvy. Šlo tedy o plně propojenou síť (*fully connected layer*). Výstupní vrstva měla 9 neuronů, jeden pro každý výkonnostní parametr motoru. Studie testovala různé modely s různými neurony skryté vrstvy.

Další konfigurace 22, 44, 88, 200 a bias byla testována dávkovým učením a pro skrytou i výstupní vrstvu byla použita aktivační funkce typu sigmoida. Důležitým zjištěním této sítě bylo, že nebyla schopna dokončit své učení, pokud nebyla přípustná značná velikost chybové funkce.

Další modifikace sítě byla o zcela odlišné konfiguraci. Tato síť měla 11 neuronů vstupní vrstvy, přičemž každý z nich měl přiřazenou odlišnou prahovou hodnotu. Skrytých vrstev bylo 11, kdy každá z nich byla tvořena dvěma neurony a biasem. Neurony mezi vrstvami byly plně propojené, biasy dané skryté vrstvy však byly vždy spojeny jen s jedním výstupem. Přenosová funkce neuronů ve skrytých vrstvách byla sigmoidální, zatímco přenosová funkce výstupní vrstvy byla lineární.

Následující část této práce se zabývala propulzním systémem turbovrtulového motoru, kde byl opět použit počítačový model pro generování parametrů k trénování neuronové sítě. Tato fáze studia vedla k vytvoření 6 000 vzorků. V tomto případě měla ANN jednu vstupní vrstvu se 13 neurony (hodnoty tlaku v každé komoře + bias), jednu skrytou vrstvu s 25 neurony + bias kdy se tato volba ukázala jako nejlepší kompromis mezi přesností sítě a dobou jejího trénování, a nakonec výstupní vrstvu se 7 neurony. Pro učení ANN bylo použito 2000 vygenerovaných vzorků. Parametr učení byl nastaven na 0,7 a po 5 000 epochách učení chybová funkce dosáhla hodnoty $1,7 \cdot 10^{-7}$. Pro trénink ANN byla použita metoda konjugovaného gradientu.

Po naučení a testování byla vyvinutá ANN použita jako základní část kódu pro diagnostiku průtokového systému.

Výzkum v [33] se zabývá predikcí výkonových parametrů kompresoru a turbíny pomocí trénování modelu Neuro-Fuzzy (ANFIS) na základě map komponent. Teplota,

rychlost otáčení, hmotnostní průtok, tlak a další parametry jsou vytvářeny pomocí termodynamických vzorců a zákonů zachování energie. S ohledem na tyto vypočtené hodnoty jsou provedeny výpočty chyb, které jsou následně porovnány s údaji o cyklu motoru ze souvisejících simulačních podmínek. Výsledky simulace ukazují, že navržené parametry modelu motoru mají přijatelnou přesnost jak pro konstrukční, tak mimo konstrukční podmínky, od doby nečinnosti až po maximální provozní výkon, a to s ohledem na hodnoty tohoto cyklu. Navržený model motoru lze přizpůsobit i jiným typům motorů s plynovou turbínou.

V této práci je tedy modelován turbovrtulový motor, kdy pro predikci výkonu kompresoru a turbíny je použita síť ANFIS.

V [34] je představen nový přístup pro získání inverzního mapování termodynamicky optimalizovaných ideálních proudových motorů pomocí Group Method of Data Handling (GMDH) a evolučních algoritmů (EA). EA se používají ve dvou různých aspektech.

Za prvé, pro optimalizaci termodynamického cyklu ideálních turbovrtulových motorů s ohledem na čtyři důležité protichůdné termodynamické cíle, a to na specifický tah, pro který se používají více cílové EA s mechanismem zachování nové diverzity, dále měrnou spotřebu paliva, účinnost pohonu a tepelnou účinnost. Nejlepším získaným Paretoovým optimem byla pak datová tabulka reprezentována datovými dvojicemi nedominovaných vektorů návrhových proměnných, kterými jsou Machovo číslo a velikost tlaku a odpovídající čtyři objektivní funkce.

Za druhé, EA a dekompozice singulární hodnoty, které jsou současně použity pro optimální návrh konfigurace konektivity, respektive hodnot koeficientů neuronových sítí typu GMDH, které se používají pro inverzní modelování vstupně-výstupních hodnot získaných jako nejlepší Paretoovo optimum. Existují tedy dva různé polynomické vztahy mezi čtyřmi termomechanickými ději. Machovo číslo i poměr tlaku jsou stanoveny opět pomocí Paretova optima. Výsledky získané v tomto článku jsou velmi slibné a ukazují, že takové důležité vztahy mohou existovat a mohly by být objeveny pomocí více cílových EA a evolučně navržených neuronových sítí typu GMDH.

Ve studii [31] se autoři zabývali odhadem teploty výfukových plynů (EGT) turbofanového motoru CFM56-7B pomocí ANN. A to při dvou různých nastaveních výkonu, konkrétně při maximálním nepřetržitém a při vzletu. Studie byla provedena

s využitím informací o provozních parametrech motoru, jako je čistý tah, průtok paliva, otáčky motoru, teplota přiváděného vzduchu a specifická spotřeba paliva. Všechna tato data byla převzata z měření zkušebních laboratoří během provozu motorů na zemi. Výsledky predikce modelů ukazuje, že ANN je schopna přesně predikovat EGT v použitých letadlových motorech.

Pro řešení tohoto problému byla použita síť MLP a optimalizace probíhala pomocí Levenberg-Marquardtova algoritmu. Síť se skládala z osmi vstupů a jednoho výstupu. Vstupy sítě byly čistý tah (FN), průtok paliva (WF), nízké otáčky (N1), vysoké otáčky (N2), tlakový poměr (PR), teplota vzduchu na vstupu motoru (TA), vzletová teplota (TM) a specifická spotřeba paliva (TSFC). Výstupem sítě MLP pak byla EGT.

Po několika pokusech bylo zjištěno, že nejvhodnější konfigurací sítě byly tři skryté vrstvy s deseti neurony pro první skrytou vrstvu, tři neurony pro druhou skrytou vrstvu a jeden neuron pro třetí skrytou vrstvu. Vstupní a výstupní vrstva MLP mají lineární přenosovou funkci a skryté vrstvy mají funkce tangenciální a sigmoidální. Učící parametr sítě byl nastaven na 0,001. EGT bylo měřeno pomocí termočlánků T49.5 se senzorem typu K.

V [32] je použito síť Kernel Extreme Learning Machine (KELM) pro diagnostiku poruch leteckých motorů. Protože je však její výpočetní složitost úměrná velikosti dat pro trénování, je její použití v reálném čase omezené. Pro řešení tohoto nedostatku je v tomto článku navržen nový algoritmus DKELMs. Tento algoritmus byl vyvinut za účelem snížení výpočetní složitosti. Následně byla pro návrh fúzního schématu použita teorie pravděpodobnosti a teorie důkazů pro prokázání, že přesnost sítě nebude po kompresi dat snížena. Významným zjištěním bylo, že tento algoritmus může výrazně snížit výpočetní složitost a zlepšit výkon původního algoritmu KELM v reálném čase, aniž by došlo ke snížení přesnosti modelu.

V práci [20] se zabývají letadlovým motorem, kde řeší sledování jeho výkonu. Jelikož byla vyžadována přesná funkce všech sensorů, přičemž kvůli náročným provozním podmínkám byla ovlivněna jejich životnost a přesnost, byl přijat nový přístup pro validaci a odhad sensorů pomocí neuronových sítí.

Výzkum popsán v [22] řeší moderní metody monitorování stavu letadlových motorů za cílem snížení nákladů na údržbu, zvýšení bezpečnosti letadel a snížení spotřeby paliva.

Hodnoty parametrů, jakožto otáčky motoru, vibrace, tlak oleje, teplota oleje, teplota výfukových plynů (EGT) a průtok paliva jsou použity ke stanovení změny výkonu motorů. Během této studie byl vyvinut nový model pro získání informací o stavu leteckého motoru. Tento model, založený na vícenásobné regresní analýze, byl vytvořen za účelem stanovení závislostí letových parametrů na teplotě výfukových plynů a její predikci.

Práce popsaná v [23] se zabývá genetickými algoritmy pro návrh optimální počáteční hodnoty parametrů a topologie umělé neuronové sítě, která je optimalizována vylepšeným algoritmem zpětné propagace pomocí faktoru hybnosti tak, aby se maximalizovala její efektivita. V této studii je provedeno komplexní dynamické modelování součástí turbovrtulového motoru pomocí genetického algoritmu umělé neuronové sítě. Nově odvozený model GA-ANN zohledňuje pět nezávislých proměnných motoru, a to točivý moment, výkon, otáčky generátoru plynu, hmotnostní průtok vzduchu v motoru a průtok paliva. Tyto proměnné jsou použity jako vstupy ANN, zatímco exergie komponent je brána jako výstupní parametr sítě. Výsledky ukazují, že kombinace s genetickým algoritmem významným způsobem zlepšila přesnost sítě

V případě [27] se autoři věnují analýze proudění propulzní látky za účelem diagnostiky plynových turbín letadlových motorů, například tedy ke sledování zhoršení jejich výkonu. Ve srovnání s jinými metodami založenými na údajích dosahuje tato metoda vyšší diagnostické přesnosti. Díky tomu je tedy možné skrz optimalizaci plánu údržby snížit její náklady. Práce [29] se věnuje obdobnému tématu, stejně tak jako i [30].

Ve studii [21] je podrobně popsána ANN pro predikci charakteristik diesellového motoru, konkrétně k predikci točivého momentu, výkonu, spotřeby paliva a emisí, včetně oxidu uhličitého, oxidu uhelnatého, oxidů dusíku nebo například celkových uhlovodíků.

Dle [19] jsou neuronové sítě využívány také v následujících oblastech: Identifikace poruch a hodnocení průmyslové elektrárny, poruchy senzorů a akčních členů v automobilových motorech, diagnostika poruch v jaderných elektrárnách, detekce mechanického poškození způsobeného vibracemi ovlivňujících převodovku a pouzdro ložiska. Cílem této práce bylo vyvinout metodiku diagnostiky poruch plynové turbíny pomocí ANN.

3.3.2 Shrnutí poznatků z výzkumů a studií v oblasti lopatkových motorů s využitím ANN

Zjištěním této rešeršní části je, že neuronové sítě jsou v oblasti letadlových motorů nejčastěji používány za účelem jejich prediktivní údržby a diagnostiky. Dále je také častou aplikací neuronových sítí řešena predikce teploty a složení látek výfukových spalin a plynů. Mnoho vědeckých činností je také zaměřeno na predikci degradace materiálů nejen částí motorů, ale také vysoko/nízkotlakých turbín generátorů plynových okruhů v elektrárnách.

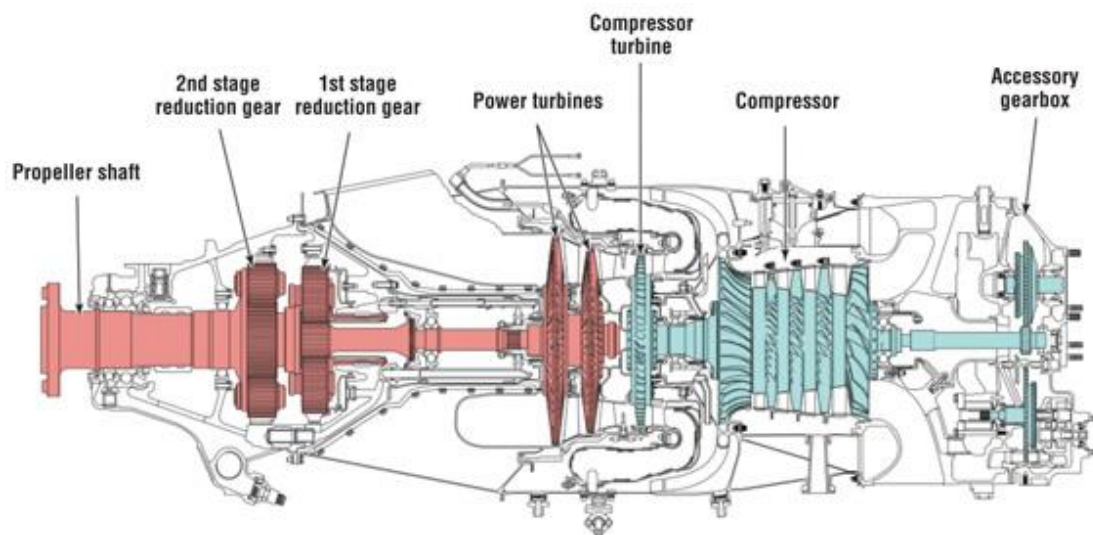
Svou náplní a podstatou s predikcí teplot mezi jednotlivými řezy turbovrtulového letadlového motoru pak byla nejbližší práce popsána v [24], kdy bylo jejím cílem vyvinout ANN pro studium vlivu průtoku vzduchu propulzním systémem letadla. Výsledkem v této práci byly nakonec sítě dvě, přičemž první z nich sloužila pro návrhovou simulaci turbofanového motoru a za účelem dosažení co nejvyšší přesnosti sítí byly otestovány jejich nejrozumnější architektury. Významným zjištěním bylo, že použití různých množství skrytých vrstev pro každý vstupní neuron bylo efektivní volbou modifikací neuronových sítí. Druhá síť pak umožnila studium průtoku vzduchu tímto propulzním systémem, kdy se jejím prostřednictvím simuloval pracovní cyklus motoru jak s chybami, tak i bez nich. V následné části této práce pak byla vyvinuta ANN pro diagnostiku a popis závad tohoto letadlového motoru.

Tím nejzásadnějším zjištěním této rešerše je fakt, že pro predikci různých veličin a vlastností propulzní látky i propulzního systému letadla jsou zpravidla využívány nejen hodnoty teploty, ale mnoho dalších veličin, které chování dějů a procesů těchto strojů ovlivňují. Těmito veličinami bývá například stechiometrický poměr, tedy relevantní množství reaktantů a produktů v chemických reakcích, které jsou založeny na zákonu zachování hmotnosti, dále hodnoty průtoku vzduchu, paliva, vstupního a výstupního tlaku, tedy průtokového objemu, nebo otáčky turbíny.

4 Datová analýza, příprava a zpracování dat z motoru Turboprop

Měření teplot v tomto experimentu bylo provedeno ve dvou lokalitách turbovrtulového letadlového motoru. Data která mám k dispozici, jsou již včetně nejistot, které by jinak bylo nutné pro toto zpracování vypočítat a zanést k hodnotám naměřeným. Jak jsem již výše uvedl, měření probíhalo na bližší nespecifikovaném motoru. Pro popis a znázornění rozmístění obou řezů zavedu jejich značení TPL1 a TPL2, přičemž v prvním případě proběhlo snímání dané veličiny celkem na 20 pozicích, a to na pěti průměrech po 90°. V případě druhém celkem na 35 pozicích, na pěti průměrech vždy po 1/7 obvodu.

Pro názornost lokalit, kde k měření v motoru docházelo, je na obr. 4.1 zobrazena kompresorová část motoru, tedy řez TPL1 modře a výkonová turbína červeně, která pak demonstruje pozici řezu TPL2.



Obr. 4.1: Komponenty turbovrtulového motoru v řezu [26]

Měření probíhalo se vzorkovací frekvencí 100 Hz po dobu přibližně 7,4 minuty, hodnot z řezu TPL1 bylo tedy odečteno téměř 900 000 a z řezu TPL2 přes 1 500 000.

Pro znázornění konkrétních měřených pozic v motoru je na obr. 12 vidět umístění senzorů pro řez TPL1 a na obr. 13 pro řez TPL2. Pro řez TPL1 jsem jednotlivé poloměry,

na kterých byly teploty měřeny, označil číselně z intervalu <1; 5> a úhly písmeny A-D. Pro řez TPL2 bylo roztečné umístění senzorů totožné s řezem předcházejícím, úhlů ale bylo 7, jejich označení je tedy A-G. Níže jsou uvedené tabulky pro znázornění značení jednotlivých senzorů, které jsem zavedl pro jejich jednoznačnou definici.

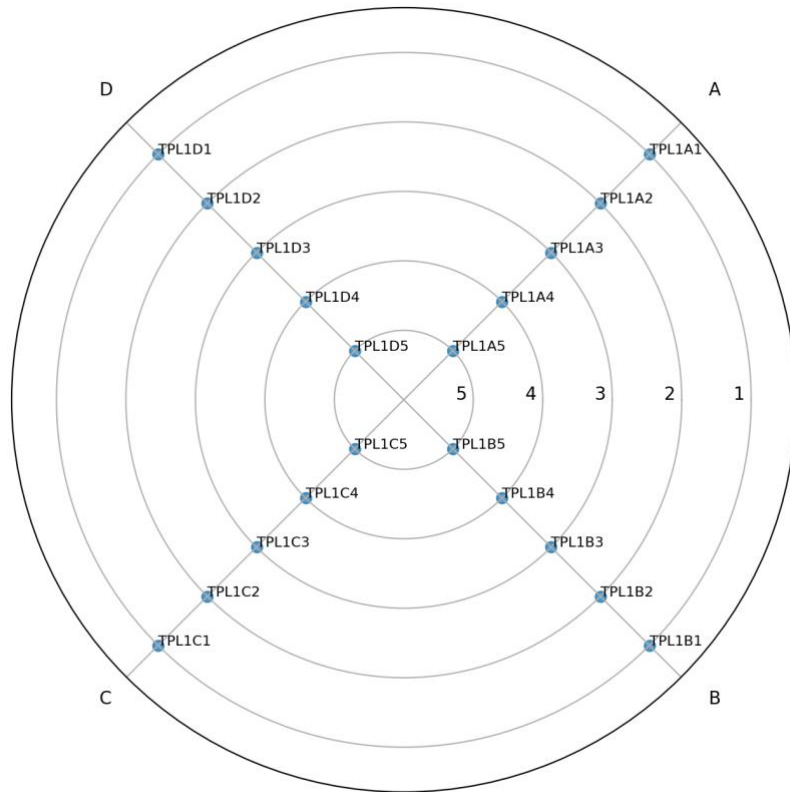
Tab. 4.1: Seznam senzorů z řezu TPL1

Senzor	r: 1	r: 2	r: 3	r: 4	r: 5
úhel A	1.	5.	9.	13.	17.
úhel B	2.	6.	10.	14.	18.
úhel C	3.	7.	11.	15.	19.
úhel D	4.	8.	12.	16.	20.

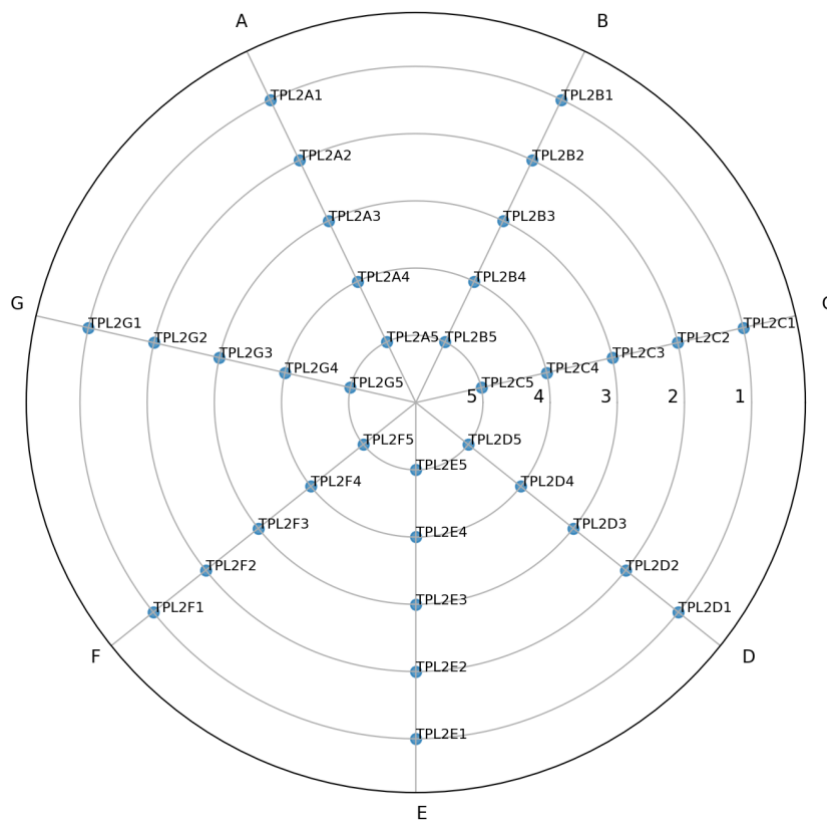
Tab. 4.2: Seznam senzorů z řezu TPL2

Senzor	r: 1	r: 2	r: 3	r: 4	r: 5
úhel A	1.	8.	15.	22.	29.
úhel B	2.	9.	16.	23.	30.
úhel C	3.	10.	17.	24.	31.
úhel D	4.	11.	18.	25.	32.
úhel E	5.	12.	19.	26.	33.
úhel F	6.	13.	20.	27.	34.
úhel G	7.	14.	21.	28.	35.

Zadáním této práce je mimo jiné definovat minimální počet relevantních senzorů z prvního řezu, které by pro predikci teploty v řezu druhém stačily. Z tohoto důvodu nejprve provedu analýzu hlavních komponent. Vzhledem k tomu, že není známa vzájemná poloha a natočení senzorů z řezu TPL1 vůči senzorům z řezu TPL2, následně v této části práce provedu korelační analýzu.



Obr. 4.2: Polohy jednotlivých senzorů v řezu TPL1



Obr. 4.3: Polohy jednotlivých senzorů v řezu TPL2

4.1 Analýza hlavních komponent – PCA

PCA (*Principal Component Analysis*) je statistický algoritmus sloužící ke snížení počtu dimenzí velkého objemu dat, a to při minimální ztrátě jejich výpovědních informací, čímž se ale sníží i potenciální přesnost adaptivního modelu. PCA však provádí redukci počtu dimenzí tak, aby potenciální přesnost adaptivního algoritmu byla ovlivněna jen minimálně. Algoritmus PCA se tedy dá použít pro předzpracování dat a datovou analýzu. Tato analýza sestává z kroků, kterými jsou normalizace, vyjádření vztahu mezi jednotlivými vstupními daty a identifikace hlavních komponent [35].

Cílem normalizace dat, často označované také jako z-scoring, je změnit rozsah hodnot vstupních proměnných tak, aby každá přispívala k analýze PCA stejným poměrem.

Normalizace je definována jako:

$$x_{norm} = \frac{x_i - \bar{x}}{\sigma_i}, \quad (38)$$

kde x_{norm} je normalizovaná hodnota, x_i je hodnota před normalizací, \bar{x} je průměrná hodnota a σ_i je směrodatná odchylka. Dalším krokem je výpočet kovarianční matice, která nám vyjádří, zdali je mezi vstupními daty nějaká provázanost. Její výpočet je:

$$\mathbf{C}_x = \mathbf{X}\mathbf{X}^T \quad (39)$$

$$\mathbf{C}_x = \begin{bmatrix} cov(X_1, X_1) & cov(X_1, X_2) & \cdots & cov(X_1, X_n) \\ cov(X_2, X_1) & cov(X_2, X_2) & \cdots & cov(X_2, X_n) \\ \vdots & \vdots & \ddots & \vdots \\ cov(X_n, X_1) & cov(X_n, X_2) & \cdots & cov(X_n, X_n) \end{bmatrix} \quad (40)$$

Nakonec provedeme analýzu hlavních komponent nalezením vlastních čísel a vektorů kovarianční matice. Máme tedy matici \mathbf{V} , jejíž vektor vlastních čísel nám říká, jak jsou jednotlivé komponenty seřazeny podle důležitosti. Tento vektor má tvar:

$$\mathbf{\Lambda} = [\lambda_1, \lambda_2, \dots, \lambda_n]^T \quad (41)$$

a každý její řádek odpovídá vlastnímu vektoru komponenty:

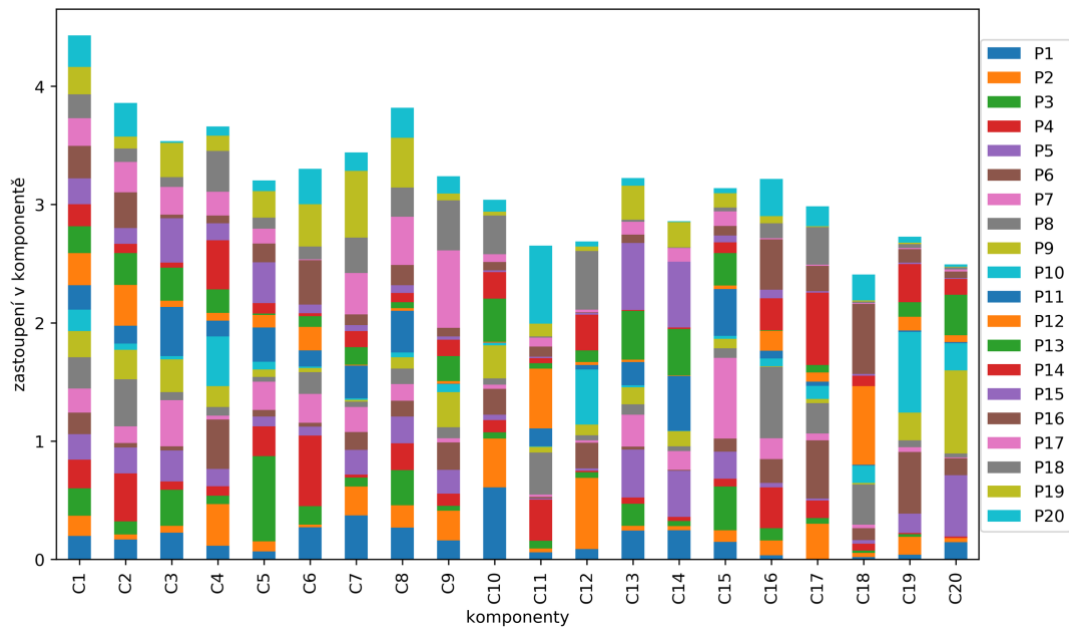
$$\mathbf{V} = [v_1, v_2, \dots, v_n]^T. \quad (42)$$

Tyto vlastní vektory nesou informaci o tom, jak máme původní hodnoty do daných komponent transformovat.

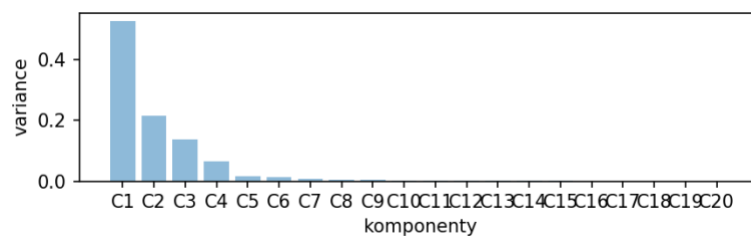
Výsledná matice má pak tvar:

$$\mathbf{X}_t = \mathbf{V}_O \mathbf{X}_z^T, \quad (43)$$

kde \mathbf{V}_O jsou vlastní vektory srovnané podle důležitosti komponenty a \mathbf{X}_z^T je transponovaná matice standardizovaných původních dat. Dimenzionalitu pak redukuje tak, že vezmeme pouze určitý počet sloupců matice \mathbf{X}_t [10].

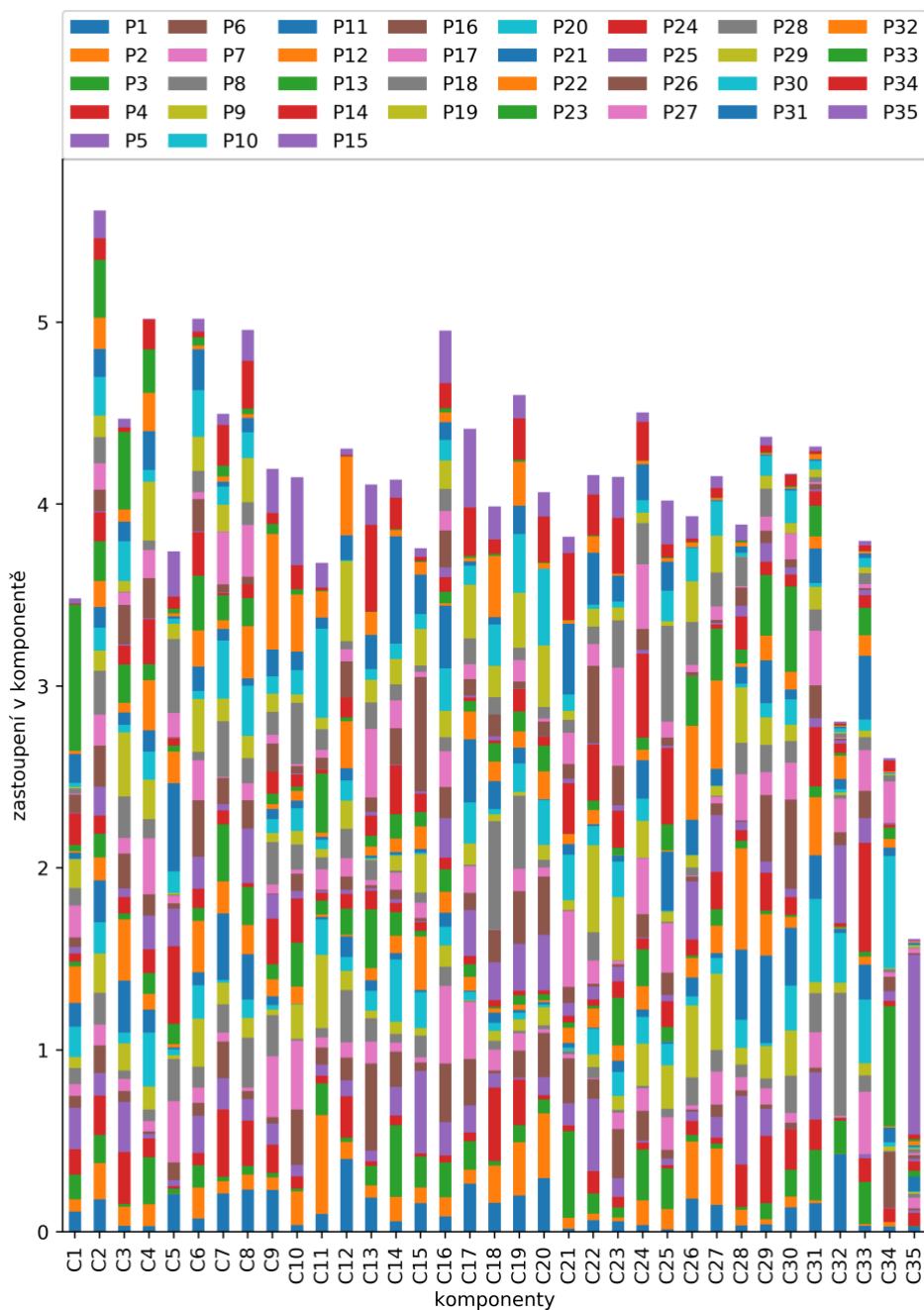


Obr. 4.4: Zastoupení jednotlivých senzorů ve všech komponentách řezu TPL1



Obr. 4.5: Důležitost jednotlivých komponent řezu TPL1

Jak je v úvodu této části uvedeno, PCA analýza se často používá pro zmenšení objemu dat pro zpracování neuronovými sítěmi. Na obr. 4.5 a 4.7 vidíme, že vzájemná provázanost všech 20 senzorů z řezu TPL1 a 35 senzorů z řezu TPL2 je vysoká.



Obr. 4.6: Zastoupení jednotlivých senzorů ve všech komponentách řezu TPL2



Obr. 4.7: Důležitost jednotlivých komponent řezu TPL2

Z analýzy PCA plyne, že nebude možné dimenzionalitu vstupních hodnot redukovat, jelikož se ani v jednom řezu motoru nenacházel senzor, jež by bylo možné zanedbat.

4.2 Korelace senzorů měření

Korelační analýza bude v této práci spočívat ve stanovení korelačních koeficientů mezi 20 senzory z řezu TPL1 <A1; D5> a 35 senzory z řezu TPL2 <A1; G5>. Výsledné korelační koeficienty, na obrázku 14 zobrazené prostřednictvím teplotní mapy, nám řeknou, nakolik jsou senzory na sobě lineárně závislé. Jinými slovy vyjádří nám míru těsnosti lineární vazby mezi sebou.

Výpočet korelace provedeme podle vztahu:

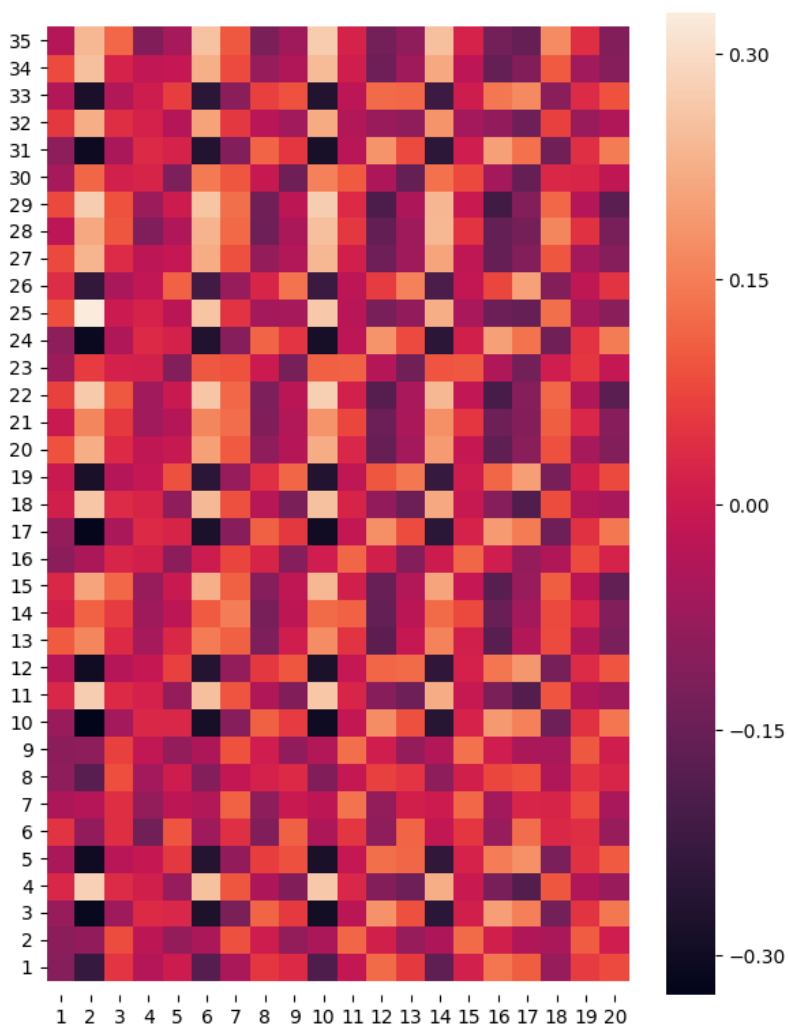
$$\text{corr}(X, Y) = \frac{\text{cov}(X, Y)}{\sigma_X \sigma_Y} = \frac{E[(X - \mu_X)(Y - \mu_Y)]}{\sigma_X \sigma_Y}, \quad (44)$$

kde $\text{cov}(X, Y)$ je kovariance, $E[.]$ je očekávaná hodnota, μ_X, μ_Y jsou střední hodnoty a $\sigma_X \sigma_Y$ je součin směrodatných odchylek.

Hodnoty korelačních koeficientů nabývají hodnot $\langle -1; 1 \rangle$. Pokud $\text{corr}(X, Y) = 1$, veličiny jsou zcela pozitivně lineárně závislé, v opačném případě jsou taktéž nanejvýš lineárně závislé, ale negativně. Pokud se $\text{corr}(X, Y) = 0$, tak mezi veličinami X, Y lineární závislost není.

Z výsledku korelace je patrné, že největší lineární závislost mezi senzory v řezech TPL1 a TPL2 nabývá hodnoty korelačního koeficientu přibližně 0,3, resp. -0,3. Na základě této analýzy vytvořím vstupně-výstupní páry pro trénování modelů sítí, které pro predikci

následně použiji. Z důvodu velkého rozměru teplotní mapy včetně hodnot korelačních koeficientů se tento obrázek nachází mezi přílohami na konci této práce.



Obr. 4.8: Korelační matice vizualizovaná pomocí teplotní mapy

Výsledkem korelace je tedy informace, že nejvyšší lineární závislost senzorů z řezu TPL1 na senzorech z řezu TPL2 jsou senzory TPL1B1, TPL1B2, TPL1B3 a TPL1B4. Tyto senzory tedy použiji jako vstupní vektor neuronových sítí.

5 Experimentální analýza a návrh neuronových sítí pro predikci teplot

Pro účely této práce jsou použity neuronové sítě pro učení s učitelem. Pro učení těchto sítí je používáno vstupně-výstupních hodnot, které jsou v našem případě reprezentovány naměřenými teplotami v řezu TPL1 jako vstupy a v řezu TPL2 jako výstupy. Vzhledem k výsledku korelace z předcházející kapitoly však budou jako vstupy uvažovány jen senzory 2., 6., 10. a 14 z řezu TPL1.

5.1 LSTM (Long-Short Term Memory)

Vzhledem k vzorkovací frekvenci senzorů (100 Hz) a rychlosti proudění propulzní látky letadlovým motorem nelze zaznamenat dynamičnost změn teploty mezi měřeními řezu Z tohoto důvodu se jeví jako nepraktické rekurentní neuronové sítě typu Long-Short Term Memory aplikovat.

Sítě LSTM patří do skupiny rekurentních neuronových sítí, které mají mnohem složitější architekturu jednotlivých neuronů než sítě dopředné. Umí řešit problém mizejících gradientů (*vanishing gradient*) a jejich struktura obsahuje prvky určené k zachování dlouhodobé informace. Oproti své původní podobě jsou vylepšeny o mechanismus zapomínajících bran, který jim umožňuje efektivně zapomínat nepotřebné informace. Bránami jsou myšleny neurony, které řídí vstup, výstup a zapomínání centrálního neuronu, ve kterém může být uchována hodnota po delší dobu. Brány jsou ovlivněny hodnotami svých vstupů, které představují interakci se zbytkem neuronové sítě. Funkce jednotlivých bran jsou následující:

- vstupní brána: určuje, jestli se bude vstup propagovat do centrální jednotky
- výstupní brána: určuje, jestli se bude výstup centrální jednotky dále
- zapomínající brána: určuje, jestli se v centrální jednotce hodnota uchová [38]

5.2 MLP (Multi Layer Perceptron)

První síť, kterou jsem použil byla více vrstvá perceptronová síť. Tato síť včetně Perceptronu byla podrobně popsána v 2.1. Pro účely této práce se jako nejvhodnější z hlediska dosažení nejlepšího výsledku predikce ukázala konfigurace této sítě s parametry o těchto hodnotách:

celkový počet vrstev = 3

aktivační funkce v pořadí: Sigmoida, Sigmoida, Lineární

optimalizační algoritmus: Adam Optimizer

počet neuronů ve vrstvách v pořadí = 4, 16, 35

learnig rate = 0,05

batch size = 100

počet epoch = 20

Na konci této práce se jako příloha nachází kompletní tabulka s hodnotami maximálních chyb (max e), absolutních velikostí středních kvadratických chyb (MSE) a směrodatných odchylek (σ) predikce teplot pro každý senzor řezu TPL2. Zde uvedu pouze zhodnocení pro nejlepší, průměrnou (*median*) a nejhorší predikci teplot senzorů. V tabulce 5.1 je kvalita sítě hodnocena podle maximální chyby predikované teploty, v tab. 5.2 podle MSE a v tab. 5.3 podle směrodatné odchylky. V krabicového grafu (obr. 5.1) je možnost porovnat statistické charakteristiky jednotlivých senzorů.

Tab. 5.1: Kvalita predikce sítí MLP podle maximální chyby

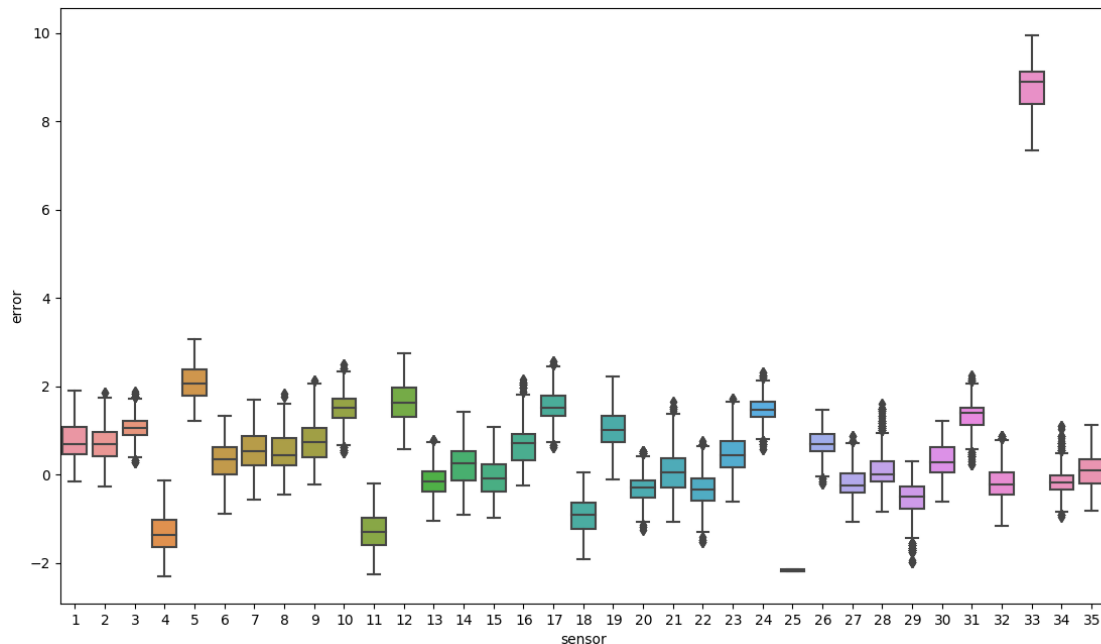
Predikce	Senzor	Označení sen.	Maximální chyba [°C]	abs. MSE [°C]	Směrodatná odchylka [°C]
Nejlepší	18.	TPL2D3	0,045	0,895	0,431
Průměrná	28.	TPL2G4	1,946	0,462	0,43
Nejhorší	33.	TPL2E5	10,014	8,853	0,488

Tab. 5.2: Kvality predikce sítí MLP podle MSE

Predikce	Senzor	Označení sen.	Maximální chyba [°C]	abs. MSE [°C]	Směrodatná odchylka [°C]
Nejlepší	34.	TPL2F5	1,449	0,262	0,29
Průměrná	7.	TPL2G1	2,128	0,995	0,42
Nejhorší	33.	TPL2E5	10,014	8,853	0,488

Tab. 5.3: Kvality predikce sítí MLP podle směrodatné odchylky

Predikce	Senzor	Označení sen.	Maximální chyba [°C]	abs. MSE [°C]	Směrodatná odchylka [°C]
Nejlepší	25.	TPL2D4	1,916	1,947	0,028
Průměrná	15.	TPL2A3	1,527	0,459	0,405
Nejhorší	21.	TPL2G3	2,07	0,543	0,491



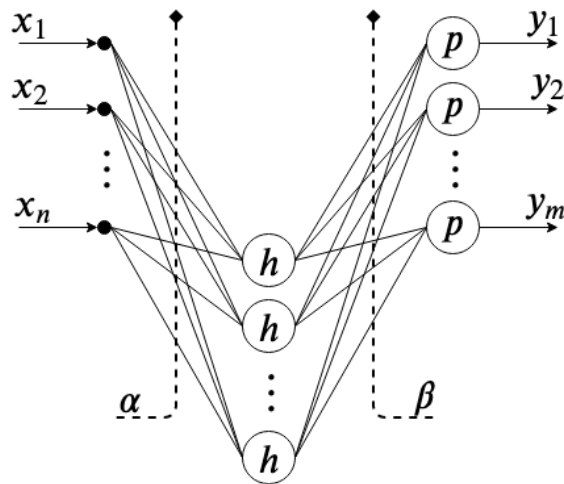
Obr. 5.1: Porovnání výsledků predikce senzorů sítí MLP

5.3 ELM (Extreme Learning Machine)

Tato síť je tvořena obvykle třemi vrstvami, tedy vstupní, skrytou a výstupní. Charakteristickým znakem ELM pak je, že váhy a bias vstupní vrstvy (na obr. 5.1 značeny jako matice α) jsou po celou dobu uvažovány jako konstanty. Jinými slovy optimalizace těchto parametrů sítě probíhá pouze mezi vrstvou skrytou a výstupní a algoritmus Back Propagation je tedy aplikován pouze pro parametry mezi těmito dvěma vrstvami (na obr. 5.1 značeny jako matice β). U této sítě neurony ve skryté vrstvě označujeme jako *hidden nodes* (h).

Jako výsledná konfigurace tohoto modelu byla zvolena síť s následujícími parametry:

celkový počet vrstev: 3
 aktivační funkce: Identity
 hidden nodes = 100



Obr. 5.2: Architektura ELM sítě

Vztah pro výstup z této sítě můžeme vyjádřit jako:

$$\mathbf{y} = \boldsymbol{\beta} \cdot \mathbf{h} \cdot (\boldsymbol{\alpha} \cdot \mathbf{x}) , \quad (45)$$

kde $\boldsymbol{\beta}$ je matice vah mezi skrytou a výstupní vrstvou, \mathbf{h} můžeme uvažovat jako vektor somatických operací neuronů, $\boldsymbol{\alpha}$ je maticí náhodných vah mezi první a druhou vrstvou neuronové sítě a \mathbf{x} je vstupem sítě.

Tab. 5.4 hodnotí kvalitu predikce podle dosažené maximální chyby ($max e$), tab. 5.5 podle absolutní hodnoty střední kvadratické chyby (MSE) a tab. 5.6 podle velikosti směrodatné odchylky (σ). Na obr. 5.3 jsou porovnány statistické charakteristiky jednotlivých senzorů prostřednictvím krabicového grafu. Kompletní tabulka výsledků sítě je opět umístěna mezi přílohami.

Tab. 5.4: Kvalita predikce sítí ELM podle maximální chyby

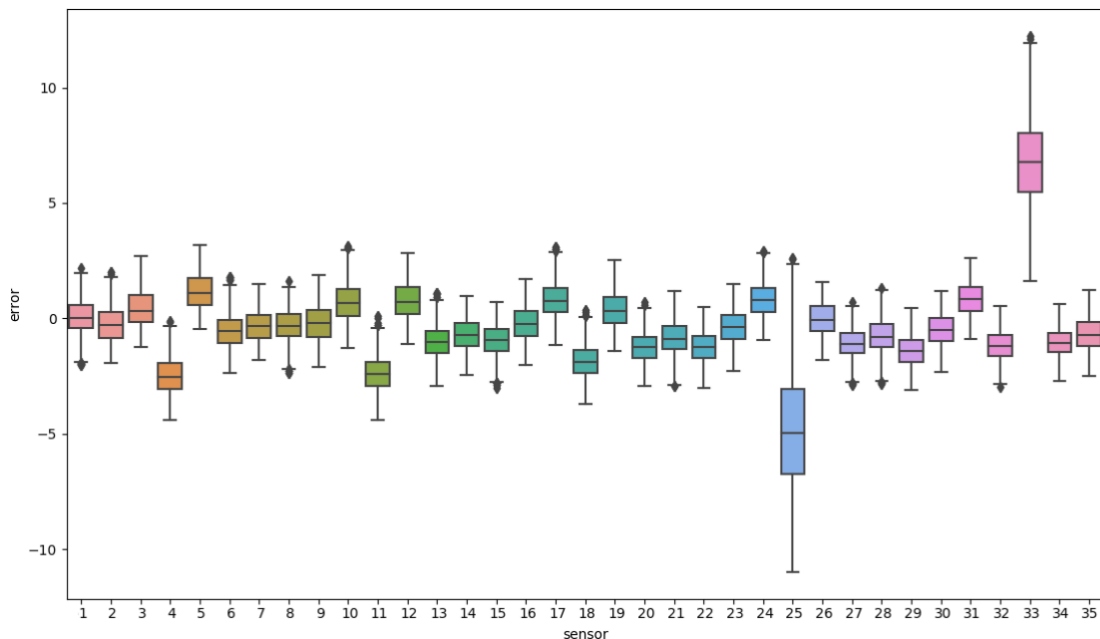
Predikce	Senzor	Označení sen.	Maximální chyba [°C]	abs. MSE [°C]	Směrodatná odchylka [°C]
Nejlepší	11.	TPL2D2	0,086	2,394	0,775
Průměrná	7.	TPL2G1	1,483	0,608	0,667
Nejhorší	33	TPL2E5	12,246	6,862	1,953

Tab. 5.5: Kvalita predikce sítí ELM podle MSE

Predikce	Senzor	Označení sen.	Maximální chyba [°C]	abs. MSE [°C]	Směrodatná odchylka [°C]
Nejlepší	26.	TPL2E4	1,588	0,571	0,689
Průměrná	17.	TPL2C3	3,112	0,896	0,766
Nejhorší	33.	TPL2E5	12,246	6,862	1,953

Tab. 5.6: Kvalita predikce sítí ELM podle směrodatné odchylky

Predikce	Senzor	Označení sen.	Maximální chyba [°C]	abs. MSE [°C]	Směrodatná odchylka [°C]
Nejlepší	34.	TPL2F5	0,641	1,066	0,623
Průměrná	24.	TPL2C4	2,9	0,897	0,739
Nejhorší	25.	TPL2D4	2,622	4,932	2,6



Obr. 5.3: Porovnání výsledků predikce senzorů sítí ELM

5.4 RVFL (Random Vector Function Link)

Sít' RVFL je svou topologií velmi podobná síti ELM. Zásadním rozdílem je pak výpočet výstupu, kdy se k jeho hodnotám, v našem případě 35 přičítá hodnota součinnu vektoru $\boldsymbol{\gamma}$ a matice vstupních hodnot \mathbf{X} . Také u této síti neurony ve skryté vrstvě označujeme jako *hidden nodes* (h). Výstup z této síti se síť se vypočítá podle vztahu:

$$\mathbf{y} = \boldsymbol{\beta} \cdot \mathbf{h} \cdot (\boldsymbol{\alpha} \cdot \mathbf{x}) + \boldsymbol{\gamma} \cdot \mathbf{X} , \quad (46)$$

kde $\boldsymbol{\beta}$ je matice vah mezi skrytou a výstupní vrstvou, \mathbf{h} můžeme uvažovat jako vektor somatických operací neuronů, $\boldsymbol{\alpha}$ ja maticí náhodných vah mezi první a druhou vrstvou neuronové síti a \mathbf{x} je vstupem síti. Po odzkoušení mnoha nastavení hodnot h , učícího parametru r a velikosti dávky (*batch*) pro jednotlivé epochy, vyšla jako nejvhodnější konfigurace následující:

celkový počet vrstev: 3

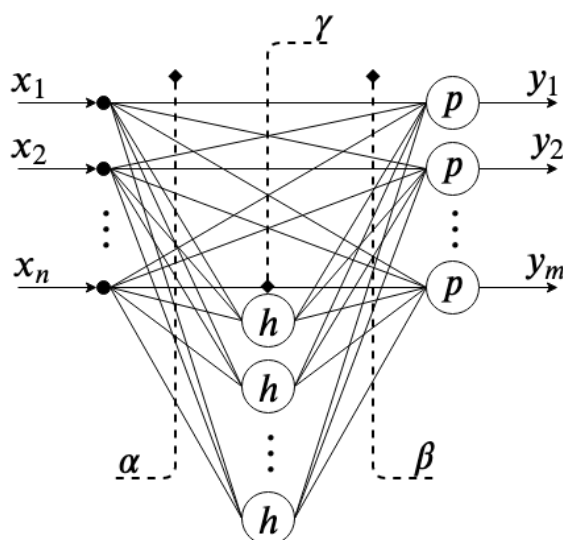
hidden nodes = 8

optimalizační algoritmus: Adam Optimizer

learnig rate = 0,0035

batches = 10

epoch = 3500



Obr. 5.4: Architektura RVFL síť

Kompletní tabulka hodnot maximální chyby, absolutní velikosti střední kvadratické chyby a směrodatné odchylky predikce teplot pro každý senzor řezu TPL2 se opět nachází v příloze. V tab 5.4 je kvalita sítě hodnocena podle maximální chyby predikované teploty, v tab 5.5 podle MSE a v tab 5.6 podle směrodatné odchylky. Na obr. 5.5 opět vidíme srovnání statistických charakteristik jednotlivých senzorů prostřednictvím krabicového grafu. Kompletní tabulka výsledků sítě je opět umístěna mezi přílohami.

Tab. 5.7: Kvalita predikce sítě RVFL podle maximální chyby

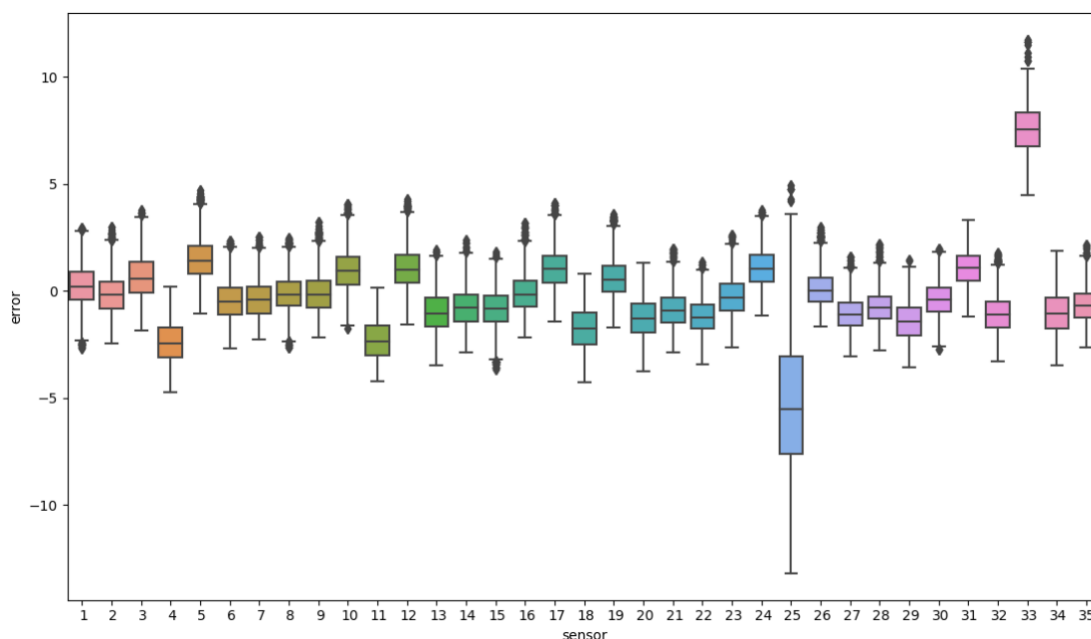
Predikce	Senzor	Označení sen.	Maximální chyba [°C]	abs. MSE [°C]	Směrodatná odchylka [°C]
Nejlepší	11.	TPL2D2	0,151	2,311	0,881
Průměrná	8.	TPL2A2	2,462	0,678	0,839
Nejhorsí	33.	TP2E5	11,735	7,531	1,209

Tab. 5.8: Kvalita predikce sítě RVFL podle MSE

Predikce	Senzor	Označení sen.	Maximální chyba [°C]	abs. MSE [°C]	Směrodatná odchylka [°C]
Nejlepší	26.	TPL2E4	2,994	0,658	0,824
Průměrná	10.	TPL2C2	4,043	1,116	0,982
Nejhorsí	33.	TPL2E5	11,735	7,531	1,209

Tab. 5.9: Kvalita predikce sítě RVFL podle směrodatné odchylky

Predikce	Senzor	Označení sen.	Maximální chyba [°C]	abs. MSE [°C]	Směrodatná odchylka [°C]
Nejlepší	27.	TPL2F4	1,568	1,15	0,787
Průměrná	20.	TPL2F3	1,305	1,352	0,921
Nejhorsí	25.	TPL2D4	4,917	5,573	3,331



Obr. 5.5: Porovnání výsledků predikce senzorů sítě RVFL

5.5 HONU (Higher Order Neural Unite)

Poslední model který jsem použil, byly sítě tvořeny polynomy vyššího stupně. Tato síť byla podrobně popsána v části 2.1. Pro predikci teplot jsem použil LNU, QNU i CNU. Vzhledem k neuspokojivým výsledkům predikce teploty modelem CNU, budou tyto výsledky predikce polynomu třetího stupně zobrazeny pouze mezi přílohami a to pouze pro ilustrativní charakter. Z tohoto důvodu nebudou v této práci ani uvedeny grafy průběhu teplot.

Nejlepších výsledků sítě bylo dosaženo při její konfiguraci o hodnotách:

optimalizační algoritmus: Levenber-Marquardt Algorithm

epoch = 30

dumping factor = 0,2

5.5.1 LNU

Níže uvedené tabulky mezi sebou porovnávají úspěšnost predikce modelem *Linear Neural Unite* v závislosti na parametrech maximální chyby (tab: 5.10), střední kvadratické chyby (tab. 5.11) a směrodatné odchylky (tab. 5.12). Na obr. 5.6 je krabicový graf pro porovnání senzorů predikovaných modelem LNU.

Tab. 5.10: Kvalita predikce sítí LNU podle maximální chyby

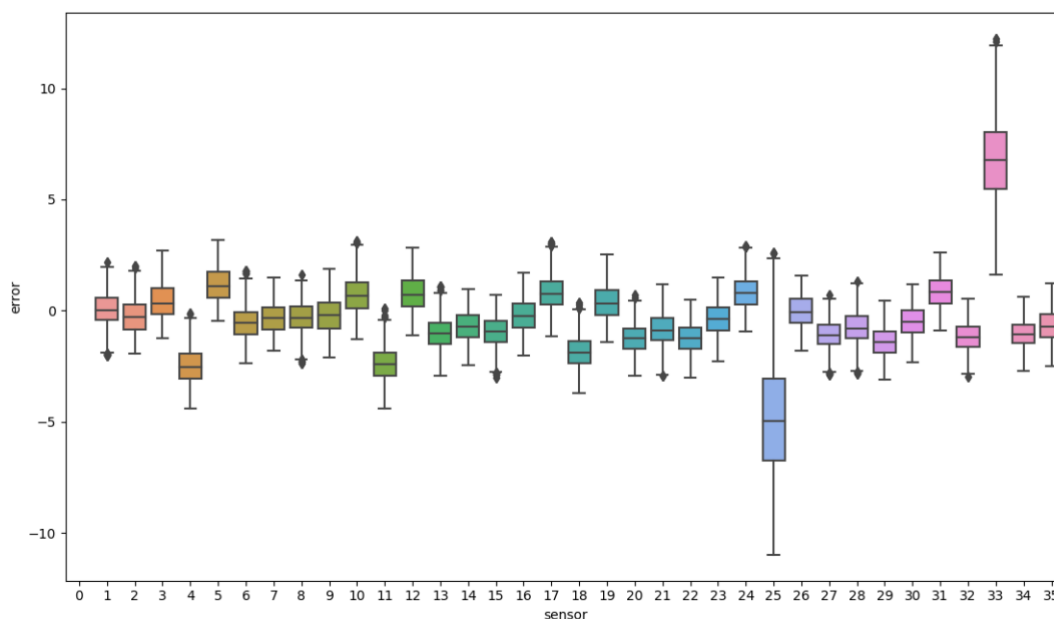
Predikce	Senzor	Označení sen.	Maximální chyba [°C]	abs. MSE [°C]	Směrodatná odchylka [°C]
Nejlepší	7.	TPL2G1	1,805	0,608	0,667
Průměrná	12.	TPL2E2	2,854	0,894	0,781
Nejhorší	33.	TPL2E5	12,246	6,862	1,953

Tab. 5.11: Kvalita predikce sítí LNU podle MSE

Predikce	Senzor	Označení sen.	Maximální chyba [°C]	abs. MSE [°C]	Směrodatná odchylka [°C]
Nejlepší	26.	TPL2E4	1,813	0,571	0,689
Průměrná	17.	TPL2C3	3,112	0,896	0,766
Nejhorší	33.	TPL2E5	12,246	6,862	1,953

Tab. 5.12: Kvalita predikce sítí LNU podle směrodatné odchylky

Predikce	Senzor	Označení sen.	Maximální chyba [°C]	abs. MSE [°C]	Směrodatná odchylka [°C]
Nejlepší	34.	TPL2F5	2,716	1,066	0,623
Průměrná	24.	TPL2C4	2,9	0,897	0,739
Nejhorší	25.	TPL2D4	10,985	4,932	2,6



Obr. 5.6: Porovnání výsledků predikce senzorů sítí LNU

5.5.2 QNU

Níže uvedené tabulky mezi sebou porovnávají úspěšnost predikce modelem *Quadratic Neural Unite* v závislosti na parametrech maximální chyby (tab. 5.13), střední kvadratické chyby (tab. 5.14) a směrodatné odchylky (tab. 5.15). Na obr. 5.7 se nachází krabicový graf pro model QNU.

Tab. 5.13: Kvalita predikce sítě QNU podle maximální chyby

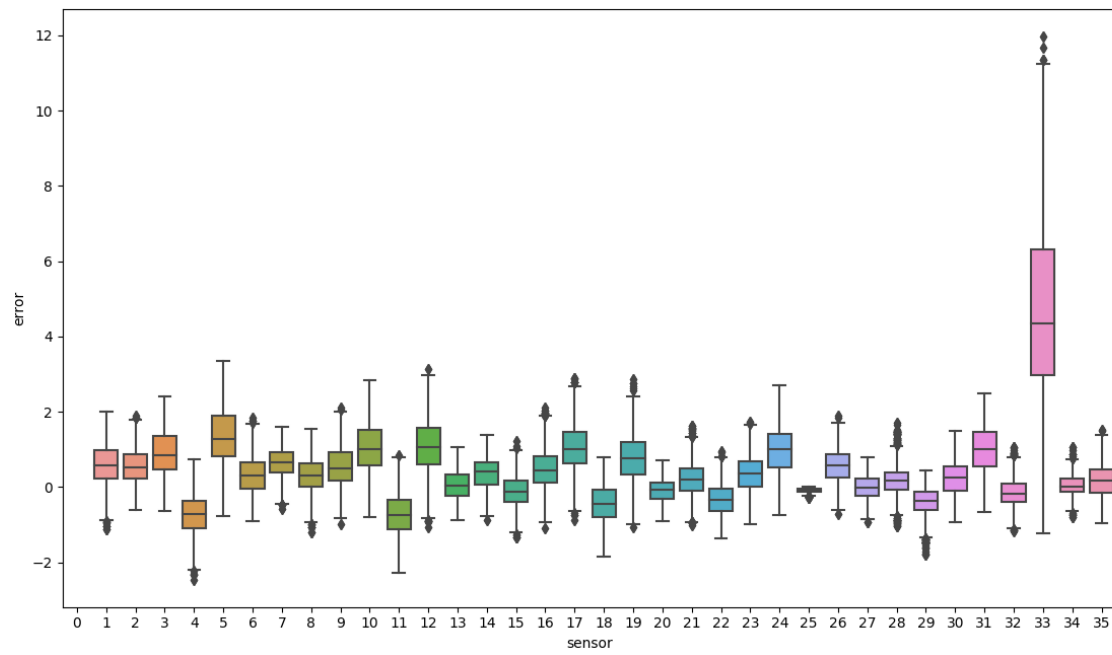
Predikce	Senzor	Označení sen.	Maximální chyba [°C]	abs. MSE [°C]	Směrodatná odchylka [°C]
Nejlepší	25.	TPL2D4	0,281	0,1	0,056
Průměrná	6.	TPL2F1	1,841	0,468	0,498
Nejhorší	33.	TPL2E5	11,978	4,727	2,355

Tab. 5.14: Kvalita predikce sítě QNU podle MSE

Predikce	Senzor	Označení sen.	Maximální chyba [°C]	abs. MSE [°C]	Směrodatná odchylka [°C]
Nejlepší	25.	TPL2D4	0,281	0,1	0,056
Průměrná	18.	TPL2D3	1,85	0,55	0,503
Nejhorší	33.	TPL2E5	11,978	4,727	2,355

Tab. 5.15: Kvalita predikce sítě QNU podle směrodatné odchylky

Predikce	Senzor	Označení sen.	Maximální chyba [°C]	abs. MSE [°C]	Směrodatná odchylka [°C]
Nejlepší	25.	TPL2D4	0,281	0,1	0,056
Průměrná	6.	TPL2F1	1,841	0,468	0,498
Nejhorší	33.	TPL2E5	11,978	4,727	2,355



Obr. 5.7: Porovnání výsledků predikce senzorů sítě QNU

6 Aplikace a zhodnocení navrhnutých modelů sítí

Všechny modely použitých sítí byly trénovány a následně testovány na různých velkých trénovacích a validačních množinách z poskytnutých dat. Výsledným zjištěním bylo, že byly zanedbatelné rozdíly, pokud jsem síť trénoval na první a následně testoval na druhé polovině poskytnutých dat, nebo trénování probíhalo na celé množině kromě posledních pěti, resp. dvou tisících měření a následně síť na těchto původně skrytých datech testoval.

Z uvedeného porovnání senzorů pomocí krabicového grafu (obr.: 5.3, 5.5, 5.6 a 5.7) i zde z obr. 5.1 a z následujících porovnání průběhů teplot reálných a predikovaných lze jednoznačně stanovit, že senzor 33. senzor z řezu TPL2, tedy TPL2E5 byl pravděpodobně poškozen, neboť odečítání jeho teploty bylo velmi odlišné oproti senzorům ostatním.

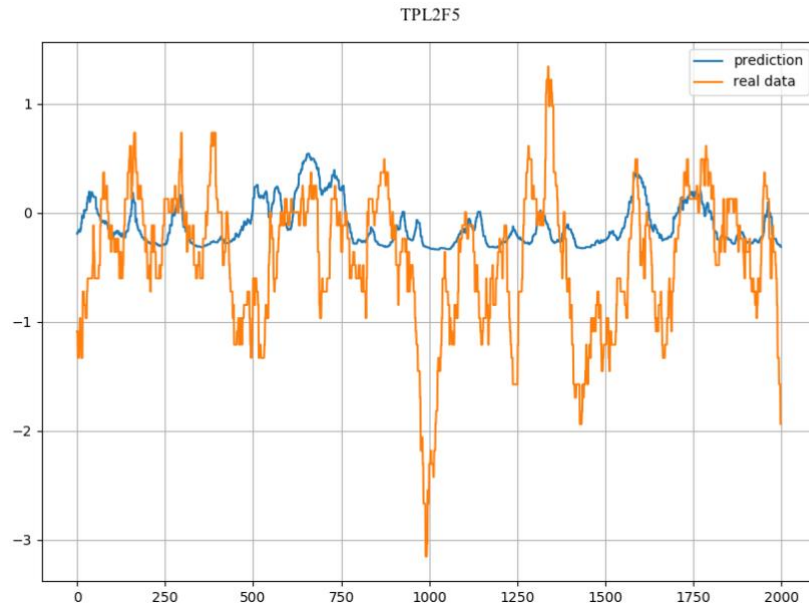
Následující porovnání výkonosti jednotlivých neuronových sítí bude zohledněno dle velikostí středních kvadratických chyb (MSE). V části 6.1 bude porovnán průběh teploty skutečně naměřené s teplotou predikovanou, a to na posledních 2000 měřeních, které měla síť během svého učení skrytá a následně tato data predikovala.

Průběhy všech teplot v této kapitole jsou pro lepší přehlednost z důvodu velkého rozptylu zobrazeny jako bezrozměrné veličiny po jejich standardizaci.

6.1 Vyhodnocení modelu MLP

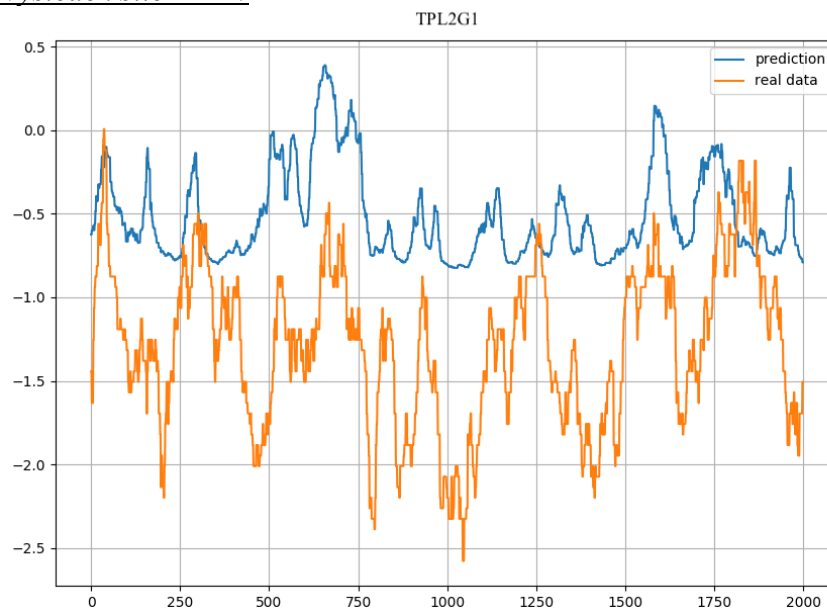
Na obr. 6.1 je zobrazen průběh teplot, kdy dle MSE vícevrstvá perceptronová síť dosáhla nejpřesnější predikce, tedy nejmenší chyby. Na obr. 6.2 je pak vidět průměrný výkon této sítě, kdy ze souboru hodnot všech MSE, pro každý senzor z řezu TPL2 byla střední chyba vybrána jako MEDIAN z tohoto souboru. Na obr. 6.3 je pak průběh teplot senzoru, u kterého byla predikce nejhorší.

Nejlepší výsledek sítě MLP:



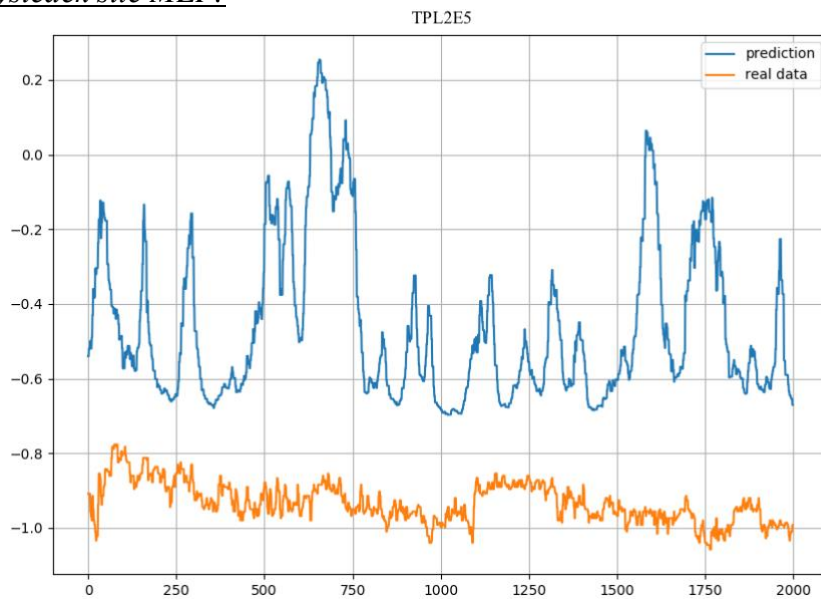
Obr. 6.1: Graf průběhu teploty reálné a predikované 34. senzoru

Průměrný výsledek sítě MLP:



Obr. 6.2: Graf průběhu teploty reálné a predikované 7. senzoru

Nejhorší výsledek sítě MLP:

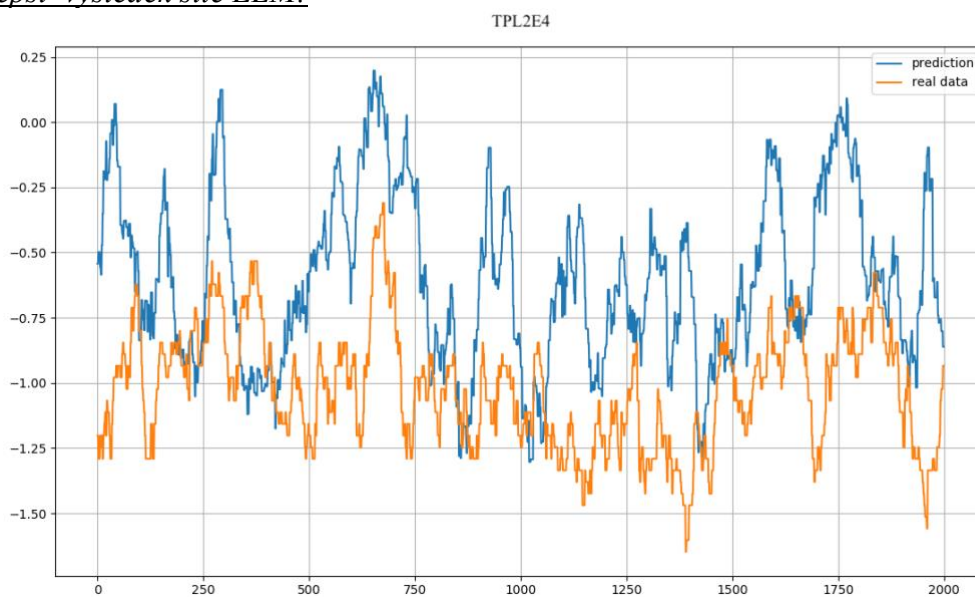


Obr. 6.3: Graf průběhu teploty reálné a predikované 33. senzoru

6.2 Vyhodnocení modelu ELM

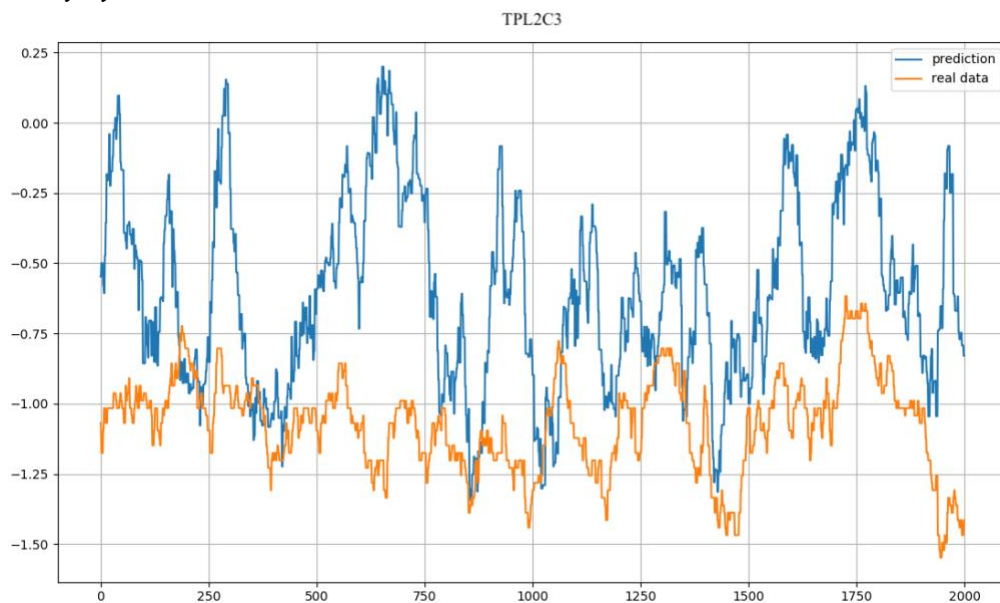
Na grafech obr. 6.7, 6.8 a 6.9 je zobrazen průběh teplot senzorů, kdy dle MSE bylo sítí Extreme Learning Machine dosaženo přesnosti predikce ve stejném smyslu, jako v předcházející části této kapitoly.

Nejlepší výsledek sítě ELM:



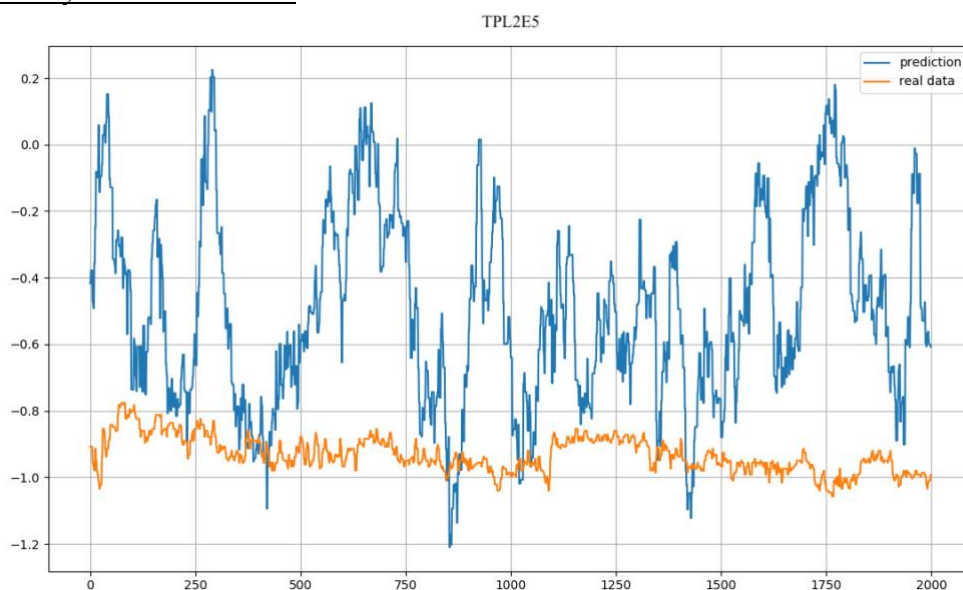
Obr. 6.4: Graf průběhu teploty reálné a predikované 26. senzoru

Průměrný výsledek sítě ELM:



Obr. 6.5: Graf průběhu teploty reálné a predikované 17. senzoru

Nejhorší výsledek sítě ELM:

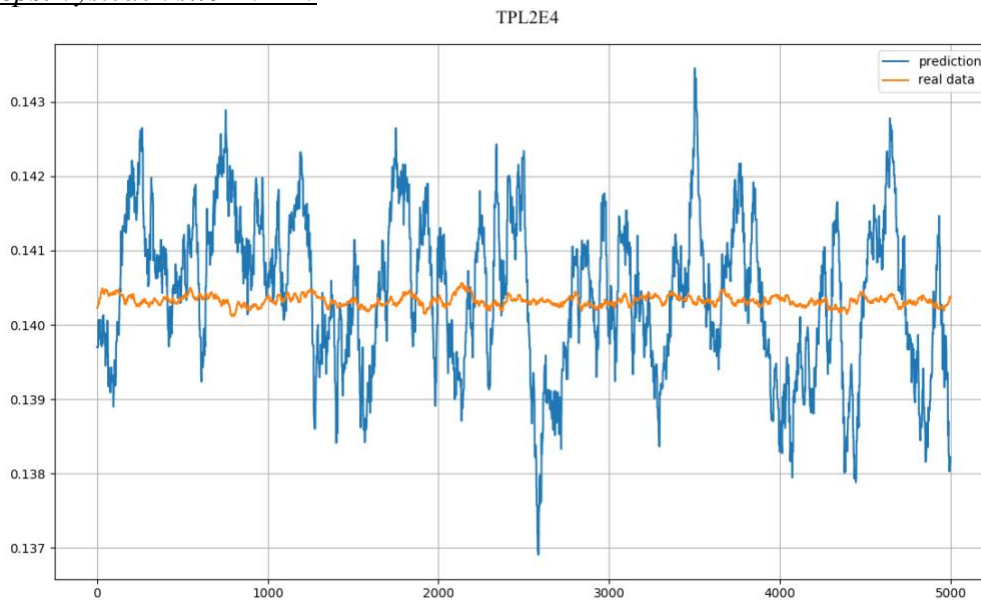


Obr. 6.6: Graf průběhu teploty reálné a predikované 33. senzoru

6.3 Vyhodnocení modelu RVFL

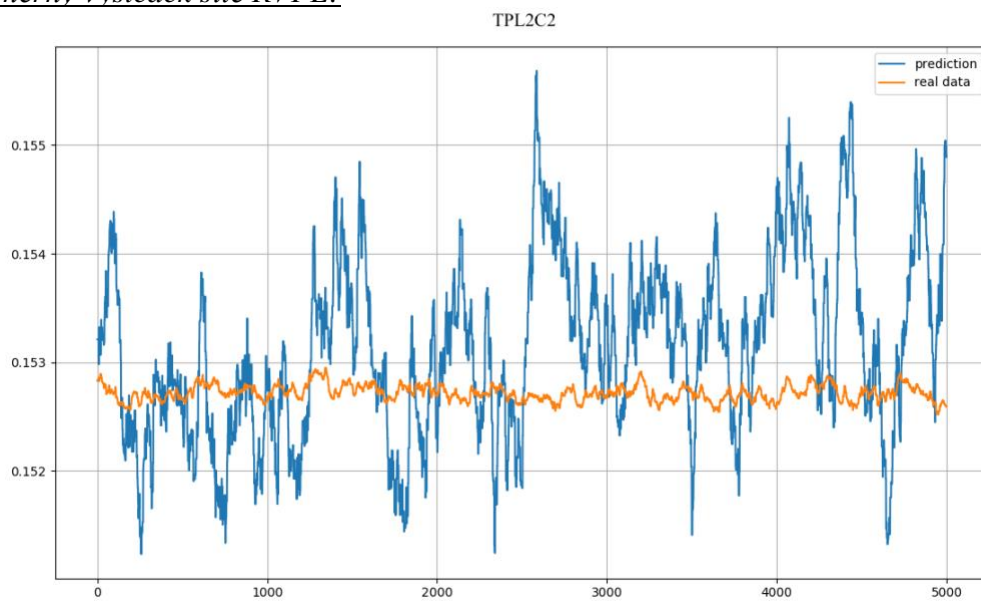
Na grafech obr. 6.7, 6.8 a 6.9 je zobrazen průběh teplot senzorů, kdy dle MSE bylo sítí Random Vector Function Link dosaženo přesnosti predikce ve stejném smyslu, jako v předcházející části této kapitoly.

Nejlepší výsledek sítě RVFL:



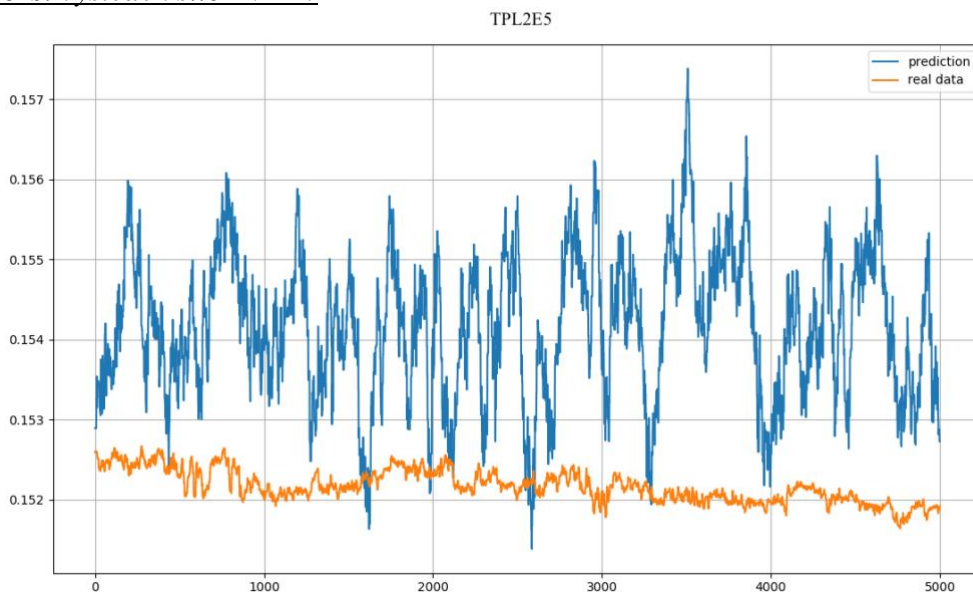
Obr. 6.7: Graf průběhu teploty reálné a predikované 26. senzoru

Průměrný výsledek sítě RVFL:



Obr. 6.8: Graf průběhu teploty reálné a predikované 26. senzoru

Nejhorší výsledek sítě RVFL:

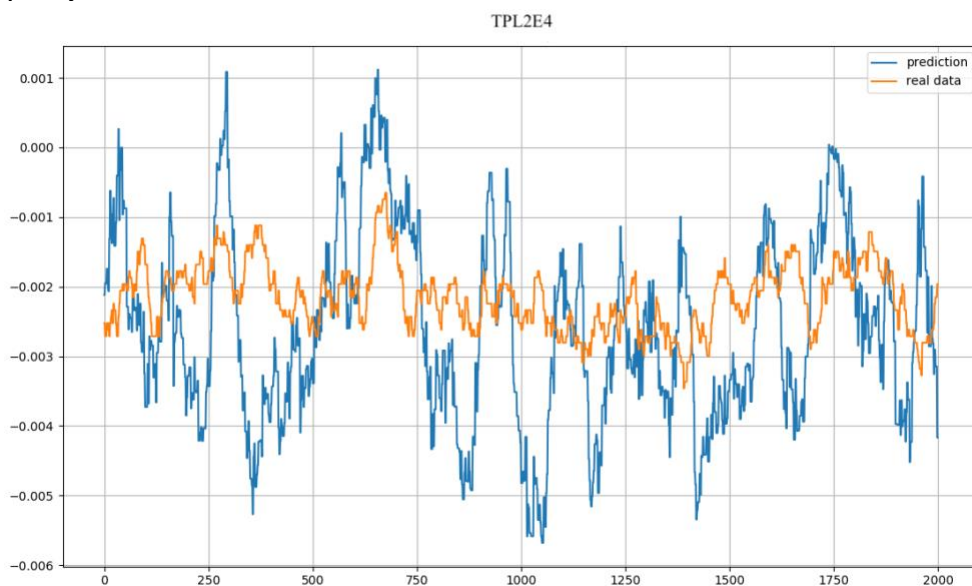


Obr. 6.9: Graf průběhu teploty reálné a predikované 26. senzoru

6.4 Vyhodnocení modelu LNU

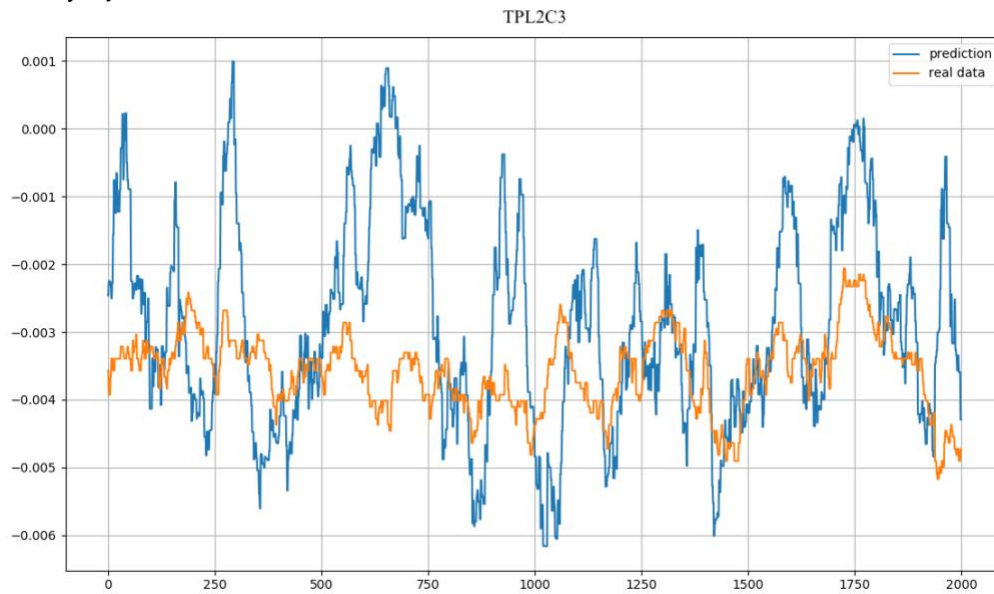
Na grafech obr. 6.10, 6.11 a 6.12 je zobrazen průběh teplot senzorů, kdy dle MSE bylo polynodem Linear Neural Unite dosaženo přesnosti predikce ve stejném smyslu, jako v předcházející části této kapitoly.

Nejlepší výsledek LNU:



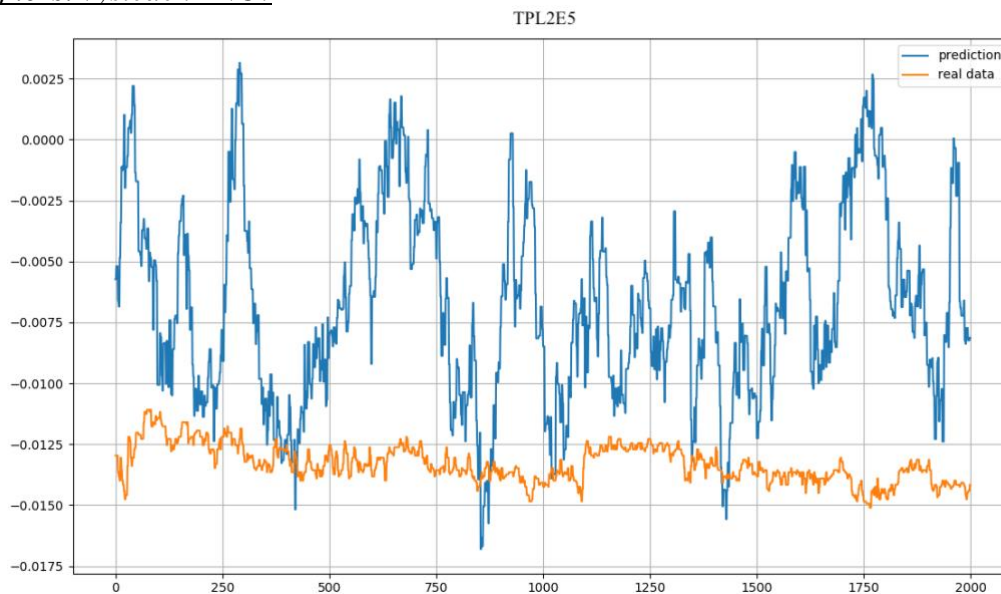
Obr. 6.10: Graf průběhu teploty reálné a predikované 26. senzoru

Průměrný výsledek LNU:



Obr. 6.11: Graf průběhu teploty reálné a predikované 17. senzoru

Nejhorší výsledek LNU:

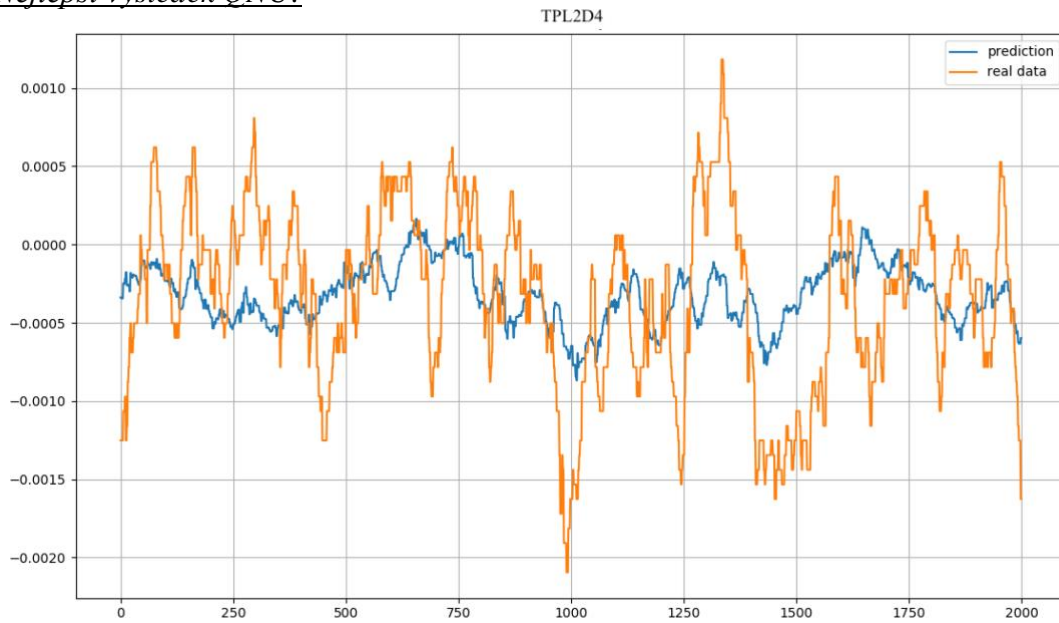


Obr. 6.12: Graf průběhu teploty reálné a predikované 33. senzoru

6.5 Vyhodnocení modelu QNU

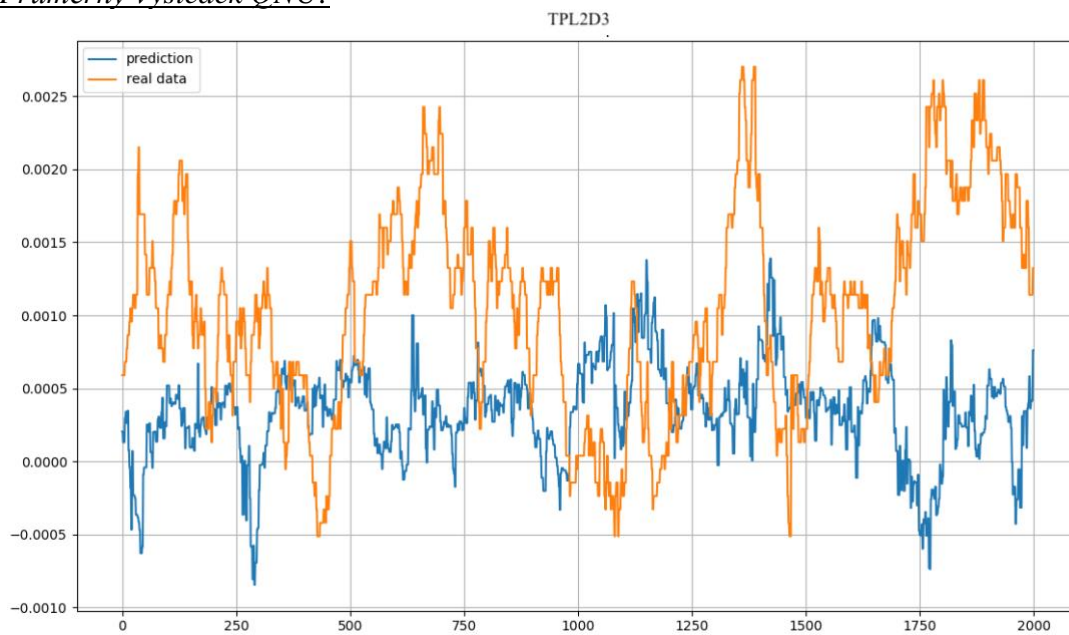
Na grafech obr. 6.13, 6.14 a 6.15 je zobrazen průběh teplot, kdy dle MSE polynomem Quadratic Neural Unite bylo dosaženo přesnosti predikce ve stejném smyslu, jako v předcházející části této kapitoly.

Nejlepší výsledek QNU:



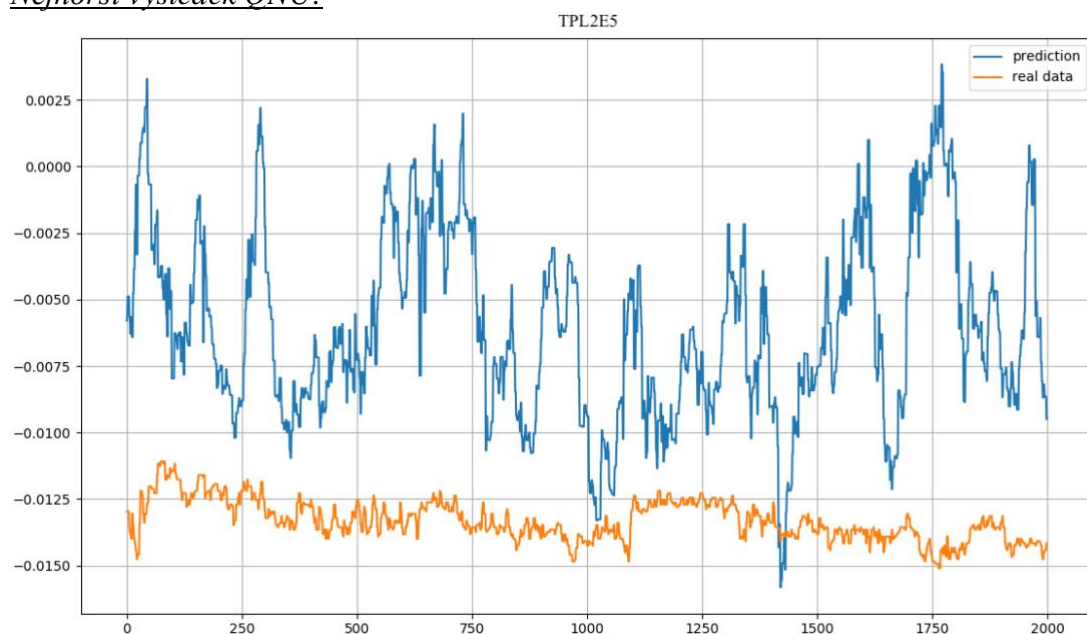
Obr. 6.13: Graf průběhu teploty reálné a predikované 25. senzoru

Průměrný výsledek QNU:



Obr. 6.14: Graf průběhu teploty reálné a predikované 18. senzoru

Nejhorší výsledek QNU:



Obr. 6.15: Graf průběhu teploty reálné a predikované 33. senzoru

6.6 Zhodnocení použitých neuronových sítí

Uvedené modely použitých neuronových sítí se od se diametrálně neliší pokud budeme uvažovat i výsledky sítě CNU, které jsou uvedeny v příloze. Pokud však nebudeme uvažovat predikci polynomem třetího stupně, jejich přesnosti se liší znatelně. O tomto faktu vypovídá tab. 6.1, kde jsou uvedeny dosažené průměrné hodnoty maximální chyby ($\overline{\max e}$), střední kvadratické chyby ($\overline{\text{abs. MSE}}$) a směrodatné odchylky ($\overline{\sigma}$) pro všech 35 senzorů řezu TPL2 a každou použitou neuronovou síť vyjma CNU.

Tab. 6.1: Dosažené průměrné hodnoty aplikovaných modelů sítí

ANN	$\overline{\max e}$ [°C]	$\overline{\text{abs. MSE}}$ [°C]	$\overline{\sigma}$ [°C]
MLP	2,107	1,293	0,388
ELM	1,869	1,288	0,822
RVFL	2,804	1,426	0,987
LNU	3,268	1,288	0,822
QNU	2,162	0,722	0,543

Na základě dosažených výsledků tedy můžeme říci, že pokud budeme výkonost a přesnost použitých neuronových sítí hodnotit měřítkem hodnoty střední kvadratické chyby, nejlepší volbou pro predikci teplot v řezu TPL2 z teplot řezu TPL1 je neuronová síť HONU, konkrétně pak polynom druhého stupně Quadratic Neural Unite.

7 Závěr

Aplikací, kde můžeme neuronové sítě využít, je celá řada. Setkáváme se s nimi stále častěji a v neustále rozsáhlejších a rozmanitějších aplikacích. Predikce je jen jedna z mnoha možností, k čemu lze tento nástroj využívat, dalšími možnostmi může být například klasifikace dat a informací. Své uplatnění nacházejí nejen v autonomním řízení vozidel, strojovém učení, rozpoznávání a klasifikaci obrazu, v oblasti zvané big data a data mining, medicíně, ekonomice, při předpovědi počasí, v oborech robotiky, kybernetiky, ale v mnoha dalších oborech a disciplínách. A stejně tak ve všech oblastech souvisejících obecně se 4. průmyslovou revolucí.

Jak je uvedeno v rešeršní části této práce, konkrétně ve studiích a výzkumech [24], [33], [34], nebo [31], k predikci teploty v motorech letadel se v praxi nevyužívají pouze naměřené hodnoty teplot, ale také hodnoty stechiometrického poměru, dále hodnoty průtoku vzduchu, paliva, průtokového objemu, nebo otáčky turbíny.

Tyto hodnoty jsem však k dispozici neměl. Pro následující výzkum a zlepšení výsledků této studie by tedy bylo vhodné pro predikci teplot tekutiny v řezu TPL2 z řezu TPL1 tyto hodnoty dále uvažovat jako relevantní. Dalším přínosem by rovněž mohla být znalost vzájemné polohy a natočení senzorů v řezu TPL1 vůči senzorům v řezu TPL2, jelikož z některých studií uvedených v rešerši této diplomové práce plyne, že vzhledem k velikosti objemového průtoku v letadlových turbínových motorech je možno proudění propulzní látky uvažovat jako laminární “průtok tekutiny hadicí“. Dále bych rovněž doporučil zvážit použití měřicí techniky s vyšší vzorkovací frekvencí pro dosažení možnosti zachycení dynamiky změn propulzní látky motoru a “spárování“ konkrétních teplot senzorů z řezu druhého s teplotami senzorů v řezu prvním.

Dle tab. 6.1 bylo nejlepší volbou pro predikci použít model QNU, kdy její nejlepší predikce se od reálné naměřené teploty lišila pouze o 0,281 °C, predikovaný výsledek nejvzdálenější skutečné teplotě pak byl 11,978 °C. Druhou nejlepší sítí byla Extreme Learning Machine. Naopak nejméně přesným modelem byla neuronová síť Random Vector Function Link.

Seznam použité literatury

- [1] KAČENKA, Petr. Neuronové sítě [online]. [cit. 2019-08-07]. Dostupné z: <https://mks.mff.cuni.cz/library/NeuronoveSitePK/NeuronoveSitePK.pdf>
- [2] MOLČÍK, Filip. *Studie využití neuronových sítí a adaptivního monitorování pro algo-trading*. Praha, 2016. Diplomová práce. České vysoké učení technické v Praze.
- [3] ROUBAL, Michal. *Neuronové sítě a jejich aplikace*. Olomouc, 2012. Diplomová práce. Univerzita Palackého v Olomouci.
- [4] SMETANA, Ladislav. *Nelineární neuro-regulátor pro úlohy automatického řízení*. Praha, 2008. Diplomová práce. České vysoké učení technické v Praze.
- [5] HOLČÍK, Jiří, KOMENDA, Martin (eds.) a kol. *Matematická biologie: e-learningová učebnice* [online]. 1. vydání. Brno: Masarykova univerzita, 2015. ISBN 978-80-210-8095-9. Dostupné z: <http://portal.matematickabiologie.cz/res/f/neuronove-site-jednotlivy-neuron.pdf>
- [6] ŠTASTNÝ, Petr. *Rozpoznávání objektů pomocí neuronových sítí*. Brno, 2014. Diplomová práce. Masarykova univerzita.
- [7] MAŘÍK, Vladimír, Olga ŠTĚPÁNKOVÁ, Jiří LAŽANSKÝ a kol. *Umělá inteligence (1)*. Praha: Academia, 1993.
- [8] HOLČÍK, Jiří, KOMENDA, Martin (eds.) a kol. *Matematická biologie: e-learningová učebnice* [online]. 1. vydání. Brno: Masarykova univerzita, 2015. ISBN 978-80-210-8095-9. Dostupné z: <http://portal.matematickabiologie.cz/res/f/neuronove-site-perceptrony.pdf>
- [9] VALÁŠEK, Michael a kol. *Mechatronika*. Praha: ČVUT, 1995. ISBN 80-01-01276-X.
- [10] PEICHL, Adam. *Adaptační predikce korekčních faktorů válcovacích sil*. Praha, 2019. Diplomová práce. České vysoké učení technické v Praze.
- [11] NOVÁČEK, Tomáš. *Rekurentní neuronové sítě pro rozpoznávání řeči*. Brno, 2016. Diplomová práce. Vysoké učení technické v Brně.
- [12] POPELKA, Ondřej, Jan ŽIŽKA, Oldřich TRENZ a Jiří ŠTASTNÝ. *Umělá inteligence 2* [online]. [cit. 2019-08-08]. Dostupné z: <https://is.mendelu.cz/eknihovna/opory/index.pl?opora=2108>

- [13] REMEK, Branko, Petr HATSCHBACH a Jiří VÁVRA. *Experimentální metody a měření v dopravní technice*. 2. Praha: České vysoké učení technické v Praze, 2011.
- [14] KREIDL, Marcel. *Měření teploty - senzory a měřicí obvody*. Praha: BEN - technická literatura, 2005.
- [15] NOVÁK, Martin. *Technická měření*. 2. Praha: České vysoké učení technické v Praze, 2018.
- [16] HANUS, Daniel. Pohon letadel a letadlové motory. A0M36BEP – Přednáška 3. [online]. FEL ČVUT, Praha, 3. 3. 2014. [cit. 12/07/2019]. 104 str. Dostupné z: http://webdav.agents.fel.cvut.cz/data/teaching/bep/BEP3_2014.pdf
- [17] HANUS, Daniel. *Pohon letadel*. 2. Praha: České vysoké učení technické v Praze, 2003. ISBN 978-80-01-04104-8.
- [18] JERIE, Jan. *Teorie motorů: Teorie propulze a vnitřní aerodynamika turbínových motorů*. Praha: Ediční středisko Českého vysokého učení technického, 1981.
- [19] Ogaji, Stephen & Singh, Riti. (2003). Advanced engine diagnostics using artificial neural networks. *Applied Soft Computing*. 3. 259-271. 10.1016/S1568-4946(03)00038-3.
- [20] B. Shah, M. Sarvajith, B. Sankar and S. Thennavarajan, "Multi-Auto Associative Neural Network based sensor validation and estimation for aero-engine," *2013 IEEE AUTOTESTCON*, Schaumburg, IL, 2013, pp. 1-7. doi: 10.1109/AUTEST.2013.6645076
- [21] Özener Orkun, Yüksek Levent, Özkan Muammer. (2013). *Artificial neural network approach to predicting engine-out emissions and performance parameters of a turbo charged diesel engine*. *Thermal Science*. 17. 153-166. 10.2298/TSCII203212200.
- [22] Yildirim, Mustagime & Kurt, Bülent. (2018). Aircraft Gas Turbine Engine Health Monitoring System by Real Flight Data. *International Journal of Aerospace Engineering*. 2018. 1-12. 10.1155/2018/9570873.
- [23] Tolga Baklacioglu, Onder Turan, Hakan Aydin, Dynamic modeling of exergy efficiency of turboprop engine components using hybrid genetic algorithm-artificial neural networks, *Energy*, Volume 86, 2015, Pages 709-721, ISSN 0360-5442, <https://doi.org/10.1016/j.energy.2015.04.025>.
- [24] Torella, Giovanni & Gamma, Fausto & Palmesano, G. (2019). NEURAL NETWORKS FOR THE STUDY OF GAS TURBINE ENGINES AIR SYSTEM.

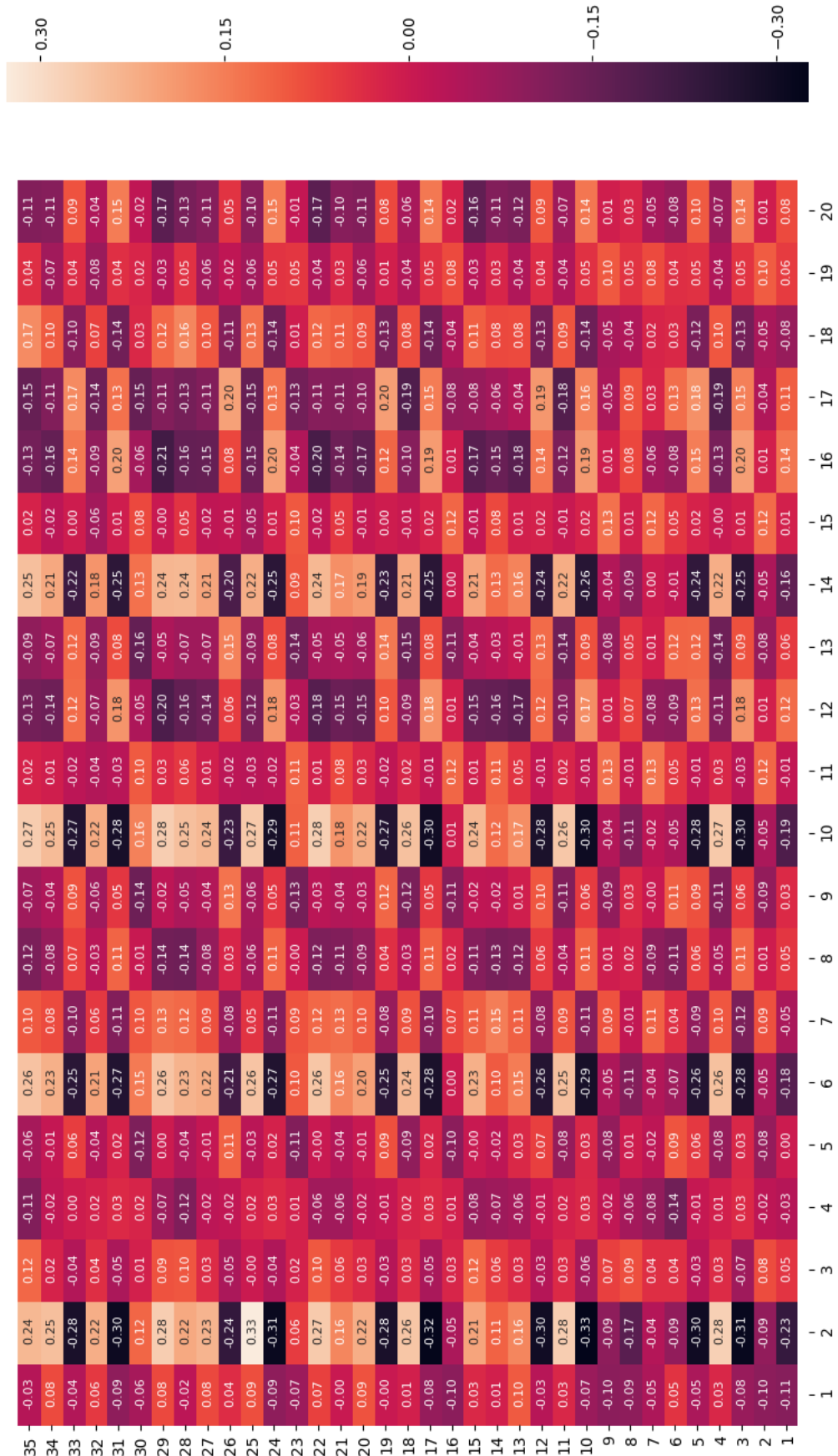
- Dostupné z:
https://www.researchgate.net/publication/228419970_NEURAL_NETWORKS_FOR_THE_STUDY_OF_GAS_TURBINE_ENGINES_AIR_SYSTEM
- [25] HOCKO, Marián a Katarína RATKOVSKÁ. Výpočtové cvičení z předmětu Plynové turbíny a turbokompresory [online]. Plzeň: Západočeská univerzita v Plzni, 2017 [cit. 2019-07-24]. Dostupné z: https://old.zcu.cz/export/sites/zcu/pracoviste/vyd/online/FST_Vypoctove-cviceni.pdf
- [26] Tracking Starts And Flights: Understanding Abbreviated Cycles and Full Cycles for PT6 Engine Maintenance. In: Covington Aircraft [online]. 2015 [cit. 2019-07-24]. Dostupné z: <http://blog.covingtonaircraft.com/2015/07/23/starts-and-flights/>
- [27] Shanxuan Tang, Hailong Tang, Min Chen, Transfer-learning based gas path analysis method for gas turbines, *Applied Thermal Engineering*, Volume 155, 2019, Pages 1-13, ISSN 1359-4311, <https://doi.org/10.1016/j.applthermaleng.2019.03.156>.
- [28] Kanelopoulos KK, Stamatis AA, Mathioudakis KK. Incorporating Neural Networks Into Gas Turbine Performance Diagnostics. ASME. *Turbo Expo: Power for Land, Sea, and Air, Volume 4: Manufacturing Materials and Metallurgy; Ceramics; Structures and Dynamics; Controls, Diagnostics and Instrumentation; Education; IGTI Scholar Award ()*:V004T15A011. doi:10.1115/97-GT-035.
- [29] Houman HANACHI, Jie LIU, Christopher MECHEFSKE, Multi-mode diagnosis of a gas turbine engine using an adaptive neuro-fuzzy system, *Chinese Journal of Aeronautics*, Volume 31, Issue 1, 2018, Pages 1-9, ISSN 1000-9361, <http://www.sciencedirect.com/science/article/pii/S1000936117302649>
- [30] Sanjay G. Barad, Ramaiah P.V., Giridhar R.K., Krishnaiah G., Neural network approach for a combined performance and mechanical health monitoring of a gas turbine engine, *Mechanical Systems and Signal Processing*, Volume 27, 2012, Pages 729-742, ISSN 0888-3270, <http://www.sciencedirect.com/science/article/pii/S0888327011003797>
- [31] Ilbas, Mustafa & Turkmen, Mahmut. (2012). Estimation of exhaust gas temperature using artificial neural network in turbofan engines. *Isi Bilimi Ve Teknigi Dergisi/ Journal of Thermal Science and Technology*. 32. 11-18.

- [32] Lu, Junjie & Huang, Jinquan & lu, Feng. (2019). Distributed Kernel Extreme Learning Machines for Aircraft Engine Failure Diagnostics. *Applied Sciences*. 9. 1707. 10.3390/app9081707.
- [33] Yazar, I., Yasa, T. and Kiyak, E. (2017), "Simulation-based steady-state aerothermal model for small-scale turboprop engine", *Aircraft Engineering and Aerospace Technology*, Vol. 89 No. 2, pp. 203-210. <https://doi.org/10.1108/AEAT-02-2015-0062>
- [34] N. Nariman-zadeh, K. Atashkari, A. Jamali, A. Pilechi & X. Yao (2005) Inverse modelling of multi-objective thermodynamically optimized turbojet engines using GMDH-type neural networks and evolutionary algorithms, *Engineering Optimization*, 37:5, 437-462, DOI: 10.1080/03052150500035591
- [35] MALÝ, Vladimír. *Metoda PCA a vícerozměrová analýza falešných sousedů pro posouzení neurčitosti redukovaného stavového vektoru provozních dat energetického zařízení*. Praha, 2010. Bakalářská práce.
- [36] KOLAŘÍK, Martin. *Hluboké učení pro klasifikaci textů* [online]. Brno, 2017 [cit. 2019-07-25]. Diplomová práce. Vysoké učení technické v Brně.
- [37] BUKOVSKÝ, Ivo. Higher Order Neurons and Supervised Learning for Prediction, Control, and Novelty Detection [online]. In: . Sendai, Japan, 25.8.2016, s. 49 [cit. 2019-08-03]. Dostupné z: https://studium.fs.cvut.cz/studium/u12110-3/UI/P8/Higher%20Order%20Neurons%20and%20Supervised%20Learning%20for%20Prediction,%20Novelty%20Detection,%20and%20Control_26.pdf
- [38] NOVÁČEK, Tomáš. *Intelligentní interakce robota s člověkem*. Brno, 2014. Bakalářská práce. Vysoké učení technické v Brně.

Přílohy

1. Obrázek teplotní mapy korelace senzorů z řezu TPL1 a TPL2 včetně korelačních koeficientů
2. Tabulka hodnot maximální chyby ($\max e$), střední kvadratické chyby (MSE) a směrodatné odchylky (σ) predikované teploty neuronovou sítí Multi Layer Perceptron pro jednotlivé senzory řezu TPL2 vůči teplotám reálným.
3. Tabulka hodnot maximální chyby ($\max e$), střední kvadratické chyby (MSE) a směrodatné odchylky (σ) predikované teploty neuronovou sítí *Extreme Learning Machine* pro jednotlivé senzory řezu TPL2 vůči teplotám reálným.
4. Tabulka hodnot maximální chyby ($\max e$), střední kvadratické chyby (MSE) a směrodatné odchylky (σ) predikované teploty neuronovou sítí *Random Vector Function Link* pro jednotlivé senzory řezu TPL2 vůči teplotám reálným.
5. Tabulka hodnot maximální chyby, střední kvadratické chyby (MSE) a směrodatné odchylky (σ) predikované teploty neuronovou sítí LNU pro jednotlivé senzory řezu TPL2 vůči teplotám reálným.
6. Tabulka hodnot maximální chyby, střední kvadratické chyby (MSE) a směrodatné odchylky (σ) predikované teploty neuronovou sítí QNU pro jednotlivé senzory řezu TPL2 vůči teplotám reálným.
7. Tabulka hodnot maximální chyby, střední kvadratické chyby (MSE) a směrodatné odchylky (σ) predikované teploty neuronovou sítí CNU pro jednotlivé senzory řezu TPL2 vůči teplotám reálným. Uvedena pouze jako ilustrativní příklad, nikoli chápána směrodatně.
8. CD se skripty

1. Obrázek teplotní mapy korelace senzorů z řezu TPL1 a TPL2 včetně korelačních koeficientů.



2. Tabulka hodnot maximální chyby ($max\ e$), střední kvadratické chyby (MSE) a směrodatné odchylky (σ) predikované teploty neuronovou sítí *Multi Layer Perceptron* pro jednotlivé senzory řezu TPL2 vůči teplotám reálným.

Senzor	Označení sen.	max e [°C]	MSE [°C]	σ [°C]
1.	TPL2A1	2,493	1,356	0,409
2.	TPL2B1	2,374	1,226	0,396
3.	TPL2C1	2,561	1,736	0,283
4.	TPL2D1	0,26	1,443	0,423
5.	TPL2E1	3,45	2,484	0,389
6.	TPL2F1	1,838	0,827	0,411
7.	TPL2G1	2,128	0,995	0,42
8.	TPL2A2	2,309	1,016	0,417
9.	TPL2B2	2,547	1,174	0,456
10.	TPL2C2	3,054	2,043	0,382
11.	TPL2D2	0,267	1,347	0,441
12.	TPL2E2	3,243	2,132	0,447
13.	TPL2F2	1,268	0,402	0,372
14.	TPL2G2	1,782	0,631	0,448
15.	TPL2A3	1,527	0,459	0,405
16.	TPL2B3	2,42	0,952	0,46
17.	TPL2C3	3,078	2,082	0,357
18.	TPL2D3	0,045	0,895	0,431
19.	TPL2E3	2,825	1,619	0,432
20.	TPL2F3	0,868	0,266	0,335
21.	TPL2G3	2,07	0,543	0,491
22.	TPL2A4	1,109	0,339	0,433
23.	TPL2B4	1,871	0,658	0,443
24.	TPL2C4	2,834	1,986	0,317
25.	TPL2D4	1,916	1,947	0,028
26.	TPL2E4	1,954	1,179	0,283
27.	TPL2F4	1,148	0,274	0,337
28.	TPL2G4	1,946	0,462	0,43
29.	TPL2A5	0,524	0,384	0,398
30.	TPL2B5	1,454	0,562	0,391
31.	TPL2C5	2,811	1,904	0,361
32.	TPL2D5	0,891	0,347	0,383
33.	TPL2E5	10,014	8,853	0,488
34.	TPL2F5	1,449	0,262	0,29
35.	TPL2G5	1,415	0,455	0,396

3. Tabulka hodnot maximální chyby ($max e$), střední kvadratické chyby (MSE) a směrodatné odchyly (σ) predikované teploty neuronovou sítí *Extrame Learning Machine* pro jednotlivé senzory řezu TPL2 vůči teplotám reálným.

Senzor	Označení sen.	max e [°C]	MSE [°C]	σ [°C]
1.	TPL2A1	2,183	0,6	0,751
2.	TPL2B1	2,019	0,703	0,819
3.	TPL2C1	2,697	0,695	0,791
4.	TPL2D1	0,131	2,479	0,802
5.	TPL2E1	3,171	1,21	0,795
6.	TPL2F1	1,788	0,785	0,791
7.	TPL2G1	1,483	0,608	0,667
8.	TPL2A2	1,625	0,621	0,705
9.	TPL2B2	1,879	0,695	0,823
10.	TPL2C2	3,121	0,858	0,83
11.	TPL2D2	0,086	2,394	0,775
12.	TPL2E2	2,854	0,894	0,781
13.	TPL2F2	1,118	1,093	0,746
14.	TPL2G2	0,989	0,818	0,687
15.	TPL2A3	0,713	0,979	0,676
16.	TPL2B3	1,693	0,684	0,804
17.	TPL2C3	3,112	0,896	0,766
18.	TPL2D3	0,35	1,881	0,716
19.	TPL2E3	2,542	0,686	0,772
20.	TPL2F3	0,69	1,281	0,689
21.	TPL2G3	1,186	0,926	0,725
22.	TPL2A4	0,477	1,251	0,689
23.	TPL2B4	1,471	0,692	0,762
24.	TPL2C4	2,9	0,897	0,739
25.	TPL2D4	2,622	4,932	2,6
26.	TPL2E4	1,588	0,571	0,689
27.	TPL2F4	0,69	1,117	0,648
28.	TPL2G4	1,337	0,86	0,713
29.	TPL2A5	0,43	1,415	0,68
30.	TPL2B5	1,183	0,685	0,688
31.	TPL2C5	2,632	0,91	0,715
32.	TPL2D5	0,529	1,203	0,654
33.	TPL2E5	12,246	6,862	1,953
34.	TPL2F5	0,641	1,066	0,623
35.	TPL2G5	1,25	0,83	0,722

4. Tabulka hodnot maximální chyby ($max e$), střední kvadratické chyby (MSE) a směrodatné odchylky (σ) predikované teploty neuronovou sítí *Random Vector Function Link* pro jednotlivé senzory řezu TPL2 vůči teplotám reálným.

Sensor	Označení sen.	max e [°C]	MSE [°C]	σ [°C]
1.	TPL2A1	2,954	0,779	0,962
2.	TPL2B1	2,992	0,798	0,988
3.	TPL2C1	3,784	0,951	1,025
4.	TPL2D1	0,19	2,416	0,925
5.	TPL2E1	4,678	1,514	1,003
6.	TPL2F1	2,327	0,854	0,939
7.	TPL2G1	2,515	0,806	0,908
8.	TPL2A2	2,462	0,678	0,839
9.	TPL2B2	3,19	0,765	0,955
10.	TPL2C2	4,043	1,116	0,982
11.	TPL2D2	0,151	2,311	0,881
12.	TPL2E2	4,296	1,142	0,949
13.	TPL2F2	1,898	1,184	0,992
14.	TPL2G2	2,381	1,005	0,932
15.	TPL2A3	1,837	0,993	0,877
16.	TPL2B3	3,176	0,748	0,93
17.	TPL2C3	4,1	1,159	0,919
18.	TPL2D3	0,774	1,766	0,97
19.	TPL2E3	3,588	0,841	0,912
20.	TPL2F3	1,305	1,352	0,921
21.	TPL2G3	1,956	1,041	0,849
22.	TPL2A4	1,329	1,271	0,832
23.	TPL2B4	2,612	0,771	0,909
24.	TPL2C4	3,746	1,133	0,874
25.	TPL2D4	4,917	5,573	3,331
26.	TPL2E4	2,994	0,658	0,824
27.	TPL2F4	1,568	1,15	0,787
28.	TPL2G4	2,182	0,935	0,815
29.	TPL2A5	1,462	1,467	0,925
30.	TPL2B5	1,944	0,753	0,822
31.	TPL2C5	3,301	1,118	0,802
32.	TPL2D5	1,762	1,202	0,883
33.	TPL2E5	11,735	7,531	1,209
34.	TPL2F5	1,864	1,218	1,024
35.	TPL2G5	2,144	0,9	0,857

5. Tabulka hodnot maximální chyby, střední kvadratické chyby (*MSE*) a směrodatné odchylky (σ) predikované teploty neuronovou sítí LNU pro jednotlivé senzory řezu TPL2 vůči teplotám reálným.

Sensor	Označení sen.	max e [°C]	MSE [°C]	σ [°C]
1.	TPL2A1	2,183	0,6	0,751
2.	TPL2B1	2,019	0,703	0,819
3.	TPL2C1	2,697	0,695	0,791
4.	TPL2D1	4,424	2,479	0,802
5.	TPL2E1	3,171	1,21	0,795
6.	TPL2F1	2,351	0,785	0,791
7.	TPL2G1	1,805	0,608	0,667
8.	TPL2A2	2,358	0,621	0,705
9.	TPL2B2	2,107	0,695	0,823
10.	TPL2C2	3,121	0,858	0,83
11.	TPL2D2	4,401	2,394	0,775
12.	TPL2E2	2,854	0,894	0,781
13.	TPL2F2	2,916	1,093	0,746
14.	TPL2G2	2,465	0,818	0,687
15.	TPL2A3	3,035	0,979	0,676
16.	TPL2B3	2,009	0,684	0,804
17.	TPL2C3	3,112	0,896	0,766
18.	TPL2D3	3,69	1,881	0,716
19.	TPL2E3	2,542	0,686	0,772
20.	TPL2F3	2,945	1,281	0,689
21.	TPL2G3	2,946	0,926	0,725
22.	TPL2A4	2,996	1,251	0,689
23.	TPL2B4	2,289	0,692	0,762
24.	TPL2C4	2,9	0,897	0,739
25.	TPL2D4	10,985	4,932	2,6
26.	TPL2E4	1,813	0,571	0,689
27.	TPL2F4	2,888	1,117	0,648
28.	TPL2G4	2,826	0,86	0,713
29.	TPL2A5	3,083	1,415	0,68
30.	TPL2B5	2,338	0,685	0,688
31.	TPL2C5	2,632	0,91	0,715
32.	TPL2D5	2,99	1,203	0,654
33.	TPL2E5	12,246	6,862	1,953
34.	TPL2F5	2,716	1,066	0,623
35.	TPL2G5	2,517	0,83	0,722

6. Tabulka hodnot maximální chyby, střední kvadratické chyby (*MSE*) a směrodatné odchylky (σ) predikované teploty neuronovou sítí QNU pro jednotlivé senzory řezu TPL2 vůči teplotám reálným.

Sensor	Označení sen.	max e [°C]	MSE [°C]	σ [°C]
1.	TPL2A1	2,007	0,678	0,559
2.	TPL2B1	1,89	0,626	0,512
3.	TPL2C1	2,407	0,945	0,591
4.	TPL2D1	2,468	0,784	0,575
5.	TPL2E1	3,351	1,354	0,749
6.	TPL2F1	1,841	0,468	0,498
7.	TPL2G1	1,605	0,659	0,372
8.	TPL2A2	1,545	0,465	0,481
9.	TPL2B2	2,108	0,633	0,553
10.	TPL2C2	2,846	1,091	0,678
11.	TPL2D2	2,28	0,782	0,578
12.	TPL2E2	3,127	1,113	0,718
13.	TPL2F2	1,054	0,313	0,38
14.	TPL2G2	1,388	0,449	0,4
15.	TPL2A3	1,33	0,34	0,41
16.	TPL2B3	2,116	0,587	0,557
17.	TPL2C3	2,898	1,09	0,657
18.	TPL2D3	1,85	0,55	0,503
19.	TPL2E3	2,861	0,842	0,651
20.	TPL2F3	0,911	0,266	0,319
21.	TPL2G3	1,637	0,396	0,465
22.	TPL2A4	1,372	0,449	0,428
23.	TPL2B4	1,73	0,509	0,519
24.	TPL2C4	2,71	1,035	0,659
25.	TPL2D4	0,281	0,1	0,056
26.	TPL2E4	1,889	0,612	0,459
27.	TPL2F4	0,945	0,26	0,315
28.	TPL2G4	1,707	0,35	0,435
29.	TPL2A5	1,794	0,449	0,398
30.	TPL2B5	1,491	0,412	0,442
31.	TPL2C5	2,5	1,037	0,658
32.	TPL2D5	1,165	0,322	0,368
33.	TPL2E5	11,978	4,727	2,355
34.	TPL2F5	1,065	0,216	0,276
35.	TPL2G5	1,527	0,373	0,435

7. Tabulka hodnot maximální chyby, střední kvadratické chyby (*MSE*) a směrodatné odchylky (σ) predikované teploty neuronovou sítí CNU pro jednotlivé senzory řezu TPL2 vůči teplotám reálným. Uvedena pouze jako ilustrativní příklad, nikoli chápána směrodatně.

Sensor	Označení sen.	max e [°C]	MSE [°C]	σ [°C]
1.	TPL2A1	123723,397	13773,634	22549,572
2.	TPL2B1	113917,738	12681,997	20762,305
3.	TPL2C1	102750,31	11438,776	18727,067
4.	TPL2D1	104442,553	11627,157	19035,114
5.	TPL2E1	84057,515	9357,794	15320,286
6.	TPL2F1	103245,997	11493,973	18817,305
7.	TPL2G1	137732,171	15333,274	25102,695
8.	TPL2A2	114773,126	12777,239	20918,281
9.	TPL2B2	104275,251	11608,54	19004,914
10.	TPL2C2	93239,491	10379,993	16993,725
11.	TPL2D2	101117,018	11256,932	18429,011
12.	TPL2E2	84516,741	9408,947	15403,952
13.	TPL2F2	106074,898	11808,891	19332,802
14.	TPL2G2	96300,734	10720,888	17551,512
15.	TPL2A3	94744,523	10547,55	17267,878
16.	TPL2B3	115857,645	12897,936	21115,866
17.	TPL2C3	88055,107	9802,87	16048,862
18.	TPL2D3	89410,55	9953,733	16295,544
19.	TPL2E3	85208,822	9485,976	15530,003
20.	TPL2F3	108897,487	12123,156	19847,235
21.	TPL2G3	121887,268	13569,301	22214,753
22.	TPL2A4	100882,403	11230,863	18386,5
23.	TPL2B4	91811,611	10220,997	16733,319
24.	TPL2C4	121194,091	13492,107	22088,624
25.	TPL2D4	92501,463	10297,859	16858,988
26.	TPL2E4	89687,206	9984,562	16346,191
27.	TPL2F4	96669,737	10761,889	17618,67
28.	TPL2G4	141683,131	15773,088	25822,687
29.	TPL2A5	102499,303	11410,861	18681,12
30.	TPL2B5	89503,645	9964,062	16312,652
31.	TPL2C5	107649,615	11984,284	19620,061
32.	TPL2D5	91090,176	10140,721	16601,755
33.	TPL2E5	74395,8	8282,253	13560,079
34.	TPL2F5	117339,929	13063,028	21385,948
35.	TPL2G5	114801,627	12780,444	20923,361

8. CD se skripty