



# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY

A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV BIOMEDICÍNSKÉHO INŽENÝRSTVÍ

DEPARTMENT OF BIOMEDICAL ENGINEERING

## METODY INICIALIZACE VAH NEURONOVÝCH SÍTÍ A JEJICH VLIV NA UČENÍ SÍTÍ

METHODS FOR INITIALIZING NEURAL NETWORK WEIGHTS AND THEIR EFFECT ON NETWORK LEARNING

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

Jakub Prukner

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Jiří Chmelík, Ph.D.

BRNO 2024

# Bakalářská práce

bakalářský studijní program **Biomedicínská technika a bioinformatika**

Ústav biomedicínského inženýrství

**Student:** Jakub Prukner

**ID:** 240554

**Ročník:** 3

**Akademický rok:** 2023/24

**NÁZEV TÉMATU:**

## Metody inicializace vah neuronových sítí a jejich vliv na učení sítí

**POKyny PRO VYPRACOVÁNÍ:**

1) Nastudujte problematiku umělých neuronových sítí a jejich učení a proveďte rešerši metod používaných pro inicializaci vah. 2) Nalezněte vhodnou veřejně dostupnou učební databázi, formulujte problém a navrhnete experiment pro analýzu vlivu inicializace vah na učení sítě. 3) Navrhnete a implementujete základní neuronovou síť a proveďte její učení na zvolené databázi. 4) Realizujte experiment a otestujte alespoň tři různé metody inicializace vah. 5) Navrhnete vhodné metriky pro vyhodnocení úspěšnosti učení sítě. Vypočtete úspěšnosti naučených sítí. 6) Dosažené výsledky statisticky vyhodnotte a získané výsledky diskutujte.

**DOPORUČENÁ LITERATURA:**

[1] NARKHEDE, Meenal V.; BARTAKKE, Prashant P. a SUTAONE, Mukul S., 2022. A review on weight initialization strategies for neural networks. Online. Artificial Intelligence Review. Roč. 55, č. 1, s. 291-322. ISSN 0269-2821. Dostupné z: <https://doi.org/10.1007/s10462-021-10033-z>. [cit. 2023-08-14].

[2] ESGUERRA, Kiara; NASIR, Muneeb; TANG, Tong Boon; TUMIAN, Afidalina a HO, Eric Tatt Wei, 2023. Sparsity-Aware Orthogonal Initialization of Deep Neural Networks. Online. IEEE Access. Roč. 11, s. 74165-74181. ISSN 2169-3536. Dostupné z: <https://doi.org/10.1109/ACCESS.2023.3295344>. [cit. 2023-08-14].

**Termín zadání:** 5.2.2024

**Termín odevzdání:** 29.5.2024

**Vedoucí práce:** Ing. Jiří Chmelík, Ph.D.

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

**UPOZORNĚNÍ:**

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

## **ABSTRAKT**

Práce se zabývá použitím různých metod inicializace vah umělých neuronových sítí a sledováním jejich vlivu na učení sítě. Jako úloha pro síť je zvolena klasifikace obrázků ze dvou databází MNIST a CIFAR-10. Teoretická část popisuje oblast umělých neuronových sítí a také různé metody inicializace vah. Praktická část obsahuje popis a implementaci experimentů, popis architektur a jejich hyperparametrů. Jednotlivé experimenty sledují vliv vybraných metod a jejich případných konfigurací na učení různých architektur umělých neuronových sítí. Výsledky jsou pro jednotlivé databáze a typy architektur porovnány a jsou vybrány metody, se kterými síť dosáhla nejlepšího naučení. Dále jsou vybrány metody, se kterými bylo dosaženo optimálního naučení sítě nejrychleji. Dosažené výsledky jsou diskutovány.

## **KLÍČOVÁ SLOVA**

Neuronové sítě, Inicializace vah, LSUV, MNIST, CIFAR-10

## **ABSTRACT**

This thesis examines the use of various methods for initialising the weights of artificial neural networks and monitoring their impact on network learning. Image classification from two databases, MNIST and CIFAR-10, is selected as the task for the network. The theoretical section provides an overview of the field of artificial neural networks, along with an analysis of different methods for initialising weights. The practical section includes a description of the experiments conducted, an explanation of the architectures and their associated hyperparameters. The individual experiments observe the effect of the selected methods and their respective configurations on the learning of different artificial neural network architectures. The results are compared for each dataset and architecture type, and the methods with which a the network achieved the best learning are selected. Furthermore, the methods with which the optimal learning of the network was achieved the fastest are selected. The results obtained are discussed.

## **KEYWORDS**

Neural networks, weight initialisation, LSUV, MNIST, CIFAR-10

PRUKNER, Jakub. *Metody inicializace vah neuronových sítí a jejich vliv na učení sítí*.  
Bakalářská práce. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komu-  
nikačních technologií, Ústav biomedicínského inženýrství, 2024. Vedoucí práce: Ing. Jiří  
Chmelík, Ph.D.

# Prohlášení autora o původnosti díla

**Jméno a příjmení autora:** Jakub Prukner  
**VUT ID autora:** 240554  
**Typ práce:** Bakalářská práce  
**Akademický rok:** 2023/24  
**Téma závěrečné práce:** Metody inicializace vah neuronových sítí a jejich vliv na učení sítí

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

Jako autor uvedené závěrečné práce dále prohlašuji, že v souvislosti s vytvořením této závěrečné práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

Brno .....

.....  
podpis autora\*

---

\*Autor podepisuje pouze v tištěné verzi.

## PODĚKOVÁNÍ

Rád bych poděkoval vedoucímu bakalářské práce panu Ing. Jiřímu Chmelíkovi, Ph.D. za odborné vedení, konzultace, trpělivost a podnětné návrhy k práci, které mi velice pomohli při psaní této bakalářské práce. Dále bych rád poděkoval své rodině, která mě podporuje po celou dobu mého studia.

# Obsah

Úvod	16
<b>1 Teoretický úvod</b>	<b>17</b>
1.1 Strojové učení	17
<b>2 Umělé neuronové sítě</b>	<b>18</b>
2.1 Historie	18
2.2 Neuron	18
2.3 Jednovrstvá dopředná síť	18
2.4 Aktivační funkce	20
2.5 Chybové funkce	21
2.6 Optimalizační algoritmy	21
2.7 Vícevrstvá dopředná síť	22
2.8 Konvoluční neuronová síť	25
<b>3 Inicializace vah</b>	<b>26</b>
3.1 Metody náhodné inicializace vah	27
3.1.1 Metody inicializace vah z intervalu	27
3.1.2 Metody inicializaci založené na škálování rozptylu	28
3.1.3 Ostatní metody náhodné inicializace vah	30
3.2 Metody inicializace vycházející z informací o trénovacích datech	30
3.3 Hybridní metody inicializace	30
3.4 Transfer learning	31
<b>4 Metriky hodnocení učení</b>	<b>32</b>
4.1 Metriky hodnocení klasifikace do více tříd	32
4.2 Počet epoch	34
<b>5 Formulace problému</b>	<b>35</b>
<b>6 Použité databáze</b>	<b>36</b>
6.1 Databáze MNIST	36
6.2 CIFAR-10	37
<b>7 Praktická část</b>	<b>38</b>
7.1 Navrhnutí experimentů	38
7.2 Předzpracování datasetu	39
7.3 Použité neuronové sítě	39
7.3.1 Popis architektur sítí	39

7.3.2	Hyperparametry sítí . . . . .	43
7.4	Implementace jednotlivých experimentů . . . . .	43
7.5	Statistické vyhodnocení výsledků . . . . .	44
<b>8</b>	<b>Výsledky a diskuse</b>	<b>46</b>
8.1	Výsledky – databáze MNIST . . . . .	46
8.1.1	Mělká vícevrstvá perceptronovská síť . . . . .	46
8.1.2	Hluboká vícevrstvá perceptronovská síť . . . . .	50
8.1.3	Jednoduchá konvoluční neuronová síť . . . . .	52
8.1.4	Hluboká konvoluční neuronová síť – ResNet18 . . . . .	55
8.2	Výsledky – databáze CIFAR–10 . . . . .	57
8.2.1	Mělká vícevrstvá perceptronovská síť . . . . .	57
8.2.2	Hluboká vícevrstvá perceptronovská síť . . . . .	59
8.2.3	Jednoduchá konvoluční neuronová síť . . . . .	61
8.2.4	Hluboká konvoluční neuronová síť – ResNet18 . . . . .	64
8.3	Srovnání a diskuse výsledků . . . . .	66
	<b>Závěr</b>	<b>73</b>
	<b>Literatura</b>	<b>74</b>
	<b>Seznam symbolů a zkratk</b>	<b>80</b>
<b>A</b>	<b>Výsledky pro databázi MNIST</b>	<b>81</b>
A.1	Mělká vícevrstvá perceptronovská síť . . . . .	81
A.1.1	LSUV . . . . .	81
A.1.2	Náhodná inicializace vah z normálního a rovnoměrného prav- děpodobnostního rozdělení . . . . .	82
A.1.3	Ostatní metody náhodné inicializace . . . . .	86
A.2	Hluboká vícevrstvá perceptronovská síť . . . . .	88
A.2.1	LSUV . . . . .	88
A.2.2	Náhodná inicializace vah z normálního a rovnoměrného prav- děpodobnostního rozdělení . . . . .	90
A.2.3	Ostatní metody náhodné inicializace . . . . .	94
A.3	Jednoduchá konvoluční neuronová síť . . . . .	96
A.3.1	Náhodná inicializace vah z normálního a rovnoměrného prav- děpodobnostního rozdělení . . . . .	98
A.3.2	Ostatní metody náhodné inicializace . . . . .	102
A.4	Hluboká konvoluční neuronová síť – ResNet18 . . . . .	104
A.4.1	LSUV . . . . .	104



A.4.2	Náhodná inicializace vah z normálního a rovnoměrného pravděpodobnostního rozdělení . . . . .	106
A.4.3	Ostatní metody náhodné inicializace . . . . .	110
<b>B</b>	<b>Výsledky pro databázi CIFAR-10</b>	<b>112</b>
B.1	Mělká vícevrstvá perceptronová síť . . . . .	112
B.1.1	LSUV . . . . .	112
B.1.2	Náhodná inicializace vah z normálního a rovnoměrného pravděpodobnostního rozdělení . . . . .	113
B.1.3	Ostatní metody náhodné inicializace . . . . .	117
B.2	Hluboká vícevrstvá perceptronová síť . . . . .	119
B.2.1	LSUV . . . . .	119
B.2.2	Náhodná inicializace vah z normálního a rovnoměrného pravděpodobnostního rozdělení . . . . .	121
B.2.3	Ostatní metody náhodné inicializace . . . . .	125
B.2.4	Jednoduchá konvoluční neuronová síť . . . . .	127
B.2.5	Hluboká konvoluční neuronová síť – ResNet18 . . . . .	135
<b>C</b>	<b>Obsah elektronické přílohy</b>	<b>143</b>

# Seznam obrázků

2.1	Obrázek umělého neuronu . . . . .	19
2.2	Ilustrační obrázek architektury CNN. . . . .	25
3.1	Vizualizace sedlového bodu chybové funkce . . . . .	26
4.1	Obrázek matice záměn pro klasifikaci do dvou tříd. . . . .	32
4.2	Obrázek ilustrující získání hodnot TP, FN, TN a FP pro klasifikaci do více tříd. . . . .	33
4.3	Příklad matice záměn, pro 4 různé třídy . . . . .	34
6.1	Ukázka šedotónových obrázků z databáze MNIST. . . . .	36
6.2	Ukázka obrázků z databáze CIFAR-10 . . . . .	37
7.1	Obrázek architektury navrhnuté konvoluční neuronové sítě. . . . .	41
7.2	Obrázek architektury ResNet18. . . . .	42
8.1	Matice záměn získaná pro testovací sadu z databáze MNIST po inicializaci mělké vícevrstvé perceptronovské sítě nulami. . . . .	46
8.2	Průběhy správností a chyb na validační a trénovací sadě pro vícevrstvou perceptronovskou síť inicializovanou nulami (databáze MNIST). . . . .	47
A.1	Průběhy chyb a správností pro mělkou vícevrstvou perceptronovskou síť využívající metodu LSUV–validační a trénovací sada MNIST . . . . .	82
A.2	Průběhy chyb a správností pro mělkou vícevrstvou perceptronovskou síť využívající metodu náhodná normální a rovnoměrné–validační a trénovací sada MNIST . . . . .	84
A.3	Průběhy chyb a správností pro mělkou vícevrstvou perceptronovskou síť využívající metody ostatní náhodné–validační a trénovací sada MNIST . . . . .	86
A.4	Průběhy chyb a správností pro hlubokou vícevrstvou perceptronovskou síť využívající metodu LSUV–validační a trénovací sada MNIST . . . . .	89
A.5	Průběhy chyb a správností pro hlubokou vícevrstvou perceptronovskou síť využívající metodu náhodné normální a rovnoměrné–validační a trénovací sada MNIST . . . . .	91
A.6	Průběhy chyb a správností pro hlubokou vícevrstvou perceptronovskou síť využívající metody náhodné ostatní–validační a trénovací sada MNIST . . . . .	94
A.7	Průběhy chyb a správností pro jednoduchou CNN využívající metodu LSUV–validační a trénovací sada MNIST . . . . .	97
A.8	Průběhy chyb a správností pro jednoduchou CNN využívající metodu náhodná normální a rovnoměrné–validační a trénovací sada MNIST . . . . .	99
A.9	Průběhy chyby a správností pro jednoduchou CNN využívající metody náhodné ostatní–validační a trénovací sada MNIST . . . . .	102

A.10 Průběhy chyb a správností pro ResNet18 využívající metodu LSUV– validační a trénovací sada MNIST . . . . .	105
A.11 Průběhy chyb a správností pro ResNet18 využívající metodu náhodné rovnoměrné a normální–validační a trénovací sada MNIST . . . . .	107
A.12 Průběhy chyb a správností pro ResNet18 využívající metody náhodné ostatní–validační a trénovací sada MNIST . . . . .	110
B.1 Průběhy chyb a správností pro mělkou vícevrstvou perceptronovskou sít využívající metodu LSUV–validační a trénovací sada CIFAR–10 . . . . .	113
B.2 Průběhy chyb a správností pro mělkou vícevrstvou perceptronovskou sít využívající metodu náhodné normální a rovnoměrné–validační a trénovací sada CIFAR–10 . . . . .	115
B.3 Průběhy chyb a správností pro mělkou vícevrstvou perceptronovskou sít využívající metody ostatní náhodné–validační a trénovací sada CIFAR–10 . . . . .	117
B.4 Průběhy chyb a správností pro hlubokou vícevrstvou perceptronov- skou sít využívající metodu LSUV–validační a trénovací sada CIFAR–10119	
B.5 Průběhy chyb a správností pro hlubokou vícevrstvou perceptronov- skou sít využívající metodu náhodná normální–validační a trénovací sada CIFAR–10 . . . . .	122
B.6 Průběhy chyb a správností pro hlubokou vícevrstvou perceptronov- skou sít využívající metody ostatní náhodné–trénovací a validační sada CIFAR–10 . . . . .	125
B.7 Průběhy chyb a správností pro jednoduchou CNN využívající metodu LSUV–validační a trénovací sada CIFAR–10 . . . . .	128
B.8 Průběhy chyb a správností pro jednoduchou CNN využívající metodu náhodné normální a rovnoměrné–validační a trénovací sada CIFAR–10	130
B.9 Průběhy chyb a správností pro jednoduchou CNN využívající metody náhodné ostatní–validační a trénovací sada CIFAR–10 . . . . .	133
B.10 Průběhy chyb a správností pro ResNet18 využívající metodu LSUV– validační a trénovací sada CIFAR–10 . . . . .	136
B.11 Průběhy chyb a správností pro ResNet18 využívající metodu náhodné normální a rovnoměrné–validační a trénovací sada CIFAR–10 . . . . .	138
B.12 Průběhy chyb a správností pro ResNet18 využívající metody ostatní náhodné–validační a trénovací sada CIFAR–10 . . . . .	141

# Seznam tabulek

7.1	Popis jednotlivých vrstev v navrhnuté mělké vícevrstvé perceptronovské síti . . . . .	39
7.2	Popis jednotlivých vrstev v navrhnuté hluboké vícevrstvé perceptronovské síti . . . . .	40
7.3	Popis architektury navrhnuté konvoluční neuronové sítě. . . . .	40
7.4	Popis počtu filtrů konvolučních vrstev v jednotlivých blocích ResNet18	42
8.1	Výsledky pro vícevrstvou perceptronovskou síť inicializovanou nulami (databáze MNIST). . . . .	47
8.2	Shrnutí nejlepších výsledků správnosti na trénovací sadě všech architektur pro databázi MNIST. . . . .	67
8.3	Shrnutí nejlepších výsledků správnosti na trénovací sadě všech architektur pro databázi CIFAR-10. . . . .	68
8.4	Shrnutí nejrychlejších optimálních naučení všech architektur pro databázi MNIST. . . . .	68
8.5	Shrnutí nejrychlejších optimálních naučení všech architektur pro databázi CIFAR-10. . . . .	69
8.6	Popis nejlepších výsledků všech architektur v počtu epoch k dosažení určité správnosti na databázi MNIST. . . . .	69
8.7	Popis nejlepších výsledků všech architektur v počtu epoch k dosažení určité správnosti na databázi CIFAR-10. . . . .	70
A.1	Výsledky pro mělkou vícevrstvou perceptronovskou síť využívající metodu LSUV-testovací sada MNIST . . . . .	81
A.2	Výsledky pro mělkou vícevrstvou perceptronovskou síť využívající metodu LSUV-validační sada MNIST . . . . .	81
A.3	Výsledky pro mělkou vícevrstvou perceptronovskou síť využívající metodu náhodná normální-testovací sada MNIST . . . . .	83
A.4	Výsledky pro mělkou vícevrstvou perceptronovskou síť využívající metodu náhodná normální-validační sada MNIST . . . . .	83
A.5	Výsledky pro mělkou vícevrstvou perceptronovskou síť využívající metodu náhodná rovnoměrné-testovací sada MNIST . . . . .	85
A.6	Výsledky pro mělkou vícevrstvou perceptronovskou síť využívající metodu náhodná rovnoměrné-validační sada MNIST . . . . .	85
A.7	Výsledky pro mělkou vícevrstvou perceptronovskou síť využívající metody ostatní náhodné-testovací sada MNIST . . . . .	87
A.8	Průběhy chyb a správností pro mělkou vícevrstvou perceptronovskou síť využívající metody ostatní náhodné-validační sada MNIST . . . . .	87

A.9	Výsledky pro hlubokou vícevrstvou perceptronovskou síť využívající metodu LSUV–testovací sada MNIST . . . . .	88
A.10	Výsledky pro hlubokou vícevrstvou perceptronovskou síť využívající metodu LSUV–validační sada MNIST . . . . .	89
A.11	Výsledky pro hlubokou vícevrstvou perceptronovskou síť využívající metodu náhodné normální–testovací sada MNIST . . . . .	90
A.12	Výsledky pro hlubokou vícevrstvou perceptronovskou síť využívající metodu náhodné normální–validační sada MNIST . . . . .	92
A.13	Výsledky pro hlubokou vícevrstvou perceptronovskou síť využívající metodu náhodné rovnoměrné–testovací sada MNIST . . . . .	93
A.14	Výsledky pro hlubokou vícevrstvou perceptronovskou síť využívající metodu náhodné rovnoměrné–validační sada MNIST . . . . .	93
A.15	Výsledky pro hlubokou vícevrstvou perceptronovskou síť využívající metody náhodné ostatní–testovací sada MNIST . . . . .	95
A.16	Výsledky pro hlubokou vícevrstvou perceptronovskou síť využívající metody náhodné ostatní–validační sada MNIST . . . . .	95
A.17	Výsledky pro jednoduchou CNN využívající metodu LSUV–testovací sada MNIST . . . . .	96
A.18	Výsledky pro jednoduchou CNN využívající metodu LSUV–validační sada MNIST . . . . .	97
A.19	Výsledky pro jednoduchou CNN využívající metodu náhodná normální–testovací sada MNIST . . . . .	98
A.20	Výsledky pro jednoduchou CNN využívající metodu náhodná normální–validační sada MNIST . . . . .	100
A.21	Výsledky pro jednoduchou CNN využívající metodu náhodná rovnoměrná–testovací sada MNIST . . . . .	100
A.22	Výsledky pro jednoduchou CNN využívající metodu náhodná rovnoměrná–validační sada MNIST . . . . .	101
A.23	Výsledky pro jednoduchou CNN využívající metody náhodné ostatní–testovací sada MNIST . . . . .	103
A.24	Výsledky pro jednoduchou CNN využívající metody náhodné ostatní–validační sada MNIST . . . . .	103
A.25	Výsledky pro ResNet18 využívající metodu LSUV–testovací sada MNIST	104
A.26	Výsledky pro ResNet18 využívající metodu LSUV–validační sada MNIST	105
A.27	Výsledky pro ResNet18 využívající metodu náhodné normální–testovací sada MNIST . . . . .	106
A.28	Výsledky pro ResNet18 využívající metodu náhodné normální–validační sada MNIST . . . . .	108

A.29	Výsledky pro ResNet18 využívající metodu náhodné rovnoměrné– testovací sada MNIST . . . . .	109
A.30	Výsledky pro ResNet18 využívající metodu náhodné rovnoměrné– validační sada MNIST . . . . .	109
A.31	Průběhy chyb a správností pro ResNet18 využívající metody náhodné ostatní–testovací sada MNIST . . . . .	111
A.32	Průběhy chyb a správností pro ResNet18 využívající metody náhodné ostatní–validační sada MNIST . . . . .	111
B.1	Výsledky pro mělkou vícevrstvou perceptronovskou síť využívající me- todu LSUV–testovací sada CIFAR–10 . . . . .	112
B.2	Výsledky pro mělkou vícevrstvou perceptronovskou síť využívající me- todu LSUV–validační sada CIFAR–10 . . . . .	112
B.3	Výsledky pro mělkou vícevrstvou perceptronovskou síť využívající me- todu náhodné normální–testovací sada CIFAR–10 . . . . .	114
B.4	Výsledky pro mělkou vícevrstvou perceptronovskou síť využívající me- todu náhodné normální–validační sada CIFAR–10 . . . . .	114
B.5	Výsledky pro mělkou vícevrstvou perceptronovskou síť využívající me- todu náhodné rovnoměrné–testovací sada CIFAR–10 . . . . .	116
B.6	Výsledky pro mělkou vícevrstvou perceptronovskou síť využívající me- todu náhodné rovnoměrné–validační sada CIFAR–10 . . . . .	116
B.7	Výsledky pro mělkou vícevrstvou perceptronovskou síť využívající me- todu ostatní náhodné–testovací sada CIFAR–10 . . . . .	118
B.8	Výsledky pro mělkou vícevrstvou perceptronovskou síť využívající me- todu ostatní náhodné–validační sada CIFAR–10 . . . . .	118
B.9	Výsledky pro hlubokou vícevrstvou perceptronovskou síť využívající metodu LSUV–testovací sada CIFAR–10 . . . . .	120
B.10	Výsledky pro hlubokou vícevrstvou perceptronovskou síť využívající metodu LSUV–validační sada CIFAR–10 . . . . .	120
B.11	Výsledky pro hlubokou vícevrstvou perceptronovskou síť využívající metodu náhodná normální–testovací sada CIFAR–10 . . . . .	121
B.12	Výsledky pro hlubokou vícevrstvou perceptronovskou síť využívající metodu náhodná normální–validační sada CIFAR–10 . . . . .	123
B.13	Výsledky pro hlubokou vícevrstvou perceptronovskou síť využívající metodu náhodná rovnoměrná–testovací sada CIFAR–10 . . . . .	124
B.14	Výsledky pro hlubokou vícevrstvou perceptronovskou síť využívající metodu náhodná rovnoměrná–validační sada CIFAR–10 . . . . .	124
B.15	Výsledky pro hlubokou vícevrstvou perceptronovskou síť využívající metody ostatní náhodné–testovací sada CIFAR–10 . . . . .	126

B.16	Výsledky pro hlubokou vícevrstvou perceptronovskou síť využívající metody ostatní náhodné–validační sada CIFAR–10 . . . . .	126
B.17	Výsledky pro jednoduchou CNN využívající metod LSUV–testovací sada CIFAR–10 . . . . .	127
B.18	Výsledky pro jednoduchou CNN využívající metod LSUV–validační sada CIFAR–10 . . . . .	128
B.19	Výsledky pro jednoduchou CNN využívající metodu náhodné normální–testovací sada CIFAR–10 . . . . .	129
B.20	Výsledky pro jednoduchou CNN využívající metodu náhodné normální–validační sada CIFAR–10 . . . . .	131
B.21	Výsledky pro jednoduchou CNN využívající metodu náhodné rovnoměrné–testovací sada CIFAR–10 . . . . .	132
B.22	Výsledky pro jednoduchou CNN využívající metodu náhodné rovnoměrné–validační sada CIFAR–10 . . . . .	132
B.23	Výsledky pro jednoduchou CNN využívající metody ostatní náhodné–testovací sada CIFAR–10 . . . . .	134
B.24	Výsledky pro jednoduchou CNN využívající metody ostatní náhodné–validační sada CIFAR–10 . . . . .	134
B.25	Výsledky pro ResNet18 využívající metodu LSUV–testovací sada CIFAR–10 . . . . .	135
B.26	Výsledky pro ResNet18 využívající metodu LSUV–validační sada CIFAR–10 . . . . .	136
B.27	Výsledky pro ResNet18 využívající metodu náhodné normální–testovací sada CIFAR–10 . . . . .	137
B.28	Výsledky pro ResNet18 využívající metodu náhodné normální–validační sada CIFAR–10 . . . . .	139
B.29	Výsledky pro ResNet18 využívající metodu náhodné rovnoměrné–testovací sada CIFAR–10 . . . . .	139
B.30	Výsledky pro ResNet18 využívající metodu náhodné rovnoměrné–validační sada CIFAR–10 . . . . .	140
B.31	Výsledky pro ResNet18 využívající metody ostatní náhodné–testovací sada CIFAR–10 . . . . .	142
B.32	Výsledky pro ResNet18 využívající metody ostatní náhodné–validační sada CIFAR–10 . . . . .	142

# Úvod

Oblast neuronových sítí a jejich využití je poslední dobou velice populární téma. Při tvorbě samotné sítě se musí určit důležité parametry sítě, takzvané hyperparametry. Volba hyperparametrů může velice ovlivnit proces učení neuronové sítě a proto volba vhodných hyperparametrů neuronových sítí je oblast, která se hodně prozkoumává. Mezi hyperparametry neuronových sítí například patří počet neuronů ve vrstvě, použitá aktivační funkce, krok učení nebo také metoda inicializace vah.

V průběhu učení se váhy pomocí optimalizačního algoritmu upravují až do té doby, než je dosaženo minima chybové funkce. Proto vhodná inicializace vah může zrychlit nalezení tohoto minima a tím zrychlit samotné učení. Naopak nevhodnými počátečními váhami lze negativně ovlivnit proces učení.

Z těchto důvodů se práce zabývá prozkoumáním a porovnáním vlivu na proces učení různých přístupů v oblasti inicializace vah neuronových sítí. Dále jsou porovnávány různé konfigurace vybraných metod inicializace. Jako úloha neuronových sítí je vybraná klasifikace obrázků ze dvou různých databází MNIST a CIFAR-10. Vliv vybraných metod a jejich případných konfigurací je zkoumán pro čtyři architektury.

Práce obsahuje teoretickou rešerši o umělých neuronových sítích a o různých metodách inicializace vah. Dále jsou v práci zmíněny metriky, kterými bylo ohodnoceno učení sítí a popis použitých databází. Předposlední část je věnována návrhnutí a popisu experimentů, které byly navrženy na základě zpracované teorie. Následně jsou uvedeny dosažené výsledky. Následuje diskuse a srovnání dosažených výsledků s literaturou.



# 1 Teoretický úvod

## 1.1 Strojové učení

Strojové učení je jednou z aplikací umělé inteligence, které umožňuje počítačům učit se ze zkušenosti. Zde jsou zkušenostmi myšleny předešlé informace (data), které měl algoritmus strojového učení k dispozici v minulosti[1]. Tato data mohou mít různou podobu. Mohou to být například data získaná snímáním různých veličin v různých prostředích (astronomická data, burza cenných papírů, monitoring záplav, požárů, růstu plodin[2]). Mezi důležité aspekty těchto dat patří jejich kvalita a velikost. Algoritmus strojového učení využívá výpočetní metody, které aplikuje na přijímané data[3]. Algoritmy můžeme obecně rozdělit podle metod učení na algoritmy které se učí s učitelem, algoritmy které se učí bez učitele[3].

Mezi hlavní aplikace například patří:

- a) Klasifikace – zařazení objektů do kategorií (např.: klasifikace textů a dokumentů, klasifikace obrazů)
- b) Regrese – predikce reálné hodnoty pro každý objekt (např.: predikce cen akcií)
- c) Seřazování – seřazení objektů podle určitého kritéria (např.: vyhledávání webových stránek podle relevance k vyhledávanému textu)
- d) Shlukování – shlukování objektů [3]

### Metody učení bez učitele

Algoritmus strojového učení přijímá trénovací data u kterých nezná požadované výstupy. Tento algoritmus se bez jakékoliv zpětné vazby snaží najít podobnosti nebo vzory v přijímaných datech[4]. Na základě těchto podobností nastává shlukování a rozřazování jednotlivých objektů do kategorií, bez možného ověření správnosti takového zařazení[5].

### Metody učení s učitelem

Algoritmus strojového učení přijímá trénovací data u kterých zná požadované výstupy. Vlastní trénování probíhá tak, že algoritmu předložíme jednotlivá data a získáme pro ně určité výstupy. Tyto získané výstupy algoritmus porovná s požadovanými výstupy a získá odchylku. Na základě této odchylky nastává adaptace parametrů algoritmu takzvané učení[5].

## 2 Umělé neuronové sítě

### 2.1 Historie

Jednou z oblastí umělé inteligence jsou umělé neuronové sítě. V 50. letech 20. století McCulloch a Pitts jako první na světě vytvořili matematický model neuronu. O pár let později na jejich práci navázal Donald Hebb, který navrhl pravidlo, díky kterému je možné tento neuron učit[6]. Na konci 60. let 20. století Frank Rosenblatt, inspirovaný prací od McCullocha a Pittse, vytvořil první umělý neuron neboli perceptron[7]. I přes veškeré objevy, vývoj neuronových sítí od 60. let 20. století na 20 let upadá. K znovuobjevení dochází až v 80. letech 20. století a od té doby dochází k rychlému vývoji[6].

### 2.2 Neuron

Základní funkční jednotkou Centrální Nervové Soustavy (dále jen CNS) je neuron (počet neuronů v CNS je asi  $10^{10} - 10^{13}$ ). Neurony jsou schopné přijímat, vydávat a přenášet informace ve formě elektrických impulsů. Jednotlivé neurony jsou navzájem spojeny pomocí synapsí. Synapse jsou spojení, díky kterým může nastávat k přenosu elektrických impulsů na jiné neurony.

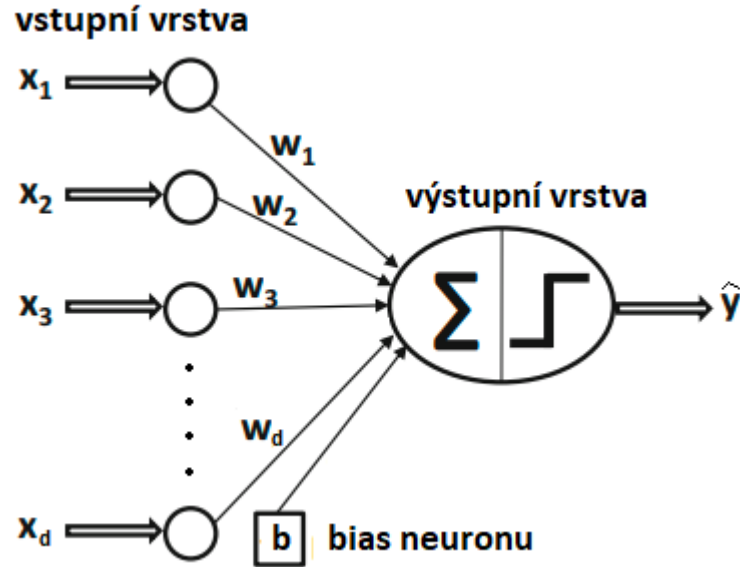
Neuron skládá ze tří hlavních částí: těla, dendritů a axonu. Axon se skládá zpravidla z jednoho vlákna, který vede informaci z těla neuronu. Na konci axonu je terminální rozšíření, kde se předává informace pomocí synapsí dendritům jiných neuronů. Dendrity jsou vícečetné výběžky, které neustále přijímají impulzy od ostatních neuronů. Tělo (perikaryon) neuronu obsahuje cytoplazmu a organely. Tělo má na starosti zpracování informace, které přichází z dendritů[7, 8].

### 2.3 Jednovrstvá dopředná síť

Jedním z typů umělých neuronových sítí jsou jednovrstvé dopředné neuronové sítě. V dopředné síti je daný tok informací a to ze vstupní vrstvy do vrstvy výstupní. Mezi základní typy sítí používající tuto architekturu je ADELINÉ a Perceptron[7].

#### Perceptron

Nejjednodušší umělá neuronová síť je nazývána perceptron viz obrázek 2.1. Tato neuronová síť obsahuje dvě vrstvy. První vrstva je vrstva vstupní, do této vstupní vrstvy vstupují trénovací data. Jednotlivá data jsou ve formě  $(\bar{X}, y)$ , kde každé



Obr. 2.1: Obrázek umělého neuronu, na kterém lze vidět popis částí vstupní a výstupní vrstvy. Písmeno  $w$  značí váhu jednotlivého spojení. Převzato a upraveno z [9].

$\bar{X} = [x_1, \dots, x_d]$  obsahuje  $d$  proměnných a  $y$  obsahuje požadovaný výstup. Počet uzlů ve vstupní vrstvě odpovídá počtu proměnných  $d$ .

Další vrstvou je vrstva skrytá, která je u perceptronu zároveň i vrstvou výstupní. Jednotlivá spojení mezi jednotlivými uzly vstupní vrstvy a výstupní vrstvou jsou ohodnoceny váhami  $\bar{W} = [w_1, \dots, w_d]$ . Důležité je také podotknout, že veškeré výpočty probíhají na výstupní vrstvě, konkrétně se jedná o lineární funkci

$$u = \bar{W} \cdot \bar{X} = \sum_{i=1}^d w_i x_i \quad (2.1)$$

Dále se také k jednotlivým výstupům funkce popsanou rovnicí 2.1 přičítá bias  $b = -\theta$ , kde  $\theta$  je excitační práh neuronu. Výsledná rovnice lineární funkce má tedy podobu

$$u = \bar{W} \cdot \bar{X} = \sum_{i=1}^d w_i x_i + b \quad (2.2)$$

Pokud si bias v rovnici 2.2 vyjádříme jako váhu  $w_0$ , který váhuje fiktivní vstup  $x_0 = 1$  ( $b = w_0 x_0$ ), vyjde nám rovnice ve tvaru

$$u = \bar{W} \cdot \bar{X} = \sum_{i=0}^d w_i x_i \quad (2.3)$$

### Dopředné šíření

Na jednotlivé výsledky (reálné hodnoty) funkce popsané rovnicí 2.3 je aplikována jedna z aktivačních funkcí  $f$  viz kapitola 2.4. Tímto dosáhneme na výstupu prediko-

vanou hodnotu  $\hat{y}$

$$\hat{y} = f\{u\} = f\left\{\sum_{i=0}^d w_i x_i\right\} \quad (2.4)$$

### Učení perceptronu

Učení probíhá v epochách. Jedna epocha znamená, že naším modelem prošli všechny objekty učení. Pokud se predikovaný výstup perceptronu  $\hat{y}$  a očekávaný výstup  $y$  shodují, úprava vah a s tím spojené učení perceptronu neprobíhá. Pokud se ale výstup perceptronu  $\hat{y}$  a očekávaný výstup  $y$  neshodují, nastává adaptace vah[9].

Tato adaptace probíhá pomocí takzvaného delta pravidla [9], které je možné vyjádřit jako

$$\delta = \hat{y} - y \quad (2.5)$$

Velikost takovéto adaptace je závislá na  $\delta$  a kroku učení  $\eta$ . Výslednou adaptaci vah můžeme popsat vztahem

$$w_i^{nová} = w_i^{stará} + \eta \cdot \delta \cdot x_i \quad (2.6)$$

Adaptace pomocí delta pravidla je z dnešního pohledu zastaralá a již se v praxi nevyužívá.

## 2.4 Aktivační funkce

Aktivační funkce dostane na vstup jednu reálnou hodnotu a s touto hodnotou provede určitou matematickou operaci. Aktivační funkce rozhoduje zda li bude perceptron aktivovaný nebo nebude[10].

### ReLU

ReLU<sup>1</sup> je poslední dobou velice populární aktivační funkce. Mezi výhody patří zrychlení konvergence u metody gradientního sestupu oproti funkcím hyperbolický tangens a sigmoida.. Pokud je  $u$  menší nebo rovno nule, výsledná hodnota je rovna 0, jinak je výsledná hodnota funkce rovna  $u$ [1]. Rovnice ReLU může být vyjádřena jako

$$f(u) = \max(0, u) \quad (2.7)$$

---

<sup>1</sup>z ang. Rectified Linear Unit

## Softmax

Funkce softmax se velice hodí pro klasifikaci objektů do více tříd, a proto se hojně používá ve výstupní vrstvě neuronových sítí. Na výstup této funkce dostaneme pravděpodobnost příslušnosti výstupu k jednotlivým třídám[10]. Softmax funkci můžeme popsat rovnicí

$$f(u_i) = \left( \frac{e^{u_i}}{\sum_j e^{u_j}} \right) j = 1, \dots, n \quad (2.8)$$

## 2.5 Chybové funkce

Chybová funkce, někdy také nazývaná kriteriální funkce se používá při procesu trénování. Jedná se o ohodnocení našeho modelu. Díky této chybové funkci můžeme vypočítat rozdíl mezi predikovaným výstupem  $\hat{y}$  a výstupem požadovaným  $y$ . Při učení je snaha o co nejnižší hodnotu chybové funkce, protože čím menší je hodnota zvolené chybové funkce, tím menší je rozdíl mezi hodnotou predikovanou a požadovanou[11].

### Kategorická vzájemná entropie

Kategorická vzájemná entropie, dále jen CCE<sup>3</sup> je chybová funkce, která se využívá zejména při klasifikaci do více třídy. CCE počítá chybu pomocí predikovaného pravděpodobnostního rozdělení a požadovaného pravděpodobnostního rozdělení. Výsledná rovnice je dána vztahem:

$$CCE = -\frac{1}{N} \sum_{i=1}^N \sum_{j=1}^C y_i \cdot j \log(p_i, j) \quad (2.9)$$

kde N je počet vzorků, C je počet tříd, y je požadovaná hodnota a p je predikovaná pravděpodobnost pro jednotlivé třídy.

Při použití CCE je vhodné, aby požadované hodnoty pro jednotlivé vzorky byly přepsány pomocí one-hot kódování[11].

## 2.6 Optimalizační algoritmy

Optimalizační algoritmy určují jak se budou upravovat váhy v průběhu procesu učení, tak aby docházelo k pokles chyby, kterou získáme pomocí chybové funkce[12]. Použitím optimalizačních algoritmů také dosáhneme zvýšení přesnosti modelu, bez nutnosti navýšení času trénování[13].

---

<sup>3</sup>z ang. Categorical Cross Entropy

## Stochastický gradientní sestup

Mezi jeden z nejpobulárnějších optimalizačních algoritmů patří stochastický gradientní sestup, dále jen SGD<sup>4</sup>. Při použití SGD můžeme popsat úpravu vah jednotlivých trénovacích vzorků pomocí následující rovnice

$$\theta_{t+1} = \theta_t - \eta \nabla C(\theta_t; x_i; y_i) \quad (2.10)$$

kde  $\theta_{t+1}$  je vektor nových vah  $w$ ,  $\theta_t$  je vektor starých vah  $w$ ,  $C$  je chybová funkce,  $\nabla C$  je gradient chybové funkce a  $\eta$  je krok učení.  $\nabla C$  je derivace chybové funkce podle vektoru vah  $\theta_t$  ( $\nabla C = \frac{\partial C}{\partial \theta_t}$ ) [13, 14].

## Adaptivní odhad momentu

Adaptivní odhad hybnosti, především známý jako Adam<sup>5</sup> bere v potaz exponenciální klouzavý průměr gradientu a gradientu umocněného na druhou. To má za následek zlepšení učení. Další změnou oproti SGD je použití odhadu momentu, díky tomuto odhadu můžeme snadněji najít globální minimum [13, 14]. Jednotlivé vztahy můžeme popsat následujícími rovnicemi:

$$\begin{aligned} m_t &= \beta_1 m_{t-1} + (1 - \beta_1) g_t \\ v_t &= \beta_2 v_{t-1} + (1 - \beta_2) g_t^2 \end{aligned} \quad (2.11)$$

kde  $m_t$  a  $v_t$  je odhad momentů gradientů,  $\beta_1$  a  $\beta_2$  jsou jednotlivé momenty.  $m_t$  a  $v_t$  jsou vektory, které na počátku inicializace obsahují samé nuly. To vede k určitému ovlivnění výsledných odhadů momentů gradientů. Abychom se zbavili ovlivnění, je nutné přepsat odhady jako

$$\hat{m}_t = \frac{m_t}{1 - \beta_1^t}, \hat{v}_t = \frac{v_t}{1 - \beta_2^t} \quad (2.12)$$

nyň už můžeme definovat rovnici pro úpravu vah. Tato rovnice má tvar

$$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{\hat{v}_t} + \epsilon} \hat{m}_t \quad (2.13)$$

kde  $\theta_{t+1}$  je vektor nových vah,  $\theta_t$  je vektor starých vah  $w$ ,  $\hat{m}_t$  a  $\hat{v}_t$  jsou upravené odhady momentů gradientů,  $\eta$  je krok učení a  $\epsilon$  je malá hodnota zabraňující dělení nulou [14, 13].

## 2.7 Vícevrstvá dopředná síť

Vícevrstvé sítě kromě vstupní a výstupní vrstvy obsahují i další vrstvy. Jelikož jednotlivé výpočetní operace, v těchto dalších vrstvách, jsou pro uživatele skryté,

---

<sup>4</sup>z ang. Stochastic Gradient Descent

<sup>5</sup>z ang. Adaptive moment estimate

tyto vrstvy se nazývají skryté. Stejně jako u jednovrstvé neuronové dopředné sítě, je tok informací daný. V případě vícevrstvé sítě je tok informací ze vstupních vrstev do skrytých vrstev a z vrstev skrytých do vrstev výstupních. Také je nutné zmínit, že všechny uzly ve vrstvě, jsou propojeny se všemi uzly ve vrstvě následující[9].

### Vícevrstvá perceptronová síť

Vícevrstvá perceptronová síť obsahuje jednu vrstvu vstupní, jednu vrstvu výstupní a minimálně jednu a nebo více vrstev skrytých. Jednotlivé uzly ve všech vrstvách jsou navzájem propojeny. Mezi aplikace takovýchto sítí například patří dělání předpovědí, rozpoznávání vzorů a separace více lineárně neseparovatelných tříd[7, 9].

### Učení vícevrstvého perceptronu

Do vstupní vrstvy vstupují objekty ve formě  $(\bar{X}, y)$ , kde  $\bar{X}$  obsahuje pro každý objekt  $d$  proměnných a je ve tvaru  $\bar{X} = [x_0, x_1, \dots, x_d]$  a  $y$  obsahuje pro každý objekt  $d$  proměnných a je ve tvaru  $y = [y_1, \dots, y_d]$ . Dále potřebujeme vyjádřit váhy  $\bar{W}^l$ , které váhují jednotlivá spojení mezi všemi vrstvami. Jednotlivé váhy můžeme popsat jako  $w_{jk}^l$ , kde váha vede z  $k$ -tého neuronu ve vrstvě  $l-1$ , do  $j$ -tého neuronu ve vrstvě  $l$ . Pokud si vyjádříme bias  $b$  vyjádříme jako  $b = w_0x_0$ , kde  $x_0 = 1$ , tak získáme matici jednotlivých vah mezi jednotlivými vrstvami[15]. Tuto matici můžeme vyjádřit jako

$$\bar{W}^l = \begin{bmatrix} w_{10}^l & w_{11}^l & w_{12}^l & \cdots & w_{1k}^l \\ w_{20}^l & w_{21}^l & w_{22}^l & \cdots & w_{2k}^l \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ w_{j0}^l & w_{j1}^l & w_{j2}^l & \cdots & w_{jk}^l \end{bmatrix}$$

Mezi jednotlivými vrstvami jsou jednotlivé vstupy váhovány váhami z matice  $\bar{W}^l$ . Výsledný produkt  $z_k^l$  dosadíme do aktivační funkce a získáme  $a_k^l$  viz rovnice 2.14 a 2.15.

$$z_k^l = \sum_{i=0}^d w_{jk}^l a_k^{l-1} \quad (2.14)$$

$$a_k^l = f(z_k^l) \quad (2.15)$$

U  $z_k^1$  je  $a_k^{l-1} = \bar{X}$ , výsledný vztah můžeme popsat viz rovnice 2.16.

$$z_k^1 = \begin{bmatrix} w_{10}^l & w_{11}^l & w_{12}^l & \cdots & w_{1k}^l \\ w_{20}^l & w_{21}^l & w_{22}^l & \cdots & w_{2k}^l \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ w_{j0}^l & w_{j1}^l & w_{j2}^l & \cdots & w_{jk}^l \end{bmatrix} \circ \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_k \end{bmatrix} = \begin{bmatrix} z_1^1 \\ z_2^1 \\ \vdots \\ z_k^1 \end{bmatrix} \quad (2.16)$$

Pomocí výstupu výstupní vrstvy  $a_k^l$  a požadovaného výstupu  $y$ , získáme hodnotu  $C(d)$  použitím chybové funkce  $L$ , pro každý  $d$ -tý trénovací vzorek viz rovnice 2.17, se

kteřou pak následně budeme počítat v optimalizačním algoritmu. Je nutné upravit všechny váhy pomocí algoritmu zpětného šíření chyby, za využití obecného delta pravidla [7, 9, 15].

$$C(d) = L(y_d, a_k^l) \quad (2.17)$$

Pokud jako optimalizační algoritmus použijeme metodu gradientní sestup, úpravy jednotlivých vah můžeme popsat následovně

$$w_{jk}^l = w_{jk}^l - \eta \nabla C \quad (2.18)$$

$\nabla C$  lze vyjádřit pomocí řetězového pravidla vyjádřit jako

$$\nabla C(d) = \frac{\partial C(d)}{\partial w_{jk}^l} = \frac{\partial C(d)}{\partial a_k^l} \frac{\partial a_k^l}{\partial z_k^l} \frac{\partial z_k^l}{\partial w_{jk}^l} \quad (2.19)$$

kde  $\frac{\partial C(d)}{\partial a_k^l}$  je derivace chybové funkce podle  $a_k^l$ ,  $\frac{\partial a_k^l}{\partial z_k^l}$  je derivace aktivační funkce podle  $z_k^l$  a  $\frac{\partial z_k^l}{\partial w_{jk}^l}$  je aktuální vstup neuronu viz rovnice 2.20

$$\begin{aligned} \frac{\partial C(d)}{\partial a_k^l} &= C'(y_d, a_k^l) \\ \frac{\partial a_k^l}{\partial z_k^l} &= f'(z_k^l) \\ \frac{\partial z_k^l}{\partial w_{jk}^l} &= a_k^{l-1} \end{aligned} \quad (2.20)$$

Pokud si vyjádříme chybu  $j$ -tého neuronu v  $l$ -té vrstvě jako delta  $\delta_j^l$  viz rovnice 2.21.

$$\delta_j^l = \frac{\partial C(d)}{\partial z_k^l} = \frac{\partial C(d)}{\partial a_k^l} \frac{\partial a_k^l}{\partial z_k^l} = C'(y_d, a_k^l) f'(z_k^l) \quad (2.21)$$

pak můžeme rovnici 2.19 přepsat jako

$$\nabla C(d) = \frac{\partial C(d)}{\partial w_{jk}^l} = \delta_j^l a_k^{l-1} \quad (2.22)$$

Rovnici pro výpočet následujících  $\delta_j^l$  můžeme definovat jako

$$\delta_j^l = \sum_k \delta_k^{l+1} w_{kj}^{l+1} f'(z_k^l) \quad (2.23)$$

Zprůměrováním všech gradientů p trénovacích vzorků získáme  $\nabla C$  viz rovnice 2.24

$$\nabla C = \frac{\partial C}{\partial w_{jk}^l} = \frac{1}{p} \sum \frac{\partial C(d)}{\partial w_{jk}^l} \quad (2.24)$$

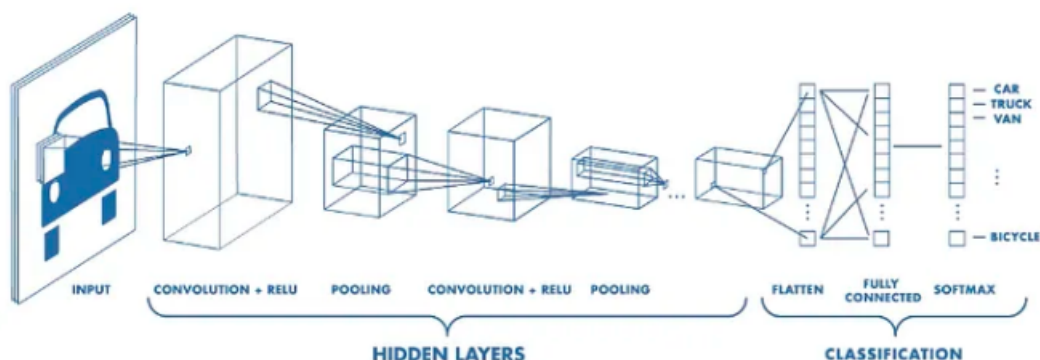
Takto vyjádřené  $\nabla C$  použijeme do rovnice 2.18, výsledná rovnice úpravy vah má tedy tvar [7, 9, 15]

$$w_{jk}^l = w_{jk}^l - \eta \frac{\partial C}{\partial w_{jk}^l} \quad (2.25)$$



## 2.8 Konvoluční neuronová síť

Jedním z typů neuronových sítí jsou konvoluční neuronové sítě, dále jen CNN<sup>6</sup>, které pracují s daty s mřížkovou topologií, například obrázky nebo textem. Z názvu této sítě vyplývá, že síť používá matematickou operaci konvoluce. CNN obsahuje obvykle konvoluční, kompresní vrstvy a vrstvu plně propojených neuronů viz obrázek 2.2[9, 16].



Obr. 2.2: Ilustrační obrázek architektury CNN. popisující jednotlivé vrstvy CNN. převzato z [17]

Konvoluční vrstva provádí skalární součin mezi dvěma maticemi, jedna matice je filtr a druhá matice je například celý nebo část obrázku. Rozměry filtru nesmí být větší než rozměry obrázku. Klouzáním filtru po obrázku získáme reprezentaci dané oblasti. Tímto se vytvoří nová dvourozměrná reprezentace původního obrázku tzv. mapa příznaků. Jako součást konvoluční vrstvy je také použita aktivační funkce, například ReLu[1, 9, 16, 17].

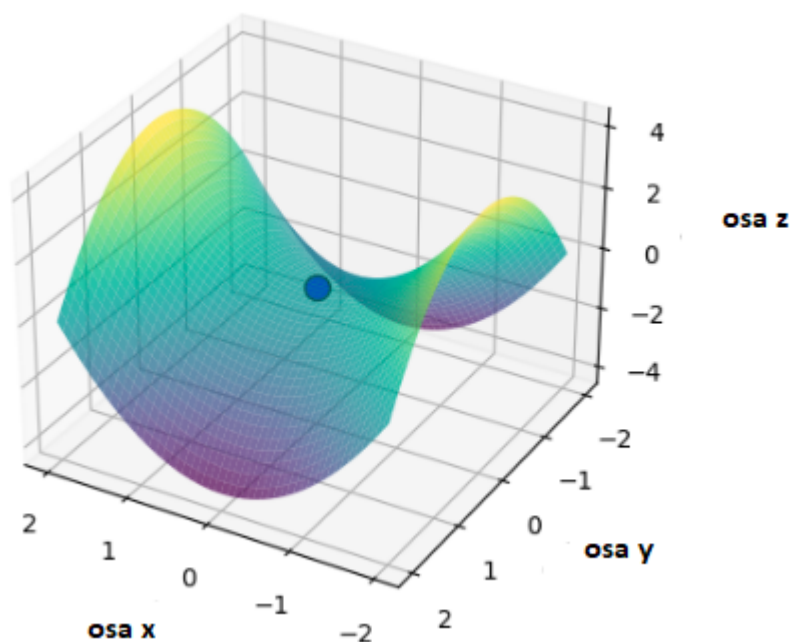
Kompresní vrstva („pooling“) provede kompresi výstupu konvoluční vrstvy a tím zmenší rozměry mapy příznaků. Většinou se jedná o operaci, která ve čtvercové oblasti hledá maximum, minimum nebo průměr[1, 16, 17].

Vrstva plně propojených neuronů je podobná jako u běžná vícevrstvé perceptronové sítě, Odlišnost je pouze v použití operace, díky které lze převést matici o rozměrech  $M \times N$  na matici o rozměrech  $1 \times (M \cdot N)$ [1, 17].

<sup>6</sup>z ang. Convolutional Neural Network

### 3 Inicializace vah

Výběr vhodných hyperparametrů je základ při tvorbě kvalitní neuronové sítě. Mezi hyperparametry sítě patří například: počet neuronů ve vrstvách, počet skrytých vrstev, použitý optimalizační algoritmus nebo také metoda inicializace vah. Právě metoda inicializace vah je velice důležitý, ale často opomíjený hyperparametr[18]. Vhodným výběrem metody inicializace vah můžeme podstatně ovlivnit proces učení. V průběhu učení se počáteční váhy opakovaně upravují, dokud hodnota chybové funkce nedosáhne svého minima. Vhodně zvolené počáteční váhy ovlivňují rychlost konvergence sítě[19]. Naopak nevhodně zvolená metoda inicializace vah může zabránit optimalizačnímu algoritmu konvergovat[18]. Další problémy, které se mohou vyskytnout při špatné inicializaci jsou: mizející gradient, explodující gradient[19], problémy uvíznutí v lokálním minimu chybové funkce u jednodušších typů neuronových sítí nebo problém uvíznutí v tzv. sedlovém bodu chybové funkce u složitějších neuronových sítích a vysoká počáteční chyba. Vizualizaci sedlového bodu lze vidět na obrázku 3.1[20].



Obr. 3.1: Vizualizace sedlového bodu chybové funkce, která má tvar  $f(x, y) = x^2 - y^2$ . Sedlový bod označen modrou kuličkou.

Metody inicializace vah, lze obecně rozdělit podle trénovacího algoritmu, a to na iterativní a neiterativní. Iterativní lze rozdělit na metody bez předchozího učení sítě a na metody s předchozím učením sítě. Pro sítě používající neiterativní trénování se především využívají evoluční algoritmy.

Mezi metody inicializace vah bez předchozího učení sítě patří náhodná inicializace, inicializace, která vychází z informací o trénovacích datech a hybridní metody, které kombinují náhodnou inicializaci s inicializací, která vychází z informací o trénovacích datech[19].

Jedním z počátečních myšlenek, jak inicializovat váhy bylo nastavení všech vah na 0. Toto se ukázalo jako chyba, protože se všechny neurony učí stejně, mají stejný výstup, a při algoritmu zpětného šíření chyby se váhy také upraví stejně.[19] A tak se neuronová síť chová jako lineární model.[21]

## 3.1 Metody náhodné inicializace vah

Jednotlivé váhy jsou buď přímo vybírány z náhodného rozdělení nebo jsou zprvu vybírány z náhodného rozdělení a poté dále upravovány pomocí různých metod[19]. Hlavní myšlenkou náhodné inicializace je vybírání různých hodnot blízkých nule, tak abychom rozbili symetričnost modelu. [1]

Metody náhodné inicializace můžeme rozdělit na inicializaci z intervalu, inicializaci založenou na škálování rozptylu a ostatní metody náhodné inicializace[19].

### 3.1.1 Metody inicializace vah z intervalu

Váhy jsou vybírány z optimálního intervalu nebo také náhodně vybírány a upravovány tak, aby nepřesahovaly určitá minimální nebo maximální hranice[19].

#### Metoda Nguyen–Widrow

Tato metoda byla popsána pouze pro dvojvrstvou perceptronovskou síť. Na začátku se každému neuronu ve skryté vrstvě sítě přiřadí určitý interval z celého možného rozsahu vah. Celý rozsah vah je definován podle řešeného problému. Následuje náhodné vybírání vah z rovnoměrného pravděpodobnostního rozdělení. Následuje uprava vah tak aby jednotlivé uzly ve skryté vrstvě byly lineární pouze na svém určeném intervalu. Při procesu učení mohou jednotlivé uzly ve skryté vrstvě libovolně upravovat pozici a velikost intervalu[22].

#### Metoda Kim–Ra

Y.K Kim a J.B Ra představili metodu inicializace vah, pro vícevrstvé dopředné sítě, na základě pozorování dynamiky při tvorbě rozhodovací hranice. Jednotlivé váhy jsou inicializovány z uniformního pravděpodobnostního rozdělení[19] tak, aby byla splněna podmínka o minimální velikosti váhy.[23] Kde pro jednotlivé počáteční váhy  $w_i$  musí platit:  $|w_i| < \sqrt{\frac{\eta}{N}}$ , pro  $i=0, \dots, N$ , kde  $\eta$  je rychlost učení.[24]

## Metoda inicializace vah pomocí vzájemné informace

Vzájemná informace je jeden z konceptů v teorii informací. Vzájemná informace udává závislost dvou proměnných. Užitečné informace ze vstupních proměnných jsou získány pomocí výpočtu vzájemné informace mezi vstupními a výstupními proměnnými. Počáteční rozdělení vah je v souladu s rozdělením informací, které jsem získali ze vstupních proměnných. Tato metoda byla použita pro dopřednou vícevrstvou síť používající ve skrytých vrstvách jako aktivační funkci sigmoidu. Díky téhle metodě mají počáteční váhy vyšší šanci se přiblížit globálnímu minimu a tím udělat síť efektivnější a stabilnější. Abychom zařídili, aby všechny neurony ve skryté vrstvě byly aktivní a nedocházelo k jejich předčasné saturaci, je vypočítán interval, ze kterého se váhy vybírají. Nakonec se vyberou váhy z poslední skryté vrstvy to výstupní vrstvy z rovnoměrného rozdělení z intervalu  $\langle -1, 1 \rangle$ [25].

## Náhodná inicializace z rovnoměrného pravděpodobnostního rozdělení

Hodnoty vah jsou získávány z rovnoměrného pravděpodobnostního rozdělení a to stanovením limitů maximální a minimální hodnoty  $\langle \text{minimum}, \text{maximum} \rangle$ . Tato metoda byla používána jako jedna z prvních, ale je zde velice těžké odhadnout správné rozmezí. Optimální rozmezí se pro vícevrstvou perceptronovskou síť pohybovalo od  $\langle -4, 4 \rangle$  až do  $\langle -0, 5; 0, 5 \rangle$ [26].

## Náhodná inicializace z rovnoměrného pravděpodobnostního rozdělení

Hodnoty vah jsou získávány z rovnoměrného pravděpodobnostního rozdělení, kde průměr je 0 a volíme směrodatnou odchylku  $\sigma$ . Vhodným zvolením odchylky  $\sigma$ , dostaneme malé hodnoty vah, které rozbijí symetričnost neuronů při inicializace všech vah na stejnou hodnotu[1].

### 3.1.2 Metody inicializaci založené na škálování rozptylu

Počáteční váhy jsou vybírány z náhodného pravděpodobnostního rozdělení a později je nejčastěji škálován tak, aby se zachoval rozptyl výstupu nebo vstupu nebo obou[19].

## LeCun inicializace

LeCunova inicializace bere v potaz počet v neuronů ve vrstvě  $r_{in}$ . Řeší problém explodujícího a mizejícího gradientu, ale také i problém růstu rozptylu vah při růstu počtu neuronů ve vrstvě[27]. Počáteční váhy jsou získány náhodným výběrem z normálního rozdělení, kde průměr je nastaven na hodnotu 0 a směrodatná odchylka

je rovna  $1/\sqrt{r_{in}}$ . Další možností je získání vah náhodným výběrem z rovnoměrného pravděpodobnostního rozdělení  $\langle -limit, limit \rangle$ , kde je limit roven  $\sqrt{\frac{3}{r_{in}}}$  [21].

### Náhodná inicializace z rovnoměrného pravděpodobnostního rozdělení

Jednotlivé počáteční hodnoty vah jsou získány náhodným výběrem z rovnoměrného rozdělení v určitém rozmezí, například  $\langle -1/\sqrt{r_{in}}, 1/\sqrt{r_{in}} \rangle$ , kde  $r_{in}$  je počet vstupních neuronů [9]. Pro tuto metodu je vhodné použití aktivační funkce sigmoida [28]

### Xavier inicializace

Xavierova inicializace, známá také jako Glorotova nejprve inicializuje rovnoměrného pravděpodobnostního rozdělení a později je škálován pomocí počtu neuronů v aktuální vrstvě  $r_{in}$  a počtu neuronů ve vrstvě následující  $r_{out}$ . Toto vede k eliminaci explodujícího a mizejícího gradientu. Také se zlepšila propagace hodnot neuronů při dopředném šíření a gradientu při zpětném šíření.

Navržená inicializace náhodně vybírá váhy z rovnoměrného pravděpodobnostního rozdělení. Rozmezí tohoto rozdělení, lze zapsat jako  $\langle -limit, limit \rangle$  kde limit je roven  $\sqrt{\frac{6}{r_{in}+r_{out}}}$  [29]. Další možností je náhodné vybírání vah z normálního pravděpodobnostního rozdělení, který má hodnotu průměru rovnou 0 a směrodatnou odchylku rovnou  $\sqrt{\frac{2}{r_{in}+r_{out}}}$  [21].

Metoda vykazuje dobrou kompatibilitu s neuronovými sítěmi, které využívají aktivační funkci hyperbolický tangens [29]. Pro aktivační funkci ReLU není Xavierova inicializace vhodná a to proto, že při tvorbě této inicializace byla uvažováno použití lineární aktivační funkce a to ReLU není [30].

### He inicializace

He inicializace byla vymyšlena především pro neuronové sítě používající aktivační funkci ReLU<sup>7</sup>. Tyto sítě se jednodušeji trénují a dosahují lepší konvergence než klasické sigmoidální sítě. Dosavadní metody, například Xavierova inicializace nebyla vhodná, protože při tvorbě této inicializace bylo uvažováno použití lineární aktivační funkce. Proto byla vymyšlena He inicializace. He inicializace oproti Xavier inicializace dosahuje i u velice hlubokých neuronových sítí, používající ReLU, konvergence.

Bere v potaz počet neuronů v aktuální vrstvě  $r_{in}$ . Jednotlivé váhy jsou vybírány z normálního pravděpodobnostního rozdělení, kde průměr je roven 0 a směrodatná odchylka rovna  $\sqrt{\frac{2}{r_{in}}}$ . Druhou možností je náhodné vybírání vah z rovnoměrného

---

<sup>7</sup>z ang. Rectified Linear Unit

pravděpodobnostního rozdělení, kde rozmezí je dáno jako  $\langle -limit, limit \rangle$  kde limit je roven  $\sqrt{\frac{6}{r_{in}}}$ [30].

### 3.1.3 Ostatní metody náhodné inicializace vah

#### Inicializace pomocí rozkladu na ortonormální bázi

Jednotlivé matice vah  $W$  jsou získány pomocí výběru jednotlivých vah z náhodného normálního rozdělení. Tyto matice vah  $W$  pro jednotlivé vrstvy jsou následně rozloženy na ortonormální bázi pomocí singulárního rozkladu viz rovnice 3.1.

$$W = U \Sigma V^T \quad (3.1)$$

kde  $W$  jsou původní váhy,  $V^T$  a  $U$  jsou unitární ortogonální matice, které můžeme použít pro inicializaci,  $\Sigma$  je matice, která má nenulové prvky, zvané singulární hodnoty, pouze na diagonále. Nebo lze použít QR rozklad viz rovnice 3.2.

$$W = QR \quad (3.2)$$

kde  $W$  jsou původní váhy,  $Q$  je ortogonální matice a  $R$  je horní trojúhelníková matice. Jako váhy použijeme získanou matici  $Q$ [31]. Metoda splňuje podmínku dynamické izometrie při které jakobiáni získané zpětném šíření chyby, byly rovny zhruba 1. Inicializace pomocí rozkladu na ortonormální bázi toto splňuje a tím zajišťuje nezávislost doby učení sítě na hloubce sítě[19, 31].

## 3.2 Metody inicializace vycházející z informací o trénovacích datech

#### Inicializace vah pomocí analýzy hlavních komponent pro filtry CNN

Váhy pro filtry pro CNN byly získány extrakcí vzorků o stejných rozměrech z trénovacího datasetu. Převedení těchto vzorků do formy matic a použití analýzy hlavních komponent na tyto získané matice. Pro inicializace vah filtrů CNN byly použity vlastní vektory získané z analýzy hlavních komponent. Tato metoda zrychlila proces učení a zlepšila správnost klasifikace[19].

## 3.3 Hybridní metody inicializace

#### LSUV

LSUV<sup>8</sup> je metoda, která používá informace ze subsetu z trénovacích dat pro inicializaci vah. Nejprve se získají váhy z normálního rozdělení, který má rozptyl roven 1

<sup>8</sup>z ang. Layer-Sequential Unit-Variance

a průměr roven 0. Dále provedeme jejich rozklad na ortonormální bázi pomocí singulárního rozkladu nebo QR rozkladu. Následně probíhá iterativní průchod subsetu vrstvami, při kterém se určí rozptyl výstupních hodnot, dále probíhá škálování vah tak, aby se rozptyl výstupů vrstvy rovnal 1. Inicializace vah se pro jednotlivé vrstvy může ukončit třemi způsoby. První způsob je, že rozptyl výstupů je roven 1. Druhým způsobem je dosažení maximálních předem definovaných počtů průchodu subsetu vrstvou. Posledním způsobem ukončení inicializace je dosažení rozptylu výstupů, který větší než námi předem určená minimální hodnota[32].

### 3.4 Transfer learning

Proces přeneseného učení je definovaný jako získání jednotlivých vah, které byly získány z již natrénované sítě, která byla učena na podobném úkolů. Jelikož jsou tyto váhy optimalizované, samotné učení sítě je rychlejší[33]. Přenesené učení se také dá využít když máme málo nových dat, na kterých chceme trénovat model. Moderní CNN natrénované na velkém množství dat vykazují formování stejných hierarchií rysů obrázku. Tyto před trénované hierarchie rysů můžeme dobře využít pro různé aplikace. Tato metoda je velice úspěšná, protože jednotlivé váhy, jíž obsahují mnoho užitečných informací[34].

## 4 Metriky hodnocení učení

### 4.1 Metriky hodnocení klasifikace do více tříd

Nejprve si musíme definovat hodnoty pro binární klasifikaci TP<sup>9</sup>, TN<sup>10</sup>, FP<sup>11</sup> a FN<sup>12</sup> viz obrázek 4.1

Pokud budeme brát příklad z obrázku 4.1 tak

- TP udává počet predikovaných výsledků, které byly skutečně pozitivní, tedy počet pacientů, kteří byly klasifikováni správně jako pacienti.
- TN udává počet predikovaných výsledků, které byly skutečně negativní, tedy počet kontrol, které byly klasifikovány správně jako kontroly
- FN udává počet predikovaných výsledků, které byly falešně negativní, tedy počet pacientů, kteří byly klasifikováni špatně a to jako kontroly.
- FP udává počet predikovaných výsledků, které byly falešně pozitivní, tedy počet kontrol, které byly klasifikovány špatně a to jako pacienti [5]

Klasifikaci do více tříd, můžeme chápat jako set dílčích binárních klasifikací, jeden pro každou třídu. TP, FN, TN a FP pro klasifikaci do více tříd získáme z matice záměn pro více tříd [35] viz obrázek 4.2

#### Přesnost a výtěžnost

Přesnost nám udává, jak přesně dokáže náš model zařadit objekty z konkrétní třídy do jejich správné třídy. Pro výpočet přesnosti použijeme rovnici 4.1

$$přesnost = \frac{TP}{TP + FP} \quad (4.1)$$

---

<sup>9</sup>z ang. True Positive

<sup>10</sup>z ang. True Negative

<sup>11</sup>z ang. False Positive

<sup>12</sup>z ang. False Negative

Správná třída	Výsledek klasifikace	
	Pacienti (+)	Kontroly (-)
Pacienti (+)	TP	FN
Kontroly (-)	FP	TN

Obr. 4.1: Matice záměn pro klasifikaci do dvou tříd, pacienti a kontroly. převzato z [5]





		PREDICTED classification				Total
		Classes	a	b	c	
ACTUAL classification	a	6	0	1	2	9
	b	3	9	1	1	14
	c	1	0	10	2	13
	d	1	2	1	12	16
Total		11	11	13	17	52

Obr. 4.3: Příklad matice záměn, pro 4 různé třídy a, b, c a d, převzato z [37]

### F1–skóre

Metrika F1–skóre ohodnocuje celkovou výkonnost modelu. F1–skóre může být interpretováno jako váhovaný průměr mezi výtěžností a přesností, kde nejlepší hodnota je 1 a nejhorší hodnota je 0[37].

$$F1 - skóre = 2 \frac{přesnot \cdot výtěžnost}{přesnot + výtěžnost} \quad (4.4)$$

## 4.2 Počet epoch

Předmětem hodnocení bude počet potřebných epoch trénování k dosažení určité chyby a správnosti na validační sadě. Dále budeme hodnotit počet epoch potřebných k dosažení optimálního naučení sítě. Tímto způsobem můžeme hodnotit rychlost učení sítě.

## 5 Formulace problému

Problém spočívá v nalezení optimálních počátečních vah tak, aby se co nejlépe v procesu učení za použití optimalizačního algoritmu, minimalizoval rozdíl mezi požadovaným a vypočteným výstupem a tím se snížila hodnota chybové funkce[38].

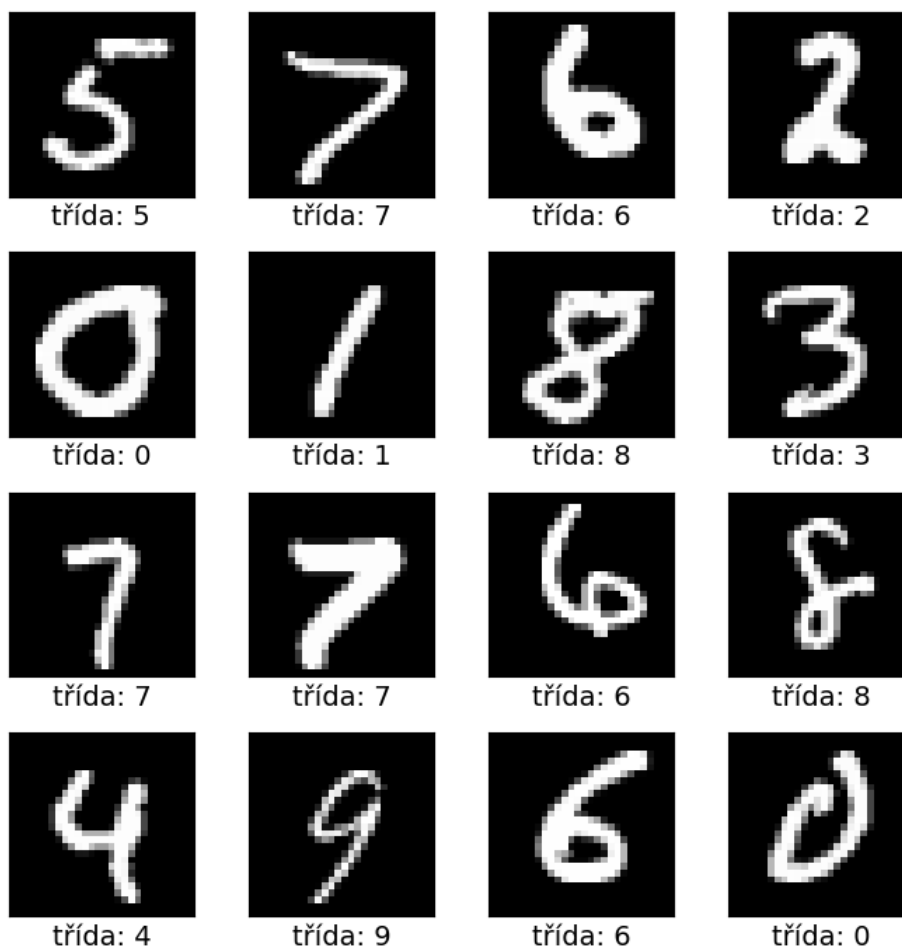
Malé hodnoty počátečních vah blízké nule zvyšují rychlost konvergence chybové funkce. Počáteční váhy s velkým rozptylem mohou vést k rychlé saturaci aktivačních funkcí a tím zpomalit nebo úplně zastavit proces učení. Také velmi malé hodnoty počátečních vah mohou výrazně zpomalit proces učení. Nastavení počátečních hodnot na konstanty také není ideální[1, 19, 27, 38].

Váhy po trénování obsahují informace, které se neuronová síť naučila, když byla trénována na trénovacích datech. Proto vhodně zvolená metoda inicializace vah, zefektivňuje nalezení co nejlepšího lokálního minima chybové funkce a tím zlepšuje proces učení[1, 9, 19, 27, 38].

## 6 Použité databáze

### 6.1 Databáze MNIST

Databáze MNIST<sup>13</sup>, který je dostupný z [39], obsahuje šedotónové obrázky ručně psaných číslic od 0 do 9. Jedná se o subset, který byl vytvořen z většího dataset z NIST. Jednotlivé obrázky mají stejné rozměry, a to 28 pixelů na šířku a 28 pixelů na výšku a jsou centrovány. Dataset celkově obsahuje 70 000 obrázků, které jsou přiřazeny do příslušných tříd. Trénovací dataset obsahuje 60 000 obrázků a testovací dataset obsahuje 10 000 obrázků. Mezi hlavní výhody datasetu patří to, že není nutné provádět rozsáhlé formátování a předzpracování[39]. Ukázka obrázků z datasetu viz obrázek6.1, který byl vytvořen pomocí funkce vygenerované ChatGPT[40]

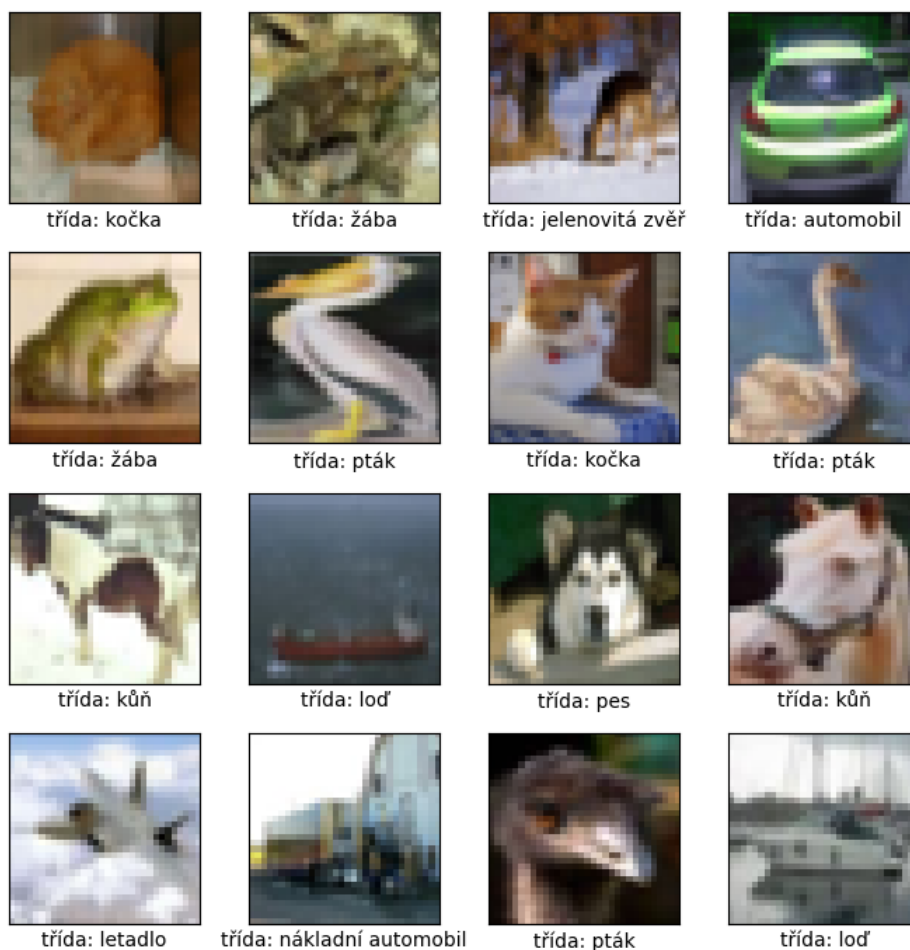


Obr. 6.1: Ukázka šedotónových obrázků z databáze MNIST, které zobrazují ručně psané číslice od 0 do 9. Jednotlivé obrázky jsou zařazeny do příslušných tříd. Obrázek byl vytvořen pomocí funkce vygenerované ChatGPT[40].

<sup>13</sup>z ang .Modified National Institute of Standards and Technology database

## 6.2 CIFAR–10

CIFAR<sup>14</sup> byla vytvořena v[41] je dataset dostupný z [42]. Jedná se o subset z datasetu Tiny Images. CIFAR–10 obsahuje 10 tříd, jmenovitě: letadlo, automobil, pták, kočka, jelenovitá zvěř, pes, žába, kůň, loď a nákladní automobil. Dataset obsahuje celkem 60 000 obrázků. Jednotlivé třídy obsahují 5000 trénovacích obrázků a 1000 testovacích obrázků. Obrázky mají jednotné rozměry a to 32 pixelů na šířku a 32 pixelů na výšku[42]. Ukázka obrázků s příslušným označením tříd z datasetu CIFAR–10 lze vidět na obrázku 6.2, který byl vytvořen pomocí funkce vygenerované ChatGPT[40].



Obr. 6.2: Ukázka obrázků z databáze CIFAR–10. Jednotlivé obrázky jsou zařazeny do příslušných tříd. Zařazení jednotlivých obrázků do tříd je vidět na popiscích pod obrázky. Obrázek byl vytvořen pomocí funkce vygenerované ChatGPT[40].

<sup>14</sup>z ang. Canadian Institute for Advanced Research

## 7 Praktická část

### 7.1 Navrhnutí experimentů

Veškeré experimenty byly programovány v programovacím jazyce Python ve vývojovém prostředí Google Colab. Dále byly použity knihovny Tensorflow a Keras

Na základě teorie bylo navrženo 5 experimentů. Zvolená úloha pro experimenty je klasifikace obrázků. Jako první a úvodní experiment byla navrhnutá a implementována základní neuronovou sít, konkrétně mělkou perceptronovskou sít, s metodou inicializace vah na hodnotu 0. Na této základní neuronové síti bylo provedeno učení na databázi MNIST. Toto základní učení ověří tvrzení z [1] o nevhodnosti použití nul jako počáteční váhy. Další experimenty sledují vliv vybraných inicializací a jejich různých konfigurací na učení mělké vícevrstvé perceptronovské sítě, hluboké vícevrstvé perceptronovské sítě, jednoduché konvoluční neuronové sítě a hluboké konvoluční neuronové sítě – ResNet18.

Vybrané metody inicializace vah jsou: náhodná inicializace z rovnoměrného rozdělení, náhodná inicializace z normálního rozdělení, He, LeCun, Glorot metoda, metoda rozkladu na ortonormální bázi a metoda LSUV. Tyto metody byly vybrány, protože reprezentují rozdílné přístupy v inicializaci vah.

Vlivy jednotlivých metod na učení pro jednotlivé sítě byly vyhodnoceny pro databáze MNIST a CIFAR-10. Databáze MNIST byla vybrána protože se jedná o hojně používanou databázi pro testování a srovnávání různých přístupů v oblasti strojového učení[43]. Databáze MNIST obsahuje jednoduché šedotónové obrázky, které mají jednotný tvar a jednotlivé vzory, konkrétně ručně psaná čísla jsou vycentrovaná, tudíž není potřeba rozsáhlé předzpracování. Právě nenutnost rozsáhlého předzpracování[39] je také jeden z důvodů proč byl dataset vybrán. Dalším důvodem vybrání databáze MNIST je její volná dostupnost. Databáze CIFAR-10 byla zvolena protože se jedná o volně dostupnou databázi, která obsahuje více komplexní obrázky než databáze MNIST[42]. Dalším důvodem použití této databáze je fakt, že se jedná o hojně používanou databázi, která slouží pro testování a porovnávání různých přístupů v oblasti strojového učení[44].

Pro zajištění reprodukovatelnosti experimentů bylo zajištěno, že se operace v Tensorflow budou chovat deterministicky. Dále bylo ovlivněno generování náhodných čísel, nastavením tzv. seedu, tak aby při opakovaném spuštění jednotlivých experimentů bylo dosaženo vždy stejných výsledků. Také bylo vyzpozorováno, že i zvolená výpočetní jednotka má vliv na reproduktibilitu experimentů. Experimenty byly realizovány na T4 GPU.

## 7.2 Předzpracování datasetu

Načtení datasetů bylo provedeno pomocí knihovny Keras. Databáze MNIST se 70 000 obrázky je již rozdělena na trénovací a testovací sadu. Trénovací sada je dále rozdělena na validační a trénovací sadu. Výsledné rozdělení je následující: validační sada obsahuje 6 000 obrázků, trénovací sada 54 000 obrázků a sada testovací 10 000 obrázků.

Databáze CIFAR-10 obsahuje 50 000 trénovacích obrázků a 10 000 obrázků testovacích. Trénovací sada je dále rozdělena na validační a trénovací sadu. Výsledné rozdělení je následující: validační sada obsahuje 5 000 obrázků, trénovací sada obsahuje 45 000 obrázků a testovací sada obsahuje 10 000 obrázků. Jednotlivé zařazení obrázků do třídy je označeno číslicemi od 0 do 9. Hodnoty v takovémto formátu jsou převedeny na formát one-hot encoding viz kapitola 2.5. Dále proběhla normalizace obrázků vydělením jednotlivých pixelových hodnot 255, tím získáme nový rozsah hodnot pixelů v rozmezí od 0 do 1. Tímto krokem zlepšíme výpočetní náročnost.

## 7.3 Použité neuronové sítě

### 7.3.1 Popis architektur sítí

#### Mělká vícevrstvá perceptronová síť

Byla vytvořena mělká vícevrstvá perceptronová síť, která obsahuje jednu vstupní vrstvu, jednu vrstvu skrytou a jednu vrstvu výstupní. Bližší detaily o konfiguraci lze vidět v tabulce 7.1

*Tab. 7.1: Popis jednotlivých vrstev v navrhnuté mělké vícevrstvé perceptronové síti. První sloupec udává číslo vrstvy, druhý sloupec udává typ vrstvy a použitou aktivační funkci a třetí sloupec udává vlastnosti vrstvy.*

Číslo vrstvy	Typ vrstvy	Vlastnosti vrstvy
1	Vstupní vrstva	Udává rozměry vstupní vrstvy
2	Plně propojené neurony – ReLU	Počet neuronů ve vrstvě: 256
3	Plně propojené neurony – Softmax	Počet neuronů ve vrstvě: 10

#### Hluboká vícevrstvá perceptronová síť

Navržená hluboká vícevrstvá perceptronová síť, obsahuje jednu vstupní vrstvu, 8 vrstev skrytých a jednu vrstvu výstupní. Bližší detaily o konfiguraci lze vidět v tabulce 7.2

Tab. 7.2: Popis jednotlivých vrstev v navržené hluboké vícevrstvé perceptronové síti. První sloupec udává číslo vrstvy, druhý sloupec udává typ vrstvy a použitou aktivační funkci a třetí sloupec udává vlastnosti vrstvy.

Číslo vrstvy	Typ vrstvy	Vlastnosti vrstvy
1	Vstupní vrstva	Udává rozměry vstupní vrstvy
2	Plně propojené neurony – ReLU	Počet neuronů ve vrstvě: 256
3	Plně propojené neurony – ReLU	Počet neuronů ve vrstvě: 256
4	Plně propojené neurony – ReLU	Počet neuronů ve vrstvě: 512
5	Plně propojené neurony – ReLU	Počet neuronů ve vrstvě: 1024
6	Plně propojené neurony – ReLU	Počet neuronů ve vrstvě: 512
7	Plně propojené neurony – ReLU	Počet neuronů ve vrstvě: 256
8	Plně propojené neurony – ReLU	Počet neuronů ve vrstvě: 128
9	Plně propojené neurony – ReLU	Počet neuronů ve vrstvě: 64
10	Plně propojené neurony – Softmax	Počet neuronů ve vrstvě: 10

### Jednoduchá konvoluční neuronová síť

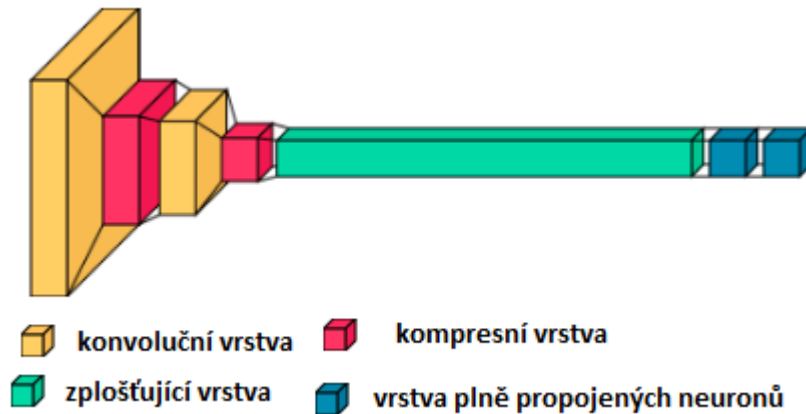
Byla vytvořena jednoduchá konvoluční neuronová síť. Tato síť má jednu vstupní vrstvu. Dále má dvě konvoluční vrstvy, dvě vrstvy kompresní, které vybírají maximum („max pooling“), a dvě vrstvy plně propojených neuronů. vizualizace architektury, která byla vytvořena pomocí balíčku vizualkeras lze vidět na obrázku 7.1 . Tabulka shrnující architekturu lze vidět v tabulce 7.3.

Tab. 7.3: Popis architektury navržené konvoluční neuronové sítě. První sloupec udává číslo vrstvy, druhý sloupec udává typ vrstvy a případně použitou aktivační funkci a třetí sloupec udává vlastnosti vrstvy.

Číslo vrstvy	Typ vrstvy	Vlastnosti vrstvy
1	Vstupní vrstva	Udává rozměry vstupních dat
2	Konvoluční 2D – ReLU	Počet filtrů: 32, Velikost filtrů: 3×3
2	Kompresní 2D – maximum	Velikost kompresních filtrů: 2×2
3	Konvoluční vrstva – ReLU	Počet filtrů: 64, Velikost filtrů: 3×3
4	Kompresní 2D – maximum	Velikost kompresních filtrů: 2×2
5	Zplošťující vrstva	Převedení na jednorozměrný obrázek
6	Plně propojené neurony – ReLU	Počet neuronů ve vrstvě: 64
7	Plně propojené neurony – Softmax	Počet neuronů ve vrstvě: 10

Posun („stride“) konvoluční vrstvy byl nastaven na hodnotu 1. Posun kompresní vrstvy („stride“) byl nastaven na hodnotu 2. Kompresní vrstva ve čtverci hledala





Obr. 7.1: Obrázek architektury navrhnuté konvoluční neuronové sítě, který byl vytvořen pomocí balíčku vizualkeras. Jednotlivé vrstvy sítě jsou barevně odlišeny.

maximum („max pooling“). Vyplnění okrajů obrázků nulami byl u kompresních a konvolučních vypnut („padding – valid“).

## ResNet18

ResNet18 představuje hlubokou konvoluční neuronovou síť. Číslo 18 značí počet hlavních vrstev. ResNet se odlišuje od klasické konvoluční sítě tím, že obsahuje propojení i mezi vrstvami, které na sebe přímo nenavazují. Tato spojení umožňují vyhnout se problému mizejícího gradientu.[45]. Kód pro tuto neuronovou síť byl přebrán z [46] a přetvořen pomocí ChatGPT [47]. Vizualizace zobrazující architekturu ResNet18 lze vidět na obrázku 7.2

První konvoluční vrstva obsahuje 64 filtrů, které mají rozměry  $7 \times 7$ , posun filtru je nastaven na hodnotu 2, doplnění obrázku nulami bylo zapnuto („padding – same“). Následuje vrstva kompresní („pooling – max“), která má rozměry filtru  $2 \times 2$ , také zde se využívá doplnění obrázku nulami. ResNet18 také obsahuje normalizační vrstvy („batch normalisation“). Síť je dále tvořena bloky, které tvoří základní stavební prvky sítě. V bloku 1 jsou dvě konvoluční vrstvy. První konvoluční vrstva bloku 1 má nastaven posun na hodnotu 2, posun druhé konvoluční vrstvy je nastaven na hodnotu 1. Rozměry filtru obou konvolučních vrstev v bloku 1 jsou  $3 \times 3$ . Obě konvoluční vrstvy mají stejný počet filtrů.

Bloku 2 obsahuje stejné dvě konvoluční vrstvy jako blok 1. Navíc obsahuje třetí konvoluční vrstvu, která má nastaven posun filtru na hodnotu 2, rozměry filtru jsou  $1 \times 1$  a doplnění obrázku nulami je zapnuto. Počet filtrů všech tří konvolučních vrstev je stejný.

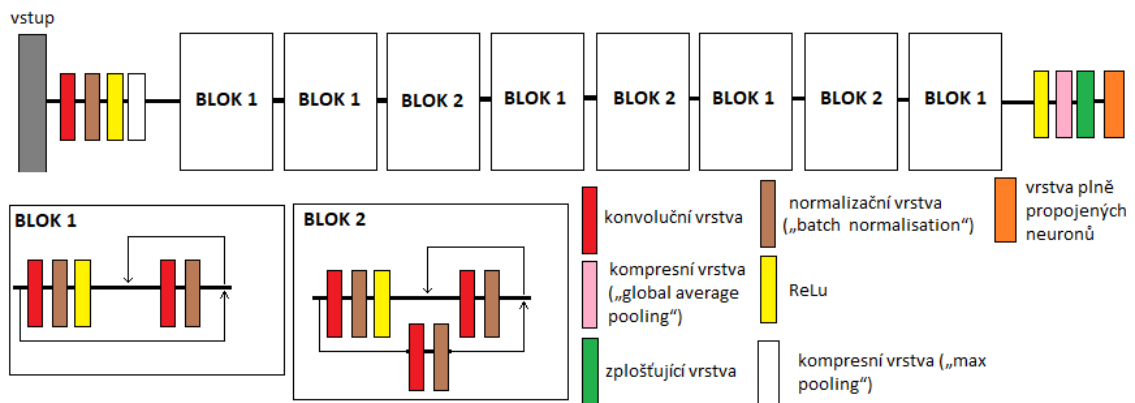
Na konci sítě, je vrstva kompresní („pooling – global average“) a vrstva plně

Tab. 7.4: Popis počtu filtrů konvolučních vrstev v jednotlivých blocích tvořící základ sítě ResNet18. První sloupec vyjadřuje pořadí bloku v síti. Druhý sloupec vyjadřuje typ bloku, který se používá pro tvorbu sítě typu ResNet a poslední sloupec vyjadřuje počty filtrů pro jednotlivé konvoluční vrstvy v jednotlivých blocích.

Pořadí bloku	Typ bloku	Počet filtrů
1	Blok 1	64
2	Blok 1	64
3	Blok 2	128
4	Blok 1	128
5	Blok 2	256
6	Blok 1	256
7	Blok 2	512
8	Blok 1	512

propojených neuronů. Počet neuronů ve vrstvě je 10, jako aktivační funkce je použita softmax.

Jako aktivační funkce všech konvolučních vrstev byla použita ReLU. Počet filtrů konvolučních vrstev v jednotlivých blocích závisí na pořadí bloku v síti. Jednotlivé počty filtrů v blocích lze vidět v tabulce 7.4. Vizualizaci bloků 1 a 2 lze vidět na obrázku 7.2.



Obr. 7.2: Obrázek architektury ResNet18, který zobrazuje poskládání jednotlivých vrstev a bloků. Na obrázku lze vidět blok 1 a blok 2, které představují základní stavební jednotky sítě typu ResNet. V pravé dolní části obrázku lze vidět popisky jednotlivých barevně odlišených prvků sítě.

### 7.3.2 Hyperparametry sítí

V následující kapitole budou popsány hyperparametry, které byly zvoleny pro všechny výše zmíněné architektury stejné.

Jako optimalizační algoritmus je zvolen Adam, kvůli zlepšení učení a snadnějšímu nalezení optimálního řešení oproti SGD[13, 14]. Pro Adam byl zvolen podle doporučení z literatury [48] krok učení roven 0,001. Jelikož převádíme požadované hodnoty one-hot kódováním, chybová funkce je zvolena na kategoričnou vzájemnou entropii. Batch size byl zvolen 64. Počet epoch učení byl stanoven na 200, ale byla zavedena funkce early stopping, která v průběhu trénování sledovala hodnotu chyby na validační sadě. Pokud se hodnota po dobu 10 následujících epoch, po zaznamenání nejnižší dosavadní hodnoty chyby, nesníží, učení se ukončí. Tímto zajistíme to, že se síť nepřeučí. Důležité je také zmínit, že po takovém ukončení se váhy nastaví na hodnoty, při kterých byla dosažena nejnižší hodnota chyby na validační sadě. Počáteční inicializace biasu je zvolena na hodnotu 0.

## 7.4 Implementace jednotlivých experimentů

### Úvodní experiment – inicializace vah na hodnotu 0

Základní trénování mělké vícevrstvé perceptronové sítě proběhlo s metodou inicializace vah na hodnotu 0. Inicializace byla implementovaná pomocí knihovny Keras a Tensorflow. Jedinou změnou byla změna počtu epoch, která byla stanovena na 10. Tato hodnota byla stanovena experimentálně po zkoušce v Google Colabu.

### LSUV

Kód pro implementaci metody LSUV byl přebrán z [49]. Přebraný kód, počítal s použitím pouze pro 2D konvoluční vrstvy a plně propojené vrstvy neuronů. Tato skutečnost nám vyhovovala, protože jiné trénovatelné vrstvy v námi zvolených sítích nebyly. Proběhla malá úprava kódu. První úprava byla úprava navrhnutého příkladu spouštění funkce LSUVinit, která předpokládala 4 rozměrný vstup jakožto argument funkce batch size a to při načítání obrázků, které jsou z databáze MNIST nebylo splněno. Druhá úprava byla přidání řádku, který ovlivňoval generování náhodných čísel.

U této hybridní metody byl zkoumán celkový vliv na učení sítí, ale také jsme prozkoumali vliv počtu obrázků, které jsou používány pro získání aktivace jednotlivých vrstev a následné škálování vah. Zvolené počty použitých obrázků byly 2, 16, 64, 256 a 1024.

Pro naši LSUV inicializaci byl zvolen počet maximálních průchodů obrázků vrstvou na 10 iterací, dle implementace z [49]. Minimální hodnota požadovaného rozptylu výstupu vrstvy pro ukončení škálování vah byla nastavena na hodnotu 0,995.

### **Náhodná inicializace vah z normálního a rovnoměrného pravděpodobnostního rozdělení**

Náhodná inicializace vah z normálního a rovnoměrného pravděpodobnostního rozdělení byla implementovaná pomocí knihoven Tensorflow a Keras. V podkapitole 3.1.1 a 3.1.1 byly zmíněny metody využívající k získání počátečních vah vzorkování z normálního nebo rovnoměrného pravděpodobnostního rozdělení. Předmětem zkoumání bude vliv změny konfigurace parametrů pravděpodobnostního rozdělení na učení sítě. U normálního pravděpodobnostního rozdělení byl průměr nastaven na 0 a směrodatná odchylka byla brána z hodnot: 1; 0,9; 0,8; 0,7; 0,6; 0,5; 0,4; 0,3; 0,2; 0,1; 0,05; 0,005; 0,0005. Pro rovnoměrné pravděpodobnostní rozdělení budeme volit z hodnot: 1; 0,9; 0,8; 0,7; 0,6; 0,5; 0,4; 0,3; 0,2; 0,1; 0,05; 0,005; 0,0005 limit pravděpodobnostního rozdělení `<-limit,limit>`. U těchto metod byl zkoumán celkový vliv na učení sítí, ale také byl prozkoumán vliv různých konfigurací na učení sítě.

### **Ostatní metody náhodné inicializace**

Metody byly implementovány pomocí knihoven Tensorflow a Keras. Vybrané metody jsou LeCun, He a Glorot. Metody reprezentují metody využívající náhodné inicializace vah založené na škálování rozptylu. Tyto metody byly použity ve dvou verzích a to verzi používající vzorkování vah z normálního pravděpodobnostního rozdělení a verzi používající vzorkování vah z rovnoměrného pravděpodobnostního rozdělení. Díky tomu můžeme sledovat rozdíly, které jsou zvoleným rozdělením způsobeny. Dále do této kategorie byla přidána inicializace vah pomocí rozkladu na ortonormální bázi. Všechny čtyři metody mají základ v náhodném výběru hodnot z pravděpodobnostního rozdělení, a proto byly dané do jedné kategorie.

## **7.5 Statistické vyhodnocení výsledků**

Pro statistické vyhodnocení výsledků byly použity navržené metriky v kapitole 4.

K ohodnocení rychlosti naučení bude sloužit počet epoch k dosažení optimálního naučení sítě. Optimální naučení sítě je zajištěno implementací funkce early stopping. Po dosažení optimálního naučení sítě byla síť ohodnocena na testovací sadě. Metriky které byly použity pro hodnocení toho, jak dobře se síť naučila jsou správnost (accuracy), přesnost (precision), výtěžnost (recall) a F1-skóre. Přesnost,

výtěžnost a F1-skóre pro jednotlivé třídy a jejich vážený průměr jsme dostali pomocí funkce `classification_report` z knihovny `scikit-learn`[50]. Hodnoty správnosti jsou násobeny 100 a uváděny ve výsledcích v procentech. Matici záměn pro vizualizaci predikcí modelu jsme dostali pomocí funkce `classification_report` z knihovny `scikit-learn`[50]. Zvolené metriky jsou získány pro jednotlivé typy sítí a jednotlivé metody inicializace s různými konfiguracemi. Nejlepší výsledky pro jednotlivé typy sítí a jednotlivé kategorie metod inicializace budou porovnány mezi sebou. Dalším ukazatelem rychlosti a toho jak dobře se síť učí je závislost počtu epoch k dosažení určité chyby a správnosti na validační sadě.

Hodnoty chyby a správnosti na trénovací a validační sadě byly v průběhu trénování ukládány do proměnných. Pomocí těchto proměnných docházelo k vykreslování jednotlivých průběhů chyb a správnosti v závislosti na počtu epoch na validační a trénovací sadě. Pro grafické porovnání metod v jedné kategorii nebo jednotlivých konfigurací metod byly vytvořeny 4 grafy. Průběhy jsou zobrazeny do epochy optimálního naučení. Funkce, která tvořila grafy byla vyhotovena pomocí ChatGPT[51].

První graf zobrazuje průběhy správnosti na trénovací sadě. Druhý graf zobrazuje průběhy chyby na trénovací sadě. Třetí graf zobrazuje průběhy správnosti na validační sadě. Čtvrtý graf zobrazuje průběhy chyby na validační sadě.

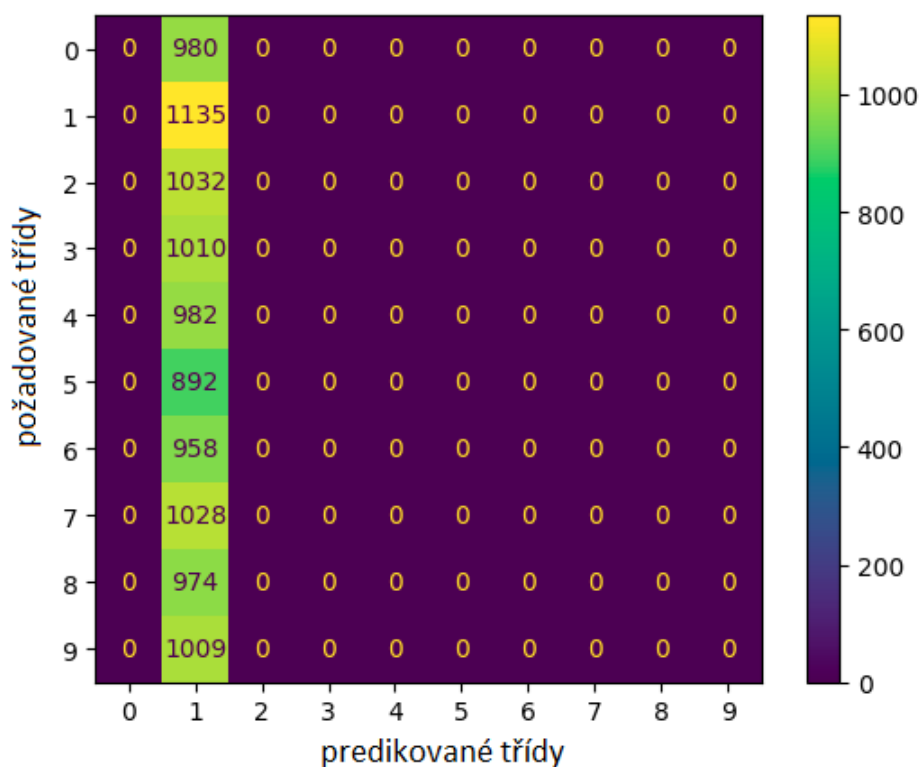
## 8 Výsledky a diskuse

### 8.1 Výsledky – databáze MNIST

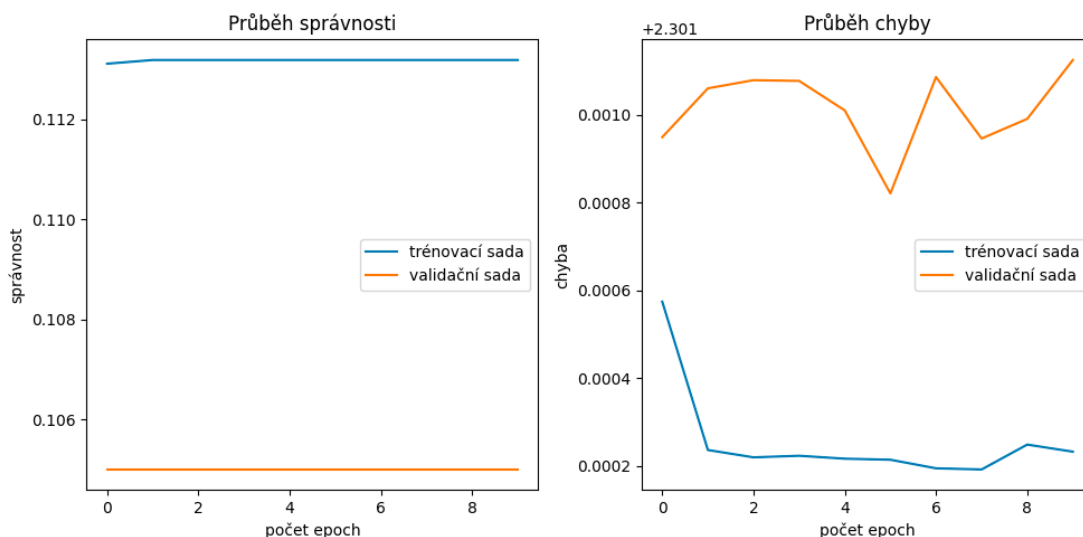
#### 8.1.1 Mělká vícevrstvá perceptronová síť

##### Úvodní experiment – inicializace vah na hodnotu 0

Výsledky získané po trénování mělké vícevrstvé perceptronové sítě potvrdily tvrzení z [1], že inicializace vah na hodnotu 0 není vhodná. Tento fakt potvrzuje i nízká dosažená správnost na testovací sadě a to 11,350 %. Takto inicializovaná síť zařadila veškeré obrázky z testovací sady do jedné třídy. Tuto skutečnost lze vidět na obrázku 8.1, na kterém je zobrazena matice záměn pro testovací sadu. Průběh učení sítě a hodnoty trénovací a validační správnosti a chyby v závislosti na počtu epoch lze vidět na obrázku 8.2. Z průběhů na obrázku 8.2, lze vidět, že se síť v průběhu epoch nezlepšuje. Hodnoty správnosti po učení, vážená přesnost, F1-skóre a výtěžnost pro testovací sadu, lze vidět v tabulce 8.1,



Obr. 8.1: Matice záměn získaná pro testovací sadu z databáze MNIST po inicializaci mělké vícevrstvé perceptronové sítě nulami. Osa  $x$  vyjadřuje predikované třídy sítě a osa  $y$  vyjadřuje požadované třídy. V pravé části obrázku je barevně označená škála počtu zařazených obrázků.



Obr. 8.2: Průběhy správností a chyby na validační a trénovací sadě v závislosti na počtu epoch, které byly získány inicializací vah mělké vícevrstvé perceptronové sítě nulami a jejím následným učením na trénovací sadě databáze MNIST.

Tab. 8.1: Popis metrik ohodnocující učení mělké vícevrstvé perceptronové sítě, která byla inicializovaná nulami a poté učena na trénovací sadě databáze MNIST. Hodnoty jsou získány pro testovací sadu databáze MNIST.

správnost po trénování	11,349 %
vážený průměr přesnosti.	0,010
vážený průměr výtěžnosti	0,110
vážený průměr F1-skóre.	0,020

## LSUV

Výsledky v tabulce A.1 vyjadřují ohodnocení rychlosti učení, ale také toho jak dobře se síť naučila. Nejlepšího výsledku správnosti na testovací sadě, konkrétně 98,030 % dosáhla síť, která používala k inicializaci vah LSUV 2 obrázky z trénovací sady. Na druhou stranu síť s touto konfigurací dosáhla optimálního naučení nejpomaleji ze všech konfigurací.

Druhá nejlepší síť, inicializovaná metodou LSUV pomocí 256 obrázku z trénovací sady, dokázala dosáhnout o něco menší správnosti na testovací sadě, konkrétně 97,760 %, ale za menší počet epoch. Naopak nejhoršího výsledku správnosti na testovací dosáhla síť inicializovaná konfigurací LSUV používající 64 obrázků z trénovací sady. Výsledky průměrného váženého F1-skóre, průměrné vážené přesnosti a výtěžnosti jsou pro testovací sadu u všech konfigurací metody stejné.

Rychlost učení, a to jak dobře se síť dokázala naučit, lze také pozorovat v tabulce

A.2, která vyjadřuje počet epoch k dosažení 85 %, 90 %, 95 %, 96 %, 97 %, 98 % a 99 % správnosti na validační sadě pro jednotlivé konfigurace LSUV. Výsledky potvrzují, že nejlépe se naučila síť používající konfiguraci LSUV s 2 obrázky. Tato síť dokázala dosáhnout 98 % správnosti na validační sadě a potřebovala k tomu 8 epoch. Síť s konfigurací LSUV s 64 obrázky a 256 obrázky nedosáhla vyšší správnosti na validační sadě než 97 %. Jako jediná síť, která nedokázala dosáhnout 97 % správnost na validační sadě za 1 epochu byla síť s konfigurací LSUV s 2 obrázky. Průběhy chyby a správnosti na validační a trénovací sadě pro 5 konfigurací metody lze vidět na obrázku A.1. Průběhy jsou zobrazeny do epochy optimálního naučení.

### **Náhodná inicializace vah z normálního a rovnoměrného pravděpodobnostního rozdělení**

Výsledky pro síť používající různé konfigurace metody inicializace vah z normálního pravděpodobnostního rozdělení, jsou vidět v tabulce A.3. Nejvyšší správnost na testovací sadě byla dosažena pomocí konfigurace normálního rozdělení ( $\mu = 0, \sigma = 0,05$ ). Díky této konfiguraci byla síť schopna dosáhnout 98,100 % správnosti na testovací sadě. K dosažení optimálního naučení potřebovala síť s touto konfigurací 11 epoch. Druhou nejlepší správnost získala síť s konfigurací normálního rozdělení ( $\mu = 0, \sigma = 0,1$ ), konkrétně 98,020 %. Této síti k dosažení optimálního naučení stačilo 10 epoch. Naopak nejvíce epoch, konkrétně 46, k dosažení optimálního naučení potřebovala síť využívající konfigurace normálního rozdělení ( $\mu = 0, \sigma = 0,8$ ). Nejmenší počet epoch k dosažení optimálního naučení sítě dosáhly dvě konfigurace normálního rozdělení ( $\mu = 0, \sigma = 0,005$ ) a ( $\mu = 0, \sigma = 0,2$ ) V tabulce A.3 si lze také všimnout tendence zvyšování dosažené průměrné vážené přesnosti, výtěžnosti a F1-skóre se snižováním hodnoty směrodatné odchylky  $\sigma$ . Z tabulky A.4, která vyjadřuje počet epoch k dosažení 85 %, 90 %, 95 %, 96 %, 97 %, 98 % a 99 % správnosti na validační sadě, vyplývá, že správnosti 98 % dosáhly jen 4 konfigurace. Použité konfigurace mají směrodatné odchylky rovny hodnotám 0,1; 0,05; 0,005 a 0,0005. Této správnosti síť dosáhla se všemi těmito konfiguracemi za 7 epoch.

Výsledné hodnoty pro síť využívající různé konfigurace metody inicializace vah z rovnoměrného pravděpodobnostního rozdělení jsou vidět v tabulce A.5. Nejvyšší správnost na testovací sadě dosáhla konfigurace rovnoměrného pravděpodobnostního rozdělení  $\langle -0,005; 0,005 \rangle$ . Dosažená správnost má hodnotu 97,960 %. Síť s touto konfigurací rovnoměrného rozdělení dosáhla optimálního naučení za 8 epoch. Tato hodnota je druhá nejlepší ze všech konfigurací. Rychleji dosáhla optimálního naučení už jen síť s konfigurací rovnoměrného rozdělení  $\langle -0,05; 0,05 \rangle$ . V tabulce A.5 si lze také všimnout, že od konfigurace  $\langle -0,3; 0,3 \rangle$  se zvýšila hodnota průměrné vážené přesnosti, výtěžnosti a F1-skóre z 0,97 na hodnotu 0,98. V tabulce A.6, vyjadřující



počet epoch k dosažení 85 %, 90 %, 95 %, 96 %, 97 %, 98 % a 99 % správnosti na validační sadě, lze pozorovat 7 konfigurací rovnoměrného pravděpodobnostního rozdělení, které dosáhly 98 % správnost. Použité konfigurace mají hodnotu limitu rovnou 0,4, 0,3, 0,2, 0,1, 0,05, 0,005 a 0,0005. (<-limit,limit>).

Porovnáním výsledků v tabulce A.3 a A.5 lze zjistit, že celkové nejlepší správnosti na testovací sadě dosáhla síť používající metodu inicializace vah z normálního pravděpodobnostního rozdělení s průměrem rozdělení rovným 0 a směrodatnou odchylkou rovnou 0,05. Celkově lze také vypočítat u obou přístupů zvyšující se hodnoty správnosti na testovací sadě se snižující se hodnotou limitu u rovnoměrného pravděpodobnostního rozdělení nebo směrodatné odchylky u normálního pravděpodobnostního rozdělení. Průběhy chyb a správností, pro síť využívající různé konfigurace metod inicializace vah z rovnoměrného a normálního rozdělení, na validační a trénovací sadě lze vidět na obrázku A.2. Průběhy jsou zobrazeny do epochy optimálního naučení.

### **Ostatní metody náhodné inicializace**

Výsledné hodnoty pro síť využívající ostatní metody náhodné inicializace a jejich případné konfigurace jsou zaznamenány v tabulce A.7. Nejvyšší hodnoty správnosti na testovací sadě dosáhla síť používající metodu inicializace vah He z normálního pravděpodobnostního rozdělení. Tato síť dosáhla 98,140 % správnosti. K dosažení optimálního naučení sítě bylo potřeba 11 epoch. Nejrychleji dosáhla síť, používající metodu inicializace vah Glorot z rovnoměrného pravděpodobnostního rozdělení, optimálního naučení za 7 epoch. Síť využívající k inicializaci vah rozklad na ortonormální bázi dosáhla optimálního naučení za 8 epoch. Hodnota průměrné vážené přesnosti, výtěžnosti a F1-skóre jsou u všech metod stejné.

V tabulce A.8, vyjadřující počet epoch k dosažení 85 %, 90 %, 95 %, 96 %, 97 %, 98 % a 99 % správnosti na validační sadě, je možno vidět, že síť se všemi metodami dosáhla 98 % správnosti na validační sadě. Nejrychleji bylo této správnosti dosaženo za 5 epoch a to sítí s inicializací vah pomocí rozkladu na ortonormální bázi.

Porovnáním výsledků v tabulce A.7 pro mělkou vícevrstvou perceptronovskou síť využívající metody LeCun, He a Glorot z rozdílných pravděpodobnostních rozdělení vychází vždy s lepší dosaženou správností na testovací sadě metody využívající normální rozdělení.

Průběhy chyby a správnosti na validační a trénovací sadě, pro mělkou vícevrstvou perceptronovskou síť, využívající metody inicializace vah LeCun, He, Glorot v různých konfiguracích a metodu inicializace vah využívající rozklad na ortonormální bázi lze vidět na obrázku A.3. Průběhy jsou zobrazeny do epochy optimálního naučení.

## 8.1.2 Hluboká vícevrstvá perceptronovská síť

### LSUV

V tabulce A.9 lze vidět výsledky, pro síť používající metodu inicializace vah LSUV v různých konfiguracích. Nejlepšího výsledku správnosti na testovací sadě dosáhla síť používající k inicializaci vah LSUV používající 16 obrázků z trénovací sady. Dosažená správnost je 97,690 %. Tato síť dosáhla optimálního naučení za 8 epoch, což je druhý nejhorší výsledek.

Nejrychleji optimálního naučení dosáhla síť inicializovaná metodou LSUV pomocí 1024 obrázků. Tato síť potřebovala pouze 3 epochy. Na druhou stranu, síť dosáhla nejhorší správnosti na testovací sadě, konkrétně 97,180 %. Nejpomaleji dosáhla optimálního naučení síť využívající konfiguraci LSUV používající 256 obrázků z trénovací sady. Síť potřebovala 11 epoch, ale dosáhla druhé nejlepší správnosti na testovací sadě. Nejlepší výsledky vážené přesnosti, výtěžnosti a váženého F1-skóre dosáhla síť využívající konfiguraci inicializace LSUV s 256 a s 16 obrázky.

V tabulce A.10, vyjadřující počet epoch potřebných k dosažení 85 %, 90 %, 95 %, 96 %, 97 %, 98 % a 99 % správnosti na validační sadě pro jednotlivé konfigurace metody LSUV, lze vidět, že všechny konfigurace LSUV dosáhly 97 % správnosti na validační sadě. Nejpomaleji této správnosti dosáhla síť využívající konfiguraci LSUV používající 16 obrázků z trénovací sady. Konfigurace LSUV s 64 a 256 obrázky jako jediné dosáhly 98 % správnosti na validační sadě. Nejrychleji této správnosti dosáhla s 12 epochami síť s konfigurací LSUV s 64 obrázky. Průběhy chyb a správností na validační a trénovací sadě pro 5 konfigurací metody LSUV, lze vidět na obrázku A.4. Průběhy jsou zobrazeny do epochy optimálního naučení.

### Náhodná inicializace vah z normálního a rovnoměrného pravděpodobnostního rozdělení

Výsledky pro síť využívající různé konfigurace metody inicializace vah z normálního pravděpodobnostního rozdělení můžeme vidět v tabulce A.11. Nejvyšší správnost na testovací sadě byla dosažena konfigurací normálního rozdělení ( $\mu = 0, \sigma = 0, 1$ ). Síť dosáhla 97,850 % správnosti na testovací sadě. K dosažení optimálního naučení potřebovala síť 14 epoch. Druhou nejlepší správnost 97,550 % na testovací sadě, dosáhla síť s konfigurací normálního rozdělení ( $\mu = 0, \sigma = 0, 05$ ). Síť potřebovala pouze 5 epoch k dosažení optimálního naučení, což je nejlepší výsledek pro všechny konfigurace. Nejvíce epoch potřebovala síť s konfigurací normálního rozdělení ( $\mu = 0, \sigma = 0, 8$ ). Trénování sítě se zastavilo až na maximální hodnotě 200 epoch. Nejhorší výsledky správnosti dosáhly konfigurace normálního rozdělení ( $\mu = 0, \sigma = 0, 005$ ) a ( $\mu = 0, \sigma = 0, 0005$ ). Obě konfigurace dosáhly pouze 11,350 % správnosti na

testovací sadě. Také jejich hodnoty vážené přesnosti, výtěžnosti a váženého F1-skóre jsou nejhorší ze všech konfigurací. Nejlepší výsledky vážené přesnosti, výtěžnosti a váženého F1-skóre dosáhla síť s konfigurací normálního rozdělení ( $\mu = 0, \sigma = 0, 1$ ) a ( $\mu = 0, \sigma = 0, 05$ ). V tabulce A.12, vyjadřující počet epoch potřebných k dosažení 85 %, 90 %, 95 %, 96 %, 97 %, 98 % a 99 % správnosti na validační sadě, lze vidět, že 98 % správnosti na validační sadě dosáhla síť s dvěma konfiguracemi, a to s konfigurací normálního rozdělení ( $\mu = 0, \sigma = 0, 1$ ) a ( $\mu = 0, \sigma = 0, 05$ ). Síť s konfigurací normálního rozdělení ( $\mu = 0, \sigma = 0, 05$ ), dosáhla této správnosti rychleji, za 11 epoch. Nejmenší počet epoch k dosažení 97 % správnosti na validační sadě, konkrétně 3, potřebovala síť využívající konfiguraci normálního rozdělení ( $\mu = 0, \sigma = 0, 05$ ). Naopak konfigurace normálního rozdělení ( $\mu = 0, \sigma = 0, 005$ ) a ( $\mu = 0, \sigma = 0, 0005$ ) nedosáhly žádné z definovaných správností na validační sadě.

Výsledné hodnoty pro síť využívající různé konfigurace metody inicializace vah z rovnoměrného pravděpodobnostního rozdělení lze vidět v tabulce A.13. Nejvyšší správnosti na testovací sadě, konkrétně 97,780 %, dosáhla konfigurace rovnoměrného rozdělení  $\langle -0,2; 0,2 \rangle$ . Síť potřebovala k dosažení optimálního naučení síť 12 epoch. Nejméně epoch k dosažení optimálního naučení potřebovala síť s konfigurací rovnoměrného rozdělení  $\langle -0,1; 0,1 \rangle$ . Síť k dosažení optimálního naučení stačilo 7 epoch. Nejvíce epoch k dosažení optimálního naučení potřeboval síť s konfigurací rovnoměrného rozdělení  $\langle -1,1 \rangle$ . Nejhorší výsledek dosáhla síť s konfigurací rovnoměrného rozdělení  $\langle -0,005; 0,005 \rangle$  a  $\langle -0,0005; 0,0005 \rangle$ . Dosáhla pouze 11,350 % správnosti na testovací sadě. Hodnoty jejich vážené přesnosti, výtěžnosti a váženého F1-skóre jsou nejhorší ze všech konfigurací. V tabulce A.14, vyjadřující počet epoch potřebných k dosažení 85 %, 90 %, 95 %, 96 %, 97 %, 98 % a 99 % správnosti na validační sadě, lze vidět, že 98 % správnosti na validační sadě dosáhla síť s 5 konfiguracemi rovnoměrného rozdělení. Nejméně epoch k dosažení této správnosti, konkrétně 7, potřebovala síť s konfigurací rovnoměrného rozdělení  $\langle -0,1; 0,1 \rangle$ . Naopak konfigurace rovnoměrného rozdělení  $\langle -0,005; 0,005 \rangle$  a  $\langle -0,0005; 0,0005 \rangle$  nedosáhly žádné z definovaných správností na validační sadě.

Při porovnání výsledků v tabulkách A.11 a A.13 lze zjistit, že celkové nejlepší správnosti na testovací sadě dosáhla síť používající metodu inicializace vah z normálního rozdělení s průměrem rozdělení rovným 0 a směrodatnou odchylkou rovnou 0,1. Průběhy chyby a správnosti na validační a trénovací sadě, pro síť využívající různé konfigurace metod inicializace vah z normálního a rovnoměrného pravděpodobnostního rozdělení lze vidět na obrázku A.5. Průběhy jsou zobrazeny do epochy optimálního naučení. Na obrázku A.5, lze vidět prudký pokles správnosti na validační i trénovací sadě na konci učení pro síť využívající konfigurace normálního rozdělení ( $\mu = 0, \sigma = 0, 7$ ) a ( $\mu = 0, \sigma = 0, 6$ ). U sítě využívající konfigurace rovnoměrného rozdělení jsou to konfigurace  $\langle -1,1 \rangle$  a  $\langle -0,9; 0,9 \rangle$ .

## Ostatní metody náhodné inicializace

Výsledné hodnoty pro síť využívající ostatní metody náhodné inicializace a jejich případné konfigurace jsou zaznamenány v tabulce A.15. Nejvyšší hodnoty správnosti na testovací sadě dosáhla síť používající metodu inicializace vah LeCun z normálního pravděpodobnostního rozdělení. Tato síť dosáhla 98,180 % správnosti. Síť dosáhla optimálního naučení za 15 epoch. Nejrychleji dosáhla síť, používající metodu inicializace vah He z rovnoměrného pravděpodobnostního rozdělení, optimálního naučení za 4 epoch. Síť využívající k inicializaci vah pomocí rozkladu na ortonormální bázi dosáhla optimálního naučení za 5 epoch, což je druhý nejlepší výsledek.

V tabulce A.16, vyjadřující počet epoch k dosažení 85 %, 90 %, 95 %, 96 %, 97 %, 98 % a 99 % správnosti na validační sadě, je možno vidět, že síť se všemi metodami dosáhla 98 % správnosti na validační sadě. Nejrychleji bylo této správnosti dosaženo za 5 epoch a to sítí s inicializací vah pomocí rozkladu na ortonormální bázi.

Porovnáním výsledků v tabulce A.15 pro hlubokou vícevrstvou perceptronovskou síť využívající metody LeCun, He a Glorot z rozdílných pravděpodobnostních rozdělení vychází vždy s lepší dosaženou správností na testovací sadě metody využívající normální rozdělení.

Průběhy chyby a správnosti na validační a trénovací sadě, pro hlubokou vícevrstvou perceptronovskou síť využívající metody inicializace vah LeCun, He, Glorot v různých konfiguracích a metodu inicializace vah využívající rozklad na ortonormální bázi lze vidět na obrázku A.6. Průběhy jsou zobrazeny do epochy optimálního naučení.

### 8.1.3 Jednoduchá konvoluční neuronová síť

#### LSUV

V tabulce A.17 lze vidět výsledky, pro síť používající metodu inicializace vah LSUV v různých konfiguracích. Nejlepšího výsledku správnosti na testovací sadě dosáhla síť využívající k inicializaci vah LSUV používající 1024 obrázky z trénovací sady. Síť dosáhla správnosti 98,930 %. Síť dosáhla optimálního naučení za 3 epochy. Tento výsledek je druhý nejlepší.

Nejrychleji dosáhla síť optimálního naučení s konfigurací LSUV využívající 64 obrázků z trénovací sady. Tato síť potřebovala pouze 2 epochy, ale dosáhla nejhorší správnosti na testovací sadě ze všech konfigurací. Nejpomaleji dosáhla optimálního naučení síť využívající konfigurace LSUV používající 16 a 256 obrázků z trénovací sady. Síť s oběma konfiguracemi potřebovala 4 epochy. Výsledky průměrné vážené přesnosti, správnosti a váženého F1-skóre jsou pro všechny konfigurace stejné.

V tabulce A.18, vyjadřující počet epoch potřebných k dosažení 85 %, 90 %, 95 %, 96 %, 97 %, 98 % a 99 % správnosti na validační sadě, lze vidět, že síť se všemi konfiguracemi LSUV dosáhla 99 % správnosti na validační sadě. Nejrychleji této správnosti dosáhla síť s konfigurací LSUV využívající 2 obrázky z trénovací sady. Nejpomaleji této správnosti dosáhla síť využívající konfiguraci LSUV používající 64 obrázků z trénovací sady. Průběhy chyb a správností na validační a trénovací sadě pro 5 konfigurací metody LSUV, lze vidět na obrázku A.7. Průběhy jsou zobrazeny do epochy optimálního naučení.

### **Náhodná inicializace vah z normálního a rovnoměrného pravděpodobnostního rozdělení**

Výsledky pro síť využívající různé konfigurace metody inicializace vah z normálního pravděpodobnostního rozdělení můžeme vidět v tabulce A.19. Nejvyšší správnost na testovací sadě, konkrétně 99,080 %, byla dosažena konfigurací normálního rozdělení ( $\mu = 0, \sigma = 0, 1$ ). K dosažení optimálního naučení síť potřebovala 5 epoch. Druhá nejlepší správnost 98,900 % na testovací sadě, byla dosažena sítí s konfigurací normálního rozdělení ( $\mu = 0, \sigma = 0, 2$ ). Síť potřebovala také 5 epoch k dosažení optimálního naučení. Nejvíce epoch k dosažení optimálního naučení, konkrétně 54, potřebovala síť s konfigurací normálního rozdělení ( $\mu = 0, \sigma = 0, 9$ ). V tabulce A.19 lze pozorovat nejvyšší hodnoty vážené přesnosti, výtěžnosti a váženého F1-skóre pro síť využívající konfiguraci normálního rozdělení s průměrem rovným 0 a se směrodatnými odchylkami menšími nebo rovnými 0,4.

V tabulce A.20, vyjadřující počet epoch potřebných k dosažení 85 %, 90 %, 95 %, 96 %, 97 %, 98 % a 99 % správnosti na validační sadě, lze pozorovat, že 99 % správnosti na validační sadě dosáhla pouze síť s konfiguracemi normálního rozdělení ( $\mu = 0, \sigma = 0, 1$ ) a ( $\mu = 0, \sigma = 0, 05$ ). Nejméně epoch k dosažení 99 % správnosti na validační sadě, konkrétně 4, potřebovala síť využívající konfiguraci normálního rozdělení ( $\mu = 0, \sigma = 0, 1$ ). 98 % správnosti na validační sadě dosáhla síť se všemi konfiguracemi normálního rozdělení. Nejrychleji této správnosti dosáhla síť využívající konfiguraci normálního rozdělení ( $\mu = 0, \sigma = 0, 1$ ). Síť potřebovala jen 1 epochu k dosažení této správnosti.

Výsledné hodnoty pro jednoduchou konvoluční neuronovou síť využívající různé konfigurace metody inicializace vah z rovnoměrného pravděpodobnostního rozdělení, lze vidět v tabulce A.21. Nejvyšší správnosti na testovací sadě, konkrétně 99,110 %, dosáhla konfigurace rovnoměrného rozdělení  $\langle -0,05; 0,05 \rangle$ . K dosažení optimálního naučení této sítě bylo potřeba 11 epoch. Nejméně epoch k dosažení optimálního naučení potřebovala síť s konfigurací rovnoměrného rozdělení  $\langle -0,2; 0,2 \rangle$ . Síť k dosažení optimálního naučení potřeboval pouze 5 epoch. Tato síť také dosáhla třetí nejlepší

správnosti na testovací sadě. Nejvíce epoch potřebovala k dosažení optimálního naučení síť s konfigurací rovnoměrného rozdělení  $\langle -1,1 \rangle$ . V tabulce A.22, vyjadřující počet epoch potřebných k dosažení 85 %, 90 %, 95 %, 96 %, 97 %, 98 % a 99 % správnosti na validační sadě, lze vidět, že 99 % správnosti na validační sadě dosáhla síť s 6 různými konfiguracemi rovnoměrného rozdělení. Nejméně epoch k dosažení 99 % správnosti na validační sadě potřebovala síť s konfigurací rovnoměrného rozdělení  $\langle -0,2;0,2 \rangle$ . Tato síť potřebovala pouze 3 epochy.

Při porovnání výsledků v tabulkách A.19 a A.21 lze zjistit, že celkového nejlepšího výsledku správnosti na testovací sadě dosáhla síť používající metodu inicializace vah z rovnoměrného rozdělení s limitem rovným 0,05 ( $\langle -0,05;0,05 \rangle$ ). Průběhy chyby a správnosti na validační a trénovací sadě, pro síť využívající různé konfigurace metod inicializace vah z normálního a rovnoměrného pravděpodobnostního rozdělení lze vidět na obrázku A.8. Průběhy jsou zobrazeny do epochy optimálního naučení.

### **Ostatní metody náhodné inicializace**

Výsledné hodnoty pro síť využívající ostatní metody náhodné inicializace a jejich případné konfigurace jsou zaznamenány v tabulce A.23. Síť inicializovaná metodou rozkladu na ortonormální bázi a Glorot vzorkující váhy z normálního pravděpodobnostního rozdělení dosáhla nejvyšší správnosti na testovací sadě. Síť s oběma konfiguracemi dosáhla 99,190 % správnosti na testovací sadě. Při porovnání počtu epoch potřebných pro dosažení optimálního naučení u těchto dvou metod pomyslně vyhrává metoda Glorot vzorkující váhy z normálního pravděpodobnostního rozdělení, protože síť používající tuto metodu dosáhla optimálního naučení za 5 epoch. Zato síť využívající k inicializaci vah rozklad na ortonormální bázi dosáhla optimálního naučení až za 8 epoch. V tabulce A.24, vyjadřující počet epoch k dosažení 85 %, 90 %, 95 %, 96 %, 97 %, 98 % a 99 % správnosti na validační sadě, je možno vidět, že síť se všemi metodami dosáhla 99 % správnosti na validační sadě. Nejrychleji bylo této správnosti dosaženo za 4 epochy a to sítí s inicializací vah metodou LeCun v obou konfiguracích a metodou Glorot také v obou konfiguracích.

Hodnoty průměrné vážené přesnosti, správnosti a průměrného váženého F1-skóre jsou pro všechny konfigurace stejné.

Průběhy chyby a správnosti na validační a trénovací sadě, pro hlubokou vícevrstvou perceptronovskou síť využívající metody inicializace vah LeCun, He, Glorot v různých konfiguracích a metodu inicializace vah využívající rozklad na ortonormální bázi lze vidět na obrázku A.9. Průběhy jsou zobrazeny do epochy optimálního naučení.

## 8.1.4 Hluboká konvoluční neuronová síť – ResNet18

### LSUV

V tabulce A.25 lze vidět výsledky, pro síť používající metodu inicializace vah LSUV v různých konfiguracích. Nejvyšší správnost na testovací sadě dosáhla síť používající konfiguraci LSUV s 16 obrázky z trénovací sady. Tato síť dosáhla správnosti 99,220 %. Optimálního naučení dosáhla za 16 epoch.

Nejméně epoch k dosažení optimálního naučení potřebovala síť využívající konfiguraci LSUV se 2 obrázky a s 64 obrázky z trénovací sady. Síť s oběma konfiguracemi potřebovala jen 9 epoch. Naopak nejvíce epoch, konkrétně 19, potřebovala síť s konfigurací LSUV využívající 1024 obrázků z trénovací sady. Výsledky vážené přesnosti, výtěžnosti a váženého F1-skóre jsou pro všechny konfigurace stejné.

V tabulce A.26, vyjadřující počet epoch potřebných k dosažení 85 %, 90 %, 95 %, 96 %, 97 %, 98 % a 99 % správnosti na validační sadě pro jednotlivé konfigurace metody LSUV, lze vidět, že síť se všemi konfiguracemi LSUV dosáhla 99 % správnosti na validační sadě. Nejpomaleji této správnosti dosáhla síť využívající konfiguraci LSUV používající 256 obrázků z trénovací sady. Nejrychleji této správnosti dosáhla síť s konfigurací LSUV s 16 obrázky a 1024 obrázky z trénovací sady. Průběhy chyb a správnosti na validační a trénovací sadě pro 5 konfigurací metody LSUV, lze vidět na obrázku A.10. Průběhy jsou zobrazeny do epochy optimálního naučení.

### Náhodná inicializace vah z normálního a rovnoměrného pravděpodobnostního rozdělení

Výsledky pro síť využívající různé konfigurace metody inicializace vah z normálního pravděpodobnostního rozdělení můžeme vidět v tabulce A.27. 99,350 % byla nejvyšší dosažená správnost na testovací sadě. Této správnosti dosáhla síť s konfigurací normálního rozdělení ( $\mu = 0, \sigma = 0, 1$ ). K dosažení optimálního naučení potřebovala síť 26 epoch. Nejméně epoch k dosažení optimálního naučení, konkrétně 5, potřebovala síť s konfigurací normálního rozdělení ( $\mu = 0, \sigma = 0, 0005$ ). Naopak nejvíce epoch pro dosažení optimálního naučení bylo potřeba u sítě využívající konfiguraci ( $\mu = 0, \sigma = 0, 6$ ). Síť se všemi konfiguracemi normálního rozdělení, kromě konfigurace ( $\mu = 0, \sigma = 0, 1$ ), dosáhla vážené přesnosti, výtěžnosti a váženého F1-skóre 0,99. V tabulce A.28, vyjadřující počet epoch potřebných k dosažení 85 %, 90 %, 95 %, 96 %, 97 %, 98 % a 99 % správnosti na validační sadě, lze pozorovat, že 99 % správnosti na validační sadě nedosáhla pouze síť s konfiguracemi normálního rozdělení ( $\mu = 0, \sigma = 0, 9$ ) a ( $\mu = 0, \sigma = 0, 8$ ). Síť využívající konfiguraci normálního rozdělení ( $\mu = 0, \sigma = 0, 005$ ) potřebovala nejméně epoch, konkrétně 4, k dosažení této správnosti na validační sadě.

Výsledné hodnoty pro síť využívající různé konfigurace metody inicializace vah z rovnoměrného pravděpodobnostního rozdělení lze vidět v tabulce A.29. Nejvyšší správnost na testovací sadě dosáhla síť s konfigurací rovnoměrného rozdělení  $\langle -0,2;0,2 \rangle$ . Tato síť dosáhla 99,400 % správnosti. Síť potřebovala k dosažení optimálního naučení 26 epoch. Nejméně epoch k dosažení optimálního naučení potřebovala síť s konfigurací rovnoměrného rozdělení  $\langle -0,1;0,1 \rangle$ . Síť k dosažení optimálního naučení stačilo 9 epoch, také tato síť dosáhla 4. nejlepší správnosti na testovací sadě. Nejvíce epoch k dosažení optimálního naučení potřebovala síť s konfigurací rovnoměrného rozdělení  $\langle -0,8,0,8 \rangle$ . Hodnoty vážené přesnosti, výtěžnosti a váženého F1-skóre jsou pro síť se všemi konfiguracemi stejné. V tabulce A.30, vyjadřující počet epoch potřebných k dosažení 85 %, 90 %, 95 %, 96 %, 97 %, 98 % a 99 % správnosti na validační sadě, lze vidět, že 99 % správnosti na validační sadě dosáhly všechny konfigurace. Nejméně epoch k dosažení této správnosti, konkrétně 5, potřebovala síť s konfiguracemi rovnoměrného rozdělení  $\langle -0,1;0,1 \rangle$  a  $\langle -0,05;0,05 \rangle$ .

Srovnáním výsledků v tabulkách A.27 a A.29 lze zjistit, že celkové nejlepší správnosti na testovací sadě dosáhla síť používající metodu inicializace vah z rovnoměrným rozdělení s hodnotou limitu rovnou 0,2 ( $\langle -0,2;0,2 \rangle$ ). Průběhy chyb a správností na validační a trénovací sadě, pro hlubokou konvoluční neuronovou síť – Resnet18 využívající různé konfigurace metod inicializace vah z normálního a rovnoměrného pravděpodobnostního rozdělení lze vidět na obrázku A.11. Průběhy jsou zobrazeny do epochy optimálního naučení.

### **Ostatní metody náhodné inicializace**

Výsledné hodnoty pro síť využívající ostatní metody náhodné inicializace a jejich případné konfigurace jsou zaznamenány v tabulce A.31. Nejvyšší správnost na testovací sadě dosáhla síť používající metodu inicializace vah pomocí rozkladu na ortonormální bázi. Tato síť dosáhla 99,400 % správnosti na testovací sadě. Síť dosáhla společně s metodou LeCun vzorkující váhy z normálního rozdělení druhý nejmenší počet epoch k dosažení optimálního naučení. Nejrychleji dosáhla síť, používající metodu inicializace vah LeCun vzorkující váhy z rovnoměrného pravděpodobnostního rozdělení, optimálního naučení za 4 epoch. Nejpomaleji dosáhla optimálního naučení síť využívající metodu He vzorkující váhy z normálního rozdělení. Tato síť potřebovala 30 epoch.

V tabulce A.32, vyjadřující počet epoch k dosažení 85 %, 90 %, 95 %, 96 %, 97 %, 98 % a 99 % správnosti na validační sadě, je možno vidět, že síť se všemi metodami dosáhla 99 % správnosti na validační sadě. Nejrychleji bylo této správnosti dosaženo sítí s inicializací vah metodou LeCun vzorkující váhy z rovnoměrného rozdělení. Tato síť potřebovala pouze 4 epochy. Všechny konfigurace dosáhly průměrné vážené



výtěžnosti, přesnosti a váženého F1-skóre rovné 0,99.

Průběhy chyby a správnosti na validační a trénovací sadě, pro hlubokou vícevrstvou perceptronovskou síť využívající metody inicializace vah LeCun, He, Glorot v různých konfiguracích a metodu inicializace vah využívající rozklad na ortonormální bázi lze vidět na obrázku A.12. Průběhy jsou zobrazeny do epochy optimálního naučení.

## 8.2 Výsledky – databáze CIFAR–10

### 8.2.1 Mělká vícevrstvá perceptronovská síť

#### LSUV

Výsledky v tabulce B.1 vyjadřují ohodnocení rychlosti učení, ale také toho jak dobře se síť naučila. Nejlepší výsledek správnosti na testovací sadě byl 48,980 % a dosáhla ho síť, která používala k inicializaci vah LSUV využívající 2 obrázky z trénovací sady. Optimálního naučení síť s touto konfigurací LSUV dosáhla za 33 epoch, což je druhý nejlepší výsledek ze všech konfigurací. Rychleji optimálního naučení dosáhla s 32 epochami síť s konfigurací LSUV využívající 16 obrázků z trénovací sady. Naopak nejvíce epoch, konkrétně 53, potřebovala síť s konfigurací LSUV využívající 64 obrázků z trénovací sady.

Výsledky průměrného váženého F1-skóre, průměrné vážené přesnosti a výtěžnosti jsou pro testovací sadu nejlepší u sítě s konfigurací LSUV využívající 2 obrázky z trénovací sady.

Rychlost učení a to jak dobře se síť dokázala naučit, lze také pozorovat v tabulce B.2, která vyjadřuje počet epoch k dosažení 20 %, 25 %, 30 %, 35 %, 40 %, 45 % a 50 % správnosti na validační sadě pro jednotlivé konfigurace LSUV. 50 % správnosti na validační sadě dosáhla jako jediná síť s konfigurací LSUV používající 2 obrázky z trénovací sady. Síť této správnosti dosáhla za 33 epoch. Síť s konfigurací LSUV používající 1024 obrázků z trénovací sady nedosáhla ani 45 % správnosti na validační sadě. Průběhy chyby a správnosti na validační a trénovací sadě pro 5 konfigurací metody lze vidět na obrázku B.1. Průběhy jsou zobrazeny do epochy optimálního naučení.

#### **Náhodná inicializace vah z normálního a rovnoměrného pravděpodobnostního rozdělení**

Výsledky mělké vícevrstvé perceptronovské sítě používající různé konfigurace metody inicializace vah z normálního pravděpodobnostního rozdělení, jsou vidět v tabulce B.3. Nejvyšší správnosti na testovací sadě bylo dosaženo pomocí sítě s konfi-

gurací normálního rozdělení ( $\mu = 0, \sigma = 0,005$ ). Síť dosáhla 50,120 % správnosti na testovací sadě. K dosažení optimálního naučení potřebovala síť s touto konfigurací 32 epoch. Nejnižší správnost na trénovací sadě byla dosažena sítí s konfigurací normálního rozdělení ( $\mu = 0, \sigma = 1$ ). Nejvíce epoch, konkrétně 122, k dosažení optimálního naučení potřebovala síť využívající konfigurace normálního rozdělení ( $\mu = 0, \sigma = 0,9$ ). Nejmenší počet epoch k dosažení optimálního naučení sítě dosáhla síť s konfigurací normálního rozdělení ( $\mu = 0, \sigma = 0,05$ ). V tabulce B.3 lze také pozorovat tendenci zvyšování dosažené průměrné vážené přesnosti, výtěžnosti a F1-skóre se snižováním hodnoty směrodatné odchylky  $\sigma$ . Z tabulky B.4, která vyjadřuje počet epoch k dosažení 20 %, 25 %, 30 %, 35 %, 40 %, 45 % a 50 % správnosti na validační sadě, vyplývá, že 50 % správnosti dosáhly 3 konfigurace. Nejrychleji této správnosti bylo dosaženo za 26 epoch sítí s konfigurací normálního rozdělení ( $\mu = 0, \sigma = 0,1$ ).

Výsledné hodnoty pro síť využívající různé konfigurace metody inicializace vah z rovnoměrného pravděpodobnostního rozdělení jsou vidět v tabulce B.5. Nejvyšší správnost na testovací sadě dosáhla konfigurace rovnoměrného pravděpodobnostního rozdělení  $\langle -0,05; 0,05 \rangle$ . Dosažená správnost je rovna 49,420 %. Síť s touto konfigurací rovnoměrného rozdělení dosáhla optimálního naučení za 26 epoch. Tato hodnota je třetí nejlepší ze všech konfigurací. Rychleji optimálního naučení bylo dosaženo už jen sítí s konfiguracemi rovnoměrného rozdělení  $\langle -0,1; 0,1 \rangle$  a  $\langle -0,0005; 0,0005 \rangle$ . Síť s těmi to konfiguracemi dosáhla optimálního naučení za 17 epoch. Nejvíce epoch k dosažení optimálního naučení potřebovala síť s konfiguracemi rovnoměrného rozdělení  $\langle -1,1 \rangle$  a  $\langle -0,8; 0,8 \rangle$ . V tabulce B.6, vyjadřující počet epoch k dosažení 25 %, 25 %, 30 %, 35 %, 40 %, 45 % a 50 % správnosti na validační sadě, lze vidět, že síť se 4 konfiguracemi rovnoměrného pravděpodobnostního rozdělení, které dosáhla 50 % správnosti. Použité konfigurace mají hodnotu limitu rovnou 0,2; 0,1; 0,05 a 0,005.

Porovnáním výsledků v tabulce B.3 a B.5 lze zjistit, že celkové nejlepší správnosti na testovací sadě dosáhla síť používající metodu inicializace vah z normálního pravděpodobnostního rozdělení s průměrem rozdělení rovným 0 a směrodatnou odchylkou rovnou 0,005. Průběhy chyb a správností, pro síť využívající různé konfigurace metod inicializace vah z rovnoměrného a normálního rozdělení, na validační a trénovací sadě lze vidět na obrázku B.2. Průběhy jsou zobrazeny do epochy optimálního naučení.

### **Ostatní metody náhodné inicializace**

Výsledné hodnoty pro síť využívající ostatní metody náhodné inicializace a jejich případné konfigurace jsou zaznamenány v tabulce B.7. Nejvyšší hodnota správnosti na testovací sadě byla dosažena sítí používající metodu inicializace vah He z nor-

málního pravděpodobnostního rozdělení. Tato síť dosáhla správnosti 49,230 %. K dosažení optimálního naučení sítě bylo potřeba 34 epoch. Nejrychleji bylo dosaženo optimálního naučení za 31 epoch a to sítí, používající metodu inicializace vah Glorot z rovnoměrného pravděpodobnostního rozdělení. Síť využívající k inicializaci vah pomocí rozkladu na ortonormální bázi dosáhla třetí nejlepší správnosti na testovací sadě a optimálního naučení dosáhla za 32 epoch.

V tabulce B.8, vyjadřující počet epoch k dosažení 20 %, 25 %, 30 %, 35 %, 40 %, 45 % a 50 % správnosti na validační sadě, lze vidět, že síť se 4 konfiguracemi dosáhla 50 % správnosti na validační sadě. Nejrychleji bylo této správnosti dosaženo za 31 epoch a to sítí používající k inicializaci vah metodu He vzorkující váhy z normálního rozdělení.

Průběhy chyby a správnosti na validační a trénovací sadě, pro mělkou vícevrstvou perceptronovskou síť, využívající metody inicializace vah LeCun, He, Glorot v různých konfiguracích a metodu inicializace vah využívající rozklad na ortonormální bázi lze vidět na obrázku B.3. Průběhy jsou zobrazeny do epochy optimálního naučení.

## 8.2.2 Hluboká vícevrstvá perceptronovská síť

### LSUV

V tabulce B.9 lze vidět výsledky, pro síť používající metodu inicializace vah LSUV v různých konfiguracích. Nejlepší výsledek správnosti na testovací sadě byl dosažen sítí používající k inicializaci konfiguraci LSUV s 16 obrázků z trénovací sady. Síť dosáhla správnosti 49,670 %. Na druhou stranu, tato síť dosáhla optimálního naučení za 15 epoch, což je nejhorší výsledek ze všech konfigurací.

Nejrychleji optimálního naučení dosáhla síť inicializovaná metodou LSUV pomocí 1024 a 16 obrázků. Tato síť s oběma konfiguracemi potřebovala 9 epoch. Nejlepší výsledky vážené přesnosti, výtěžnosti a váženého F1-skóre dosáhla síť využívající konfiguraci inicializace LSUV s 16 obrázky.

V tabulce B.10, vyjadřující počet epoch potřebných k dosažení 20 %, 25 %, 30 %, 35 %, 40 %, 45 %, 50 % správnosti na validační sadě pro jednotlivé konfigurace metody LSUV, lze vidět, že všechny konfigurace LSUV dosáhly 45 % správnosti na validační sadě. Nejrychleji, konkrétně za 6 epoch, této správnosti dosáhla síť využívající konfiguraci LSUV používající 64 obrázků z trénovací sady. Průběhy chyb a správností na validační a trénovací sadě pro 5 konfigurací metody LSUV, lze vidět na obrázku B.5. Průběhy jsou zobrazeny do epochy optimálního naučení.

## Náhodná inicializace vah z normálního a rovnoměrného pravděpodobnostního rozdělení

Výsledky pro síť využívající různé konfigurace metody inicializace vah z normálního pravděpodobnostního rozdělení můžeme vidět v tabulce B.11. Nejvyšší správnost na testovací sadě dosáhla síť s konfigurací normálního rozdělení ( $\mu = 0, \sigma = 0, 1$ ). Síť dosáhla 48,600 % správnosti na testovací sadě. K dosažení optimálního naučení potřebovala síť 12 epoch, což je, pokud nepočítáme síť s konfiguracemi, které nebyly schopny konvergovat nejlepší výsledek. Z hodnot v tabulce B.11, ale také pozorováním průběhů na obrázku B.5, lze zjistit, že pouze síť s konfiguracemi normálního rozdělení ( $\mu = 0, \sigma = 0, 3$ ), ( $\mu = 0, \sigma = 0, 2$ ), ( $\mu = 0, \sigma = 0, 1$ ) a ( $\mu = 0, \sigma = 0, 05$ ) byla schopna konvergovat. V tabulce B.12, vyjadřující počet epoch potřebných k dosažení 20 %, 25 %, 30 %, 35 %, 40 %, 45 %, 50 % správnosti na validační sadě, lze pozorovat, že 50 % správnosti dosáhla pouze síť s konfigurací normálního rozdělení ( $\mu = 0, \sigma = 0, 1$ ). Také lze pozorovat, že síť pouze se 3 konfiguracemi, konkrétně ( $\mu = 0, \sigma = 0, 2$ ), ( $\mu = 0, \sigma = 0, 1$ ) a ( $\mu = 0, \sigma = 0, 05$ ), byla schopna dosáhnout alespoň 30 % správnosti na validační sadě.

Výsledné hodnoty pro síť využívající různé konfigurace metody inicializace vah z rovnoměrného pravděpodobnostního rozdělení lze vidět v tabulce B.13. Nejvyšší správnosti na testovací sadě, konkrétně 48,820 %, bylo dosaženo sítí s konfigurací rovnoměrného rozdělení  $\langle -0,2;0,2 \rangle$ . Síť potřebovala k dosažení optimálního naučení síť 15 epoch. Když nepočítáme konfigurace, se kterou síť nebyla schopna konvergovat, tak je 15 epoch k dosažení optimálního naučení nejlepší výsledek. Z hodnot v tabulce B.11, ale také pozorováním průběhů na obrázku B.5, lze vidět, že pouze sítě se 5 konfiguracemi byly schopny konvergovat. Konkrétně sítě s konfiguracemi rovnoměrného rozdělení  $\langle -0,4;0,4 \rangle$ ,  $\langle -0,3;0,3 \rangle$ ,  $\langle -0,2;0,2 \rangle$ ,  $\langle -0,1;0,1 \rangle$  a  $\langle -0,05;0,05 \rangle$ . V tabulce B.14, vyjadřující počet epoch potřebných k dosažení 85 %, 90 %, 95 %, 96 %, 97 %, 98 % a 99 % správnosti na validační sadě, lze potvrdit, že síť pouze se 5 konfiguracemi je schopna konvergence. Síť se výše zmíněnými 5 konfiguracemi byla schopna dosáhnout 45 % správnosti na validační sadě. Nejméně epoch k dosažení této správnosti, konkrétně 8, potřebovala síť s konfigurací rovnoměrného rozdělení  $\langle -0,2;0,2 \rangle$  a  $\langle -0,1;0,1 \rangle$ .

Při porovnání výsledků v tabulkách B.11 a B.13 lze zjistit, že celkové nejlepší správnosti na testovací sadě dosáhla síť používající metodu inicializace vah z rovnoměrného rozdělení s limitem rovným 0,2. Také si lze všimnout, že síť s konfigurací normálního rozdělení využívající směrodatnou odchylku 0,5 a větší nebo 0,005 a menší nebyla schopna konvergence. U sítě využívající konfiguraci rovnoměrného rozdělení jsou hodnoty limitu, pro které nebyla síť schopna konvergence 0,5 a vyšší nebo 0,005 a menší. Průběhy chyby a správnosti na validační a trénovací sadě, pro síť

využívající různé konfigurace metod inicializace vah z normálního a rovnoměrného pravděpodobnostního rozdělení lze vidět na obrázku B.5. Průběhy jsou zobrazeny do epochy optimálního naučení.

### **Ostatní metody náhodné inicializace**

Výsledné hodnoty pro síť využívající ostatní metody náhodné inicializace a jejich případné konfigurace jsou zaznamenány v tabulce B.15. Nejvyšší hodnota správnosti na testovací sadě dosažena sítí používající metodu inicializace vah Glorot z rovnoměrného pravděpodobnostního rozdělení. Tato síť dosáhla 48,120 % správnosti. Tato síť dosáhla optimálního naučení za 15 epoch. Tento výsledek je třetí nejhorší. Nejrychleji, konkrétně za 8 epoch, optimálního naučení dosáhla síť, která používala metodu inicializace vah He z rovnoměrného pravděpodobnostního rozdělení, Síť využívající k inicializaci vah pomocí rozkladu na ortonormální bázi dosáhla optimálního naučení za 17 epoch. Tento výsledek je nejhorší ze všech metod.

V tabulce B.16, vyjadřující počet epoch k dosažení 20 %, 25 %, 30 %, 35 %, 40 %, 45 %, 50 % správnosti na validační sadě, je pozorovat, že síť se všemi metodami dosáhla 45 % správnosti na validační sadě. Nejpomaleji této správnosti dosáhla síť s metodou inicializace vah LeCun z rovnoměrného rozdělení a metodou inicializace vah pomocí rozkladu na ortonormální bázi. Nejrychleji bylo této správnosti dosaženo za 7 epoch, a to sítí s inicializací vah He z rovnoměrného rozdělení.

Průběhy chyby a správnosti na validační a trénovací sadě, pro hlubokou vícevrstvou perceptronovskou síť využívající metody inicializace vah LeCun, He, Glorot v různých konfiguracích a metodu inicializace vah využívající rozklad na ortonormální bázi lze vidět na obrázku B.6. Průběhy jsou zobrazeny do epochy optimálního naučení.

### **8.2.3 Jednoduchá konvoluční neuronová síť**

#### **LSUV**

V tabulce B.17 lze vidět výsledky, pro jednoduchou konvoluční neuronovou síť využívající metodu inicializace vah LSUV v různých konfiguracích. Nejlepšího výsledku správnosti na testovací sadě dosáhla síť využívající k inicializaci vah metodu LSUV se 2 obrázky z trénovací sady. Síť dosáhla správnosti 68,480 %. Síť, ale dosáhla nejhoršího výsledku v počtu epoch k dosažení optimálního naučení, protože potřebovala 10 epoch.

Nejrychleji dosáhla síť optimálního naučení s konfigurací LSUV využívající 1024 obrázků z trénovací sady.

V tabulce B.18, vyjadřující počet epoch potřebných k dosažení 40 %, 45 %, 50 %, 55 %, 60 %, 65 % a 70 % správnosti na validační sadě, lze vidět, že síť se všemi konfiguracemi LSUV dosáhla 65 % správnosti na validační sadě. Nejrychleji této správnosti dosáhla síť s konfigurací LSUV využívající 2 obrázky z trénovací sady. Nejpomaleji této správnosti dosáhla síť využívající konfiguraci LSUV používající 16 a 256 obrázků z trénovací sady. Průběhy chyb a správností na validační a trénovací sadě pro 5 konfigurací metody LSUV, lze vidět na obrázku B.7. Průběhy jsou zobrazeny do epochy optimálního naučení.

### **Náhodná inicializace vah z normálního a rovnoměrného pravděpodobnostního rozdělení**

Výsledky pro síť využívající různé konfigurace metody inicializace vah z normálního pravděpodobnostního rozdělení můžeme vidět v tabulce B.19. Nejvyšší správnost na testovací sadě byla dosažena konfigurací normálního rozdělení ( $\mu = 0, \sigma = 0,05$ ). Síť dosáhla správnosti 69,360 %. K dosažení optimálního naučení síť potřebovala 12 epoch. Nejméně epoch, konkrétně 8, potřebovala konvergující síť využívající konfiguraci normálního rozdělení ( $\mu = 0, \sigma = 0,1$ ). Tato síť také dosáhla druhý nejlepší výsledek správnosti na testovací sadě. Z obrázku B.8 a i vyšších hodnot v tabulce B.19, lze pozorovat, že síť byla schopna konvergence jen s 5 konfiguracemi normálního rozdělení. Tyto konfigurace jsou ( $\mu = 0, \sigma = 0,3$ ), ( $\mu = 0, \sigma = 0,2$ ), ( $\mu = 0, \sigma = 0,1$ ), ( $\mu = 0, \sigma = 0,05$ ) a ( $\mu = 0, \sigma = 0,005$ ).

V tabulce B.20, vyjadřující počet epoch potřebných k dosažení 40 %, 45 %, 50 %, 55 %, 60 %, 65 % a 70 % správnosti na validační sadě, lze vidět, že 70 % správnosti na validační sadě dosáhla pouze síť s konfiguracemi normálního rozdělení ( $\mu = 0, \sigma = 0,1$ ) a ( $\mu = 0, \sigma = 0,05$ ). Nejméně epoch k dosažení 70 % správnosti na validační sadě potřebovala síť využívající konfiguraci normálního rozdělení ( $\mu = 0, \sigma = 0,1$ ). Naopak síť s konfiguracemi normálního rozdělení s průměrem rovným 0 a se směrodatnou odchylkou rovnou 0,4 nebo větší a rovnou 0,005 nebyla schopna konvergence.

Výsledné hodnoty pro síť využívající různé konfigurace metody inicializace vah z rovnoměrného pravděpodobnostního rozdělení, lze vidět v tabulce B.21. Nejvyšší správnosti na testovací sadě, konkrétně 69,460 %, dosáhla konfigurace rovnoměrného rozdělení  $\langle -0,2; 0,2 \rangle$ . K dosažení optimálního naučení této sítě bylo potřeba 10 epoch. Pokud počítáme pouze síť s konfiguracemi, která konvergovala tak nejméně epoch, konkrétně 8 k dosažení optimálního naučení potřebovala síť s konfigurací normálního rozdělení  $\langle -0,1; 0,1 \rangle$ . Tato síť také dosáhla druhé nejlepší správnosti na testovací sadě. Z průběhů na obrázku B.8 a také z tabulky B.21, lze vidět, že síť pouze se 7 konfiguracemi byla schopna konvergence. Hodnoty limitů rovnoměrného

rozdělení jsou od 0,6 do 0,05. V tabulce B.22, vyjadřující počet epoch potřebných k dosažení 40 %, 45 %, 50 %, 55 %, 60 %, 65 % a 70 % správnosti na validační sadě, lze vidět, že 70 % správnosti na validační sadě dosáhla síť se 2 různými konfiguracemi rovnoměrného rozdělení. Nejméně epoch k dosažení 70 % správnosti na validační sadě potřebovala síť s konfigurací rovnoměrného rozdělení  $\langle -0,2;0,2 \rangle$ . Tato síť potřebovala pouze 8 epoch.

Při porovnání výsledků v tabulkách B.19 a B.21 lze zjistit, že celkového nejlepšího výsledku správnosti na testovací sadě dosáhla síť používající metodu inicializace vah z rovnoměrného rozdělení s limitem rovným 0,2 ( $\langle -0,2;0,2 \rangle$ ). Průběhy chyby a správnosti na validační a trénovací sadě, pro síť využívající různé konfigurace metod inicializace vah z normálního a rovnoměrného pravděpodobnostního rozdělení lze vidět na obrázku B.8. Průběhy jsou zobrazeny do epochy optimálního naučení.

### **Ostatní metody náhodné inicializace**

Výsledné hodnoty pro síť využívající ostatní metody náhodné inicializace a jejich případné konfigurace jsou zaznamenány v tabulce B.23. Nejvyšší správnost na testovací sadě, byla dosažena sítí využívající metodu inicializace vah LeCun z normálního rozdělení. Tato síť dosáhla 70,370 % správnosti na testovací sadě, ale k dosažení optimálního naučení potřebovala nejvíce epoch. Nejméně epoch k dosažení optimálního naučení potřebovala síť s metodou He v obou konfiguracích. Síť s metodou He vzorkující váhy z normálního rozdělení dosáhla druhé nejvyšší správnosti na validační sadě. V tabulce B.24, vyjadřující počet epoch k dosažení 40 %, 45 %, 50 %, 55 %, 60 %, 65 % a 70 % správnosti na validační sadě, je možno vidět, že síť se všemi metodami dosáhla 70 % správnosti na validační sadě. Nejpomaleji bylo této správnosti dosaženo sítí s konfigurací He vzorkující váhy z rovnoměrného rozdělení a metodou inicializace vah rozkladem na ortonormální bázi. Nejrychleji bylo této správnosti dosaženo za 4 epochy, a to sítí s inicializací vah metodou He vzorkující váhy z normálního rozdělení a metodou LeCun vzorkující váhy z rovnoměrného rozdělení.

Průběhy chyby a správnosti na validační a trénovací sadě, pro hlubokou vícevrstvou perceptronovskou síť využívající metody inicializace vah LeCun, He, Glorot v různých konfiguracích a metodu inicializace vah využívající rozklad na ortonormální bázi lze vidět na obrázku B.9. Průběhy jsou zobrazeny do epochy optimálního naučení.

## 8.2.4 Hluboká konvoluční neuronová síť – ResNet18

### LSUV

V tabulce B.25 lze vidět výsledky, pro síť používající metodu inicializace vah LSUV v různých konfiguracích. 72,170 % byla nejvyšší dosažená správnost na testovací sadě. Této správnosti dosáhla síť používající konfiguraci LSUV s 64 obrázky z trénovací sady. Optimálního naučení dosáhla za 8 epoch, což je nejhorší výsledek ze všech konfigurací.

Nejméně epoch k dosažení optimálního naučení potřebovala síť využívající konfiguraci LSUV se 2 obrázky z trénovací sady. Síť potřebovala 6 epoch. Také dosáhla 3. nejlepší správnosti na testovací sadě.

V tabulce B.26, vyjadřující počet epoch potřebných k dosažení 45 %, 50 %, 55 %, 60 %, 65 % a 70 %, 75 % správnosti na validační sadě pro jednotlivé konfigurace metody LSUV, lze vidět, že pouze síť s konfigurací LSUV využívající 64 obrázků z trénovací sady dosáhla 75 % správnosti na validační sadě. Síť dosáhla této správnosti za 13 epoch. Průběhy chyb a správnosti na validační a trénovací sadě pro 5 konfigurací metody LSUV, lze vidět na obrázku B.10. Průběhy jsou zobrazeny do epochy optimálního naučení.

### Náhodná inicializace vah z normálního a rovnoměrného pravděpodobnostního rozdělení

Výsledky pro síť využívající různé konfigurace metody inicializace vah z normálního pravděpodobnostního rozdělení můžeme vidět v tabulce B.27. Nejvyšší správnost na testovací sadě dosáhla síť s konfigurací normálního rozdělení ( $\mu = 0, \sigma = 0,005$ ). Síť dosáhla správnosti 73,380 %. K dosažení optimálního naučení potřebovala síť 9 epoch. Nejvíce epoch pro dosažení optimálního naučení bylo potřeba u sítě využívající konfiguraci ( $\mu = 0, \sigma = 1$ ) Naopak nejméně epoch k dosažení optimálního naučení, konkrétně 4, potřebovala síť s konfiguracemi normálního rozdělení ( $\mu = 0, \sigma = 0,1$ ) a ( $\mu = 0, \sigma = 0,2$ ). V tabulce B.27 si lze také všimnout, že se snižující se hodnotou směrodatné odchylky normálního pravděpodobnostního rozdělení se zvyšuje dosažená správnost na validační sadě. Schopnost dosáhnout vyšší správnosti na validační sadě se snižující se směrodatnou odchylkou lze pozorovat v tabulce B.28. Tabulka B.28, vyjadřuje počet epoch potřebných k dosažení 45 %, 50 %, 55 %, 60 %, 65 %, 70 % a 75 % správnosti na validační sadě. V této tabulce lze vidět, že 75 % správnosti na validační sadě dosáhla síť pouze se 2 konfiguracemi normálního rozdělení, konkrétně ( $\mu = 0, \sigma = 0,005$ ) a ( $\mu = 0, \sigma = 0,0005$ ). Nejrychleji této správnosti dosáhla síť s konfigurací ( $\mu = 0, \sigma = 0,005$ ).

Výsledné hodnoty pro síť využívající různé konfigurace metody inicializace vah



z rovnoměrného pravděpodobnostního rozdělení lze vidět v tabulce B.29. Nejvyšší správnost na testovací sadě dosáhla síť s konfigurací rovnoměrného rozdělení  $\langle -0,005;0,005 \rangle$ . Tato síť dosáhla 72,750 % správnosti. Síť potřebovala k dosažení optimálního naučení 9 epoch. Nejméně epoch k dosažení optimálního naučení potřebovala síť s konfiguracemi rovnoměrného rozdělení  $\langle -0,8;0,8 \rangle$  a  $\langle -0,6;0,6 \rangle$ . Tato síť potřebovala 4 epochy. Nejvíce epoch k dosažení optimálního naučení potřebovala síť s konfigurací rovnoměrného rozdělení  $\langle -1,1 \rangle$ . Z tabulky B.29 lze vyčíst, že se snižující se hodnotou limitu rovnoměrného rozdělení má tendenci stoupat i dosažená správnost na testovací sadě. Také v tabulce B.30, vyjadřující počet epoch potřebných k dosažení 45 %, 50 %, 55 %, 60 %, 65 % a 70 %, 75 % správnosti na validační sadě, si lze všimnout, že se snižujícím se limitem rovnoměrného rozdělení se zvyšuje dosažená správnost na validační sadě. V tabulce B.30 lze vidět, že 75 % správnosti na validační sadě dosáhla síť se 3 konfiguracemi. Nejméně epoch k dosažení této správnosti, konkrétně 5, potřebovala síť s konfiguracemi rovnoměrného rozdělení  $\langle -0,005;0,005 \rangle$  a  $\langle -0,0005;0,0005 \rangle$ .

Srovnáním výsledků v tabulkách B.27 a B.29 lze zjistit, že celkové nejlepší správnosti na testovací sadě dosáhla síť používající metodu inicializace vah z normálního pravděpodobnostního rozdělení jehož průměr je rovný 0 a směrodatná odchylka má hodnotu 0,005. Průběhy chyb a správností na validační a trénovací sadě, pro hlubokou konvoluční neuronovou síť - Resnet18 využívající různé konfigurace metod inicializace vah z normálního a rovnoměrného pravděpodobnostního rozdělení lze vidět na obrázku B.11. Průběhy jsou zobrazeny do epochy optimálního naučení.

### **Ostatní metody náhodné inicializace**

Výsledné hodnoty pro síť využívající ostatní metody náhodné inicializace a jejich případné konfigurace jsou zaznamenány v tabulce B.31. Nejvyšší správnost na testovací sadě dosáhla síť používající metodu inicializace vah pomocí rozkladu na ortonormální bázi. Tato síť dosáhla 72,820 % správnosti na testovací sadě. K dosažení optimálního naučení potřebovala tato síť 8 epoch. Nejrychleji, konkrétně za 6 epoch dosáhla optimálního naučení síť, používající metodu inicializace vah LeCun vzorkující váhy z normálního pravděpodobnostního rozdělení.

V tabulce B.32, vyjadřující počet epoch k dosažení 45 %, 50 %, 55 %, 60 %, 65 % a 70 % a 75 % správnosti na validační sadě, je možno vidět, že síť pouze se 2 metodami dosáhla 75 % správnosti na validační sadě. Nejrychleji bylo této správnosti dosaženo sítí s inicializací vah metodou Glorot vzorkující váhy z rovnoměrného rozdělení. Tato síť potřebovala 17 epochy.

Průběhy chyby a správnosti na validační a trénovací sadě, pro hlubokou konvoluční neuronovou - ResNet18 síť využívající metody inicializace vah LeCun, He,

Glorot v různých konfiguracích a metodu inicializace vah využívající rozklad na ortonormální bázi lze vidět na obrázku B.12. Průběhy jsou zobrazeny do epochy optimálního naučení.

### 8.3 Srovnání a diskuse výsledků

V tabulce 8.2, lze vidět použité metody inicializace vah, díky kterým bylo dosaženo pro jednotlivé typy architektur nejvyšší správnosti na testovací sadě databáze MNIST. Nejvyšší celková přesnost na testovací sadě byla dosažena sítí ResNet18, která byla inicializována metodou inicializace vah pomocí rozkladu na ortonormální bázi. Tato síť také dosáhla druhého nejlepšího výsledku v tabulce 8.2 v počtu epoch k dosažení optimálního naučení. Druhý nejvyšší výsledek správnosti na testovací sadě dosáhla v tabulce 8.2 jednoduchá konvoluční neuronová síť inicializovaná metodou Glorot vzorkující váhy z normálního pravděpodobnostního rozdělení. Rozdíl, ve správnosti na testovací sadě, mezi jednoduchou konvoluční neuronovou sítí a ResNet18 je 0,210 %. Na druhou stranu tato jednoduchá konvoluční neuronová síť dosáhla optimálního naučení za 5 epoch. V tabulce 8.4, popisující typ architektury a metodu, pomocí které bylo dosaženo nejrychlejšího optimálního naučení jednotlivých sítí, lze pozorovat, že nejméně epoch potřebovala jednoduchá konvoluční neuronová síť inicializovaná metodou LSUV s 64 obrázky. ResNet18 inicializovaná metodou LeCun vzorkující váhy z rovnoměrného pravděpodobnostního rozdělení potřebovala k dosažení optimálního naučení 4 epochy. Také v tabulce 8.4 dosáhla nejvyšší správnosti na testovací sadě. Rychlost, ale také to, jak dobře se sítě naučily, lze vidět v tabulce 8.6, vyjadřující počet epoch k dosažení určité správnosti na validační sadě. Výsledky potvrzují, že pomocí inicializace LeCun vzorkující váhy z rovnoměrného rozdělení bylo možno dosáhnout optimálního a rychlého naučení sítě ResNet18. V tabulce 8.2, lze pozorovat, že nejhoršího výsledku správnosti na testovací sadě, ze všech architektur, bylo dosaženo mělkou vícevrstvou perceptronovskou sítí, která byla inicializovaná metodou He vzorkující váhy z normálního rozdělení.

Tabulka 8.3 vyjadřuje shrnutí nejvyšších správností na testovací sadě databáze CIFAR-10 pro jednotlivé typy architektur. Pro jednotlivé typy architektur jsou zmíněny také použité metody inicializace, díky kterým byly sítě schopny dosáhnout těchto správností. Nevyšší hodnota správnosti na testovací sadě CIFAR-10 byla dosažena sítí ResNet18, používající metodu inicializace vah z normálního pravděpodobnostního rozdělení s parametry rozdělení ( $\mu = 0, \sigma = 0,005$ ). Tato síť také dokázala dosáhnout optimálního naučení nejrychleji ze všech ostatních sítí v tabulce 8.3. Také z hodnot v tabulce 8.7, která vyjadřuje počet epoch k dosažení určité správnosti na validační sadě, lze vyčíst, že Resnet18 inicializovaná z normálního pravděpodobnostního rozdělení s parametry rozdělení ( $\mu = 0, \sigma = 0,005$ ) byla jako jediná schopna

dosáhnout hodnoty správnosti 75 %. V tabulce 8.5, lze vidět, že nejrychleji dokázala dosáhnout optimálního naučení síť ResNet18, jejíž váhy byly inicializovány z normálního pravděpodobnostního rozdělení, parametry rozdělení jsou ( $\mu = 0, \sigma = 0, 1$ ). V tabulce 8.3, lze pozorovat, že nejhoršího výsledku správnosti na testovací sadě CIFAR-10, ze všech architektur, dosáhla hluboká vícevrstvá perceptronovská síť, která byla inicializována metodou LSUV využívající 16 obrázků z trénovací sady.

*Tab. 8.2: Shrnutí nejlepších výsledků správnosti na trénovací sadě všech architektur pro databázi MNIST. První sloupec vyjadřuje typ architektury a metodu, díky které bylo dosaženo nejlepší správnosti na testovací sadě. Poslední sloupec vyjadřuje počet epoch k dosažení optimálního naučení sítě. Žluté podbarvení vyjadřuje nejvyšší dosaženou správnost a oranžové podbarvení nejnižší počet epoch.*

typ architektury – metoda inicializace	testovací správnost	počet epoch k dosažení optimálního naučení
mělká vícevrstvá perceptronovská síť – He normální	98,140 %	11
hluboká vícevrstvá perceptronovská síť – LeCun normální	98,180 %	15
jednoduchá konvoluční neuronová síť – Glorot normální	99,190 %	<b>5</b>
ResNet18 – ortonormální	<b>99,400 %</b>	9

Tab. 8.3: Shrnutí nejlepších výsledků správnosti na trénovací sadě všech architektur pro databázi CIFAR-10. První sloupec vyjadřuje typ architektury a metodu, díky které bylo dosaženo nejlepší správnosti na testovací sadě. Poslední sloupec vyjadřuje počet epoch k dosažení optimálního naučení sítě. Žluté podbarvení vyjadřuje nejvyšší dosaženou správnost a oranžové podbarvení nejnižší počet epoch.

typ architektury – metoda inicializace	testovací správnost	počet epoch k dosažení optimálního naučení
mělká vícevrstvá perceptronová síť – normální ( $\mu = 0, \sigma = 0,005$ )	50,120 %	32
hluboká vícevrstvá perceptronová síť – LSUV 16 obrázků	49,670 %	15
jednoduchá konvoluční neuronová síť – LeCun normální	70,370 %	14
ResNet18 – normální ( $\mu = 0, \sigma = 0,005$ )	<b>72,380 %</b>	<b>9</b>

Tab. 8.4: Shrnutí nejrychlejších optimálních naučení všech architektur pro databázi MNIST. První sloupec vyjadřuje typ architektury a metodu, díky které bylo dosaženo nejlepší správnosti na testovací sadě. Poslední sloupec vyjadřuje počet epoch k dosažení optimálního naučení sítě. Žluté podbarvení vyjadřuje nejvyšší dosaženou správnost a oranžové podbarvení nejnižší počet epoch.

typ architektury – metoda inicializace	testovací správnost	počet epoch k dosažení optimálního naučení
mělká vícevrstvá perceptronová síť – LSUV 16 obrázků	97,180 %	3
hluboká vícevrstvá perceptronová síť – LSUV 1024 obrázků	97,180 %	3
jednoduchá konvoluční neuronová síť – LSUV 64 obrázků	98,700 %	<b>2</b>
ResNet18 – LeCun rovnoměrné	<b>99,030 %</b>	4

Tab. 8.5: Shrnutí nejrychlejších optimálních naučení všech architektur pro databázi CIFAR-10. První sloupec vyjadřuje typ architektury a metodu, díky které bylo dosaženo nejlepší správnost na testovací sadě. Poslední sloupec vyjadřuje počet epoch k dosažení optimálního naučení sítě. Žluté podbarvení vyjadřuje nejvyšší dosaženou správnost a oranžové podbarvení nejnižší počet epoch.

typ architektury – metoda inicializace	testovací správnost	počet epoch k dosažení optimálního naučení
mělká vícevrstvá perceptronovská síť – normální ( $\mu = 0, \sigma = 0, 1$ )	49,420 %	17
hluboká vícevrstvá perceptronovská síť – He normální	46,900 %	8
jednoduchá konvoluční neuronová síť – LSUV 1024 obrázků	<b>66,330 %</b>	7
ResNet18 – normální ( $\mu = 0, \sigma = 0, 1$ )	66,080 %	<b>4</b>

Tab. 8.6: Popis nejlepších výsledků všech architektur v počtu epoch k dosažení určité správnosti na databázi MNIST. První řádek vyjadřuje typ architektury a zvolená metoda inicializace vah, se kterou bylo dosaženo nejvyšší validační správnosti za nejmenší počet epoch. Nejlepší výsledek je podbarven modrou.

typ architektury – metoda inicializace	Počet epoch k dosažení určité správnosti na validační sadě databáze MNIST						
	85 %	90 %	95 %	96 %	97 %	98 %	99 %
mělká vícevrstvá perceptronovská síť – ortonormální	1	1	1	1	2	5	-
hluboká vícevrstvá perceptronovská síť – ortonormální	1	1	1	1	3	5	-
jednoduchá konvoluční neuronová síť – LSUV 2 obrázků	1	1	1	1	1	1	3
ResNet18 – LeCun rovnoměrné	1	1	2	2	2	2	4

Tab. 8.7: Popis nejlepších výsledků všech architektur v počtu epoch k dosažení určité správnosti na databázi CIFAR-10. První řádek vyjadřuje typ architektury a zvolená metoda inicializace vah, se kterou bylo dosaženo nejvyšší validační správnosti za nejmenší počet epoch. Nejlepší výsledek je podbarven modrou.

typ architektury – metoda inicializace	Počet epoch k dosažení určité správnosti na validační sadě databáze CIFAR-10						
	45 %	50 %	55 %	60 %	65 %	70 %	75 %
mělká vícevrstvá perceptronovská sít – normální ( $\mu = 0, \sigma = 0, 1$ )	7	26	-	-	-	-	-
hluboká vícevrstvá perceptronovská sít – normální ( $\mu = 0, \sigma = 0, 1$ )	5	20	-	-	-	-	-
jednoduchá konvoluční neuronová sít – normální ( $\mu = 0, \sigma = 0, 1$ )	1	1	2	3	5	7	-
ResNet18 – normální ( $\mu =$ $0, \sigma = 0, 005$ )	1	2	3	3	6	6	12

Výsledky úvodního experimentu, při kterém byla inicializována mělká vícevrstvá perceptronovská síť nulami, potvrdily tvrzení z [1] o nevhodnosti použití nul jako počáteční váhy.

Nejvyšší správnosti na testovací sadě, bylo pro mělkou vícevrstvou perceptronovskou síť klasifikující obrázky z databáze MNIST, dosaženo pomocí metody He vzorkující váhy z normálního pravděpodobnostního rozdělení. Tento výsledek souhlasí s tvrzením z [30], že se metoda He, hodí pro síť využívající aktivační funkci ReLU. Tyto výsledky se liší oproti výsledkům v [52]. V [52] byly zkoumány vlivy 4 různých metod inicializace na učení mělké vícevrstvé perceptronovské sítě a nejvyšších výsledků správnosti na testovací sadě MNIST bylo dosaženo pomocí metody Glorot, vzorkující váhy z rovnovážného rozdělení. Rozdílné výsledky mohly vzniknout z důvodu vlivu náhody při generování vah z pravděpodobnostního rozdělení. Rozdíl mohl být způsoben i použitím rozdílných hyperparametrů.

V [32] je zmíněno, že síť s metodou inicializace LSUV, dosahuje rychlejší konvergence než síť využívající ostatní metody inicializace. Toto tvrzení lze na základě provedených experimentů potvrdit. Pro databázi MNIST, 3 ze 4 typů architektur, konkrétně mělká a hluboká vícevrstvá perceptronovská síť a jednoduchá konvoluční neuronová síť, dosáhly nejrychlejšího optimálního naučení za použití různých konfigurací metody inicializace LSUV. Pro databázi CIFAR-10, bylo pouze pro jednoduchou konvoluční neuronovou síť, používající metodu inicializace LSUV, dosaženo nejrychlejšího optimálního naučení. Nutné je zmínit, že pro ResNet18 a hlubokou vícevrstvou perceptronovskou síť bylo dosaženo druhého nejrychlejšího naučení při využití metody LSUV. V obou případech se rozdíl v počtu epoch, mezi nejrychlejším a druhým nejrychlejším dosažením optimálního naučení, lišil pouze o 1 epochu. Na základě těchto výsledků experimentů, lze doporučit použití metody inicializace vah LSUV, pro dosažení co nejrychlejší konvergence sítě.

V [32] je zmíněno, že nejhoršího výsledku správnosti dosáhla síť využívající konfiguraci LSUV se 2 obrázky. Toto tvrzení se na základě experimentů nepotvrdilo, protože celkových nejhorších výsledků správnosti na testovací sadě databáze CIFAR-10 bylo dosaženo pomocí metody LSUV s 1024 obrázky. Rozdíl mezi výsledky v [32] a dosaženými výsledky mohou být způsobeny použitím rozdílné aktivační funkce a architektury sítě. Dále mohl být rozdíl způsoben, použitím rozdílných hyperparametrů sítě.

Dále je v [32] zmíněno, že počet použitých obrázku, tedy kromě 2, nemá vliv na výslednou správnost na testovací sadě. Na základě experimentů bylo dosaženo rozdílných výsledků správností na testovací sadě při užití různých počtů obrázku pro inicializaci LSUV. Nejlepších celkových výsledků správnosti na testovací sadě MNIST, pro různé architektury využívající metodu inicializace vah LSUV, bylo dosaženo pomocí konfigurace LSUV využívající 16 obrázků z trénovací sady MNIST.

Tyto výsledky jsou obdobné i pro databázi CIFAR-10. Rozdíl mezi dosaženými výsledky a výsledky v [32], může být dán použitím rozdílné aktivační funkce a architektury. Dále mohl být rozdíl způsoben, použitím rozdílných hyperparametrů sítě. Na základě experimentů, lze doporučit při použití metody inicializace vah LSUV konfiguraci, která využívá 16 obrázků z trénovací sady pro dosažení co nejlepších výsledků správnosti na testovací sadě.

Na základě proběhlých experimentů, lze říct, že se potvrdilo tvrzení z [2] a z [1] o vhodnosti použití malých náhodných hodnot blízkých nule jako počáteční váhy. Na základě experimentů, lze učinit doporučení, pro výběr optimálních parametrů pravděpodobnostního rozdělení, při použití metody náhodné inicializace vah z normálního nebo rovnoměrného pravděpodobnostního rozdělení, tak aby síť dosáhla konvergence. Hodnotu průměru u všech normálních pravděpodobnostních rozdělení je doporučeno nastavit na 0. Pro mělkou vícevrstvou perceptronovskou síť lze doporučit volbu hodnoty směrodatné odchylky normálního rozdělení z intervalu  $\langle 0,0005;0,2 \rangle$ . Stejně tak i hodnotu limitu u rovnoměrného rozdělení lze doporučit volit z intervalu  $\langle 0,0005;0,2 \rangle$ . Pro hlubokou vícevrstvou perceptronovskou síť lze doporučit volbu směrodatné odchylky pro normální rozdělení z intervalu  $\langle 0,2;0,05 \rangle$ . Stejný interval lze doporučit i pro volbu limitu rovnoměrného rozdělení. U jednoduché konvoluční neuronové sítě lze doporučit volbu hodnoty směrodatné odchylky normálního rozdělení z intervalu  $\langle 0,2;0,05 \rangle$ . Pro volbu limitu rovnoměrného rozdělení lze doporučit volit hodnoty z intervalu  $\langle 0,2;0,05 \rangle$ . U hluboké konvoluční neuronové sítě – ResNet18 lze doporučit volbu směrodatné odchylky normálního rozdělení v intervalu  $\langle 0,0005;0,2 \rangle$ . Ze stejného intervalu je lze doporučit volit limit rovnoměrného rozdělení.



# Závěr

Cílem bakalářské práce bylo seznámení se s metodami inicializace vah neuronových sítí a jejich vlivem na učení sítě. Nejprve bylo zapotřebí nastudovat problematiku a vypracovat rešerši, která se zabývá umělými neuronovými sítěmi a jejich učením a dále metodami inicializace vah. Následně bylo za úkol navrhnut a implementovat experimenty, na kterých byl ověřen vliv různých metod inicializace vah na učení sítě.

V teoretické části práce byla provedena literární rešerše témat jednovrstvé dopředné sítě aktivační funkce, chybové funkce, optimalizačních algoritmů, vícevrstvé dopředné sítě, konvoluční sítě. Další kapitola popisuje rozdělení metod inicializace vah do několika kategorií v závislosti na přístupu při inicializaci. Následuje popsání metrik, kterými bylo hodnoceno učení sítě, formulace problému a popis použitých databází MNIST a CIFAR-10.

Praktická část se zabývá navrhnutím experimentů, předzpracováním datasetů, popisem použitých neuronových sítí a jejich hyperparametrů. Následoval popis implementace navržených experimentů a popis statistického vyhodnocení. Úvodní experiment, testoval předpoklad o nevhodnosti inicializace vah na nulu. V dalších experimentech se testoval vliv vybraných metod, které představují rozdílné přístupy v inicializaci vah. Vybrané metody jsou LSUV, náhodná inicializace vah z normálního a rovnoměrného pravděpodobnostního rozložení, He, Lecun a Glorot. U těchto vybraných metod jsou také zkoumány rozdíly ve vlivu na učení sítí mezi různými konfiguracemi těchto metod. Vybrané architektury jsou mělká a hluboká vícevrstvá perceptronová síť, jednoduchá konvoluční neuronová síť a hluboká konvoluční neuronová síť – ResNet18.

V poslední části jsou uvedeny a porovnány dosažené výsledky pro dvě databáze MNIST a CIFAR-10 a různé architektury neuronových sítí. Nejvyšší správnosti na testovací sadě databáze MNIST, konkrétně 99,400 %, dosáhla síť ResNet18 využívající metodu inicializace vah pomocí rozkladu na ortonormální bázi. Nejlepších výsledků správnosti na testovací sadě, konkrétně 72,380 %, pro databázi CIFAR-10 bylo dosaženo sítí ResNet18, jejíž váhy byly inicializovány metodou inicializace vah z normálního rozložení s parametry rozložení ( $\mu = 0, \sigma = 0,005$ ). Dále bylo provedeno srovnání architektur a metod inicializace, se kterými bylo dosaženo nejrychlejšího optimálního naučení. Metoda, díky které bylo dosahováno nejrychlejší konvergence sítě napříč použitými databázemi a architekturami, je LSUV. Následuje diskuse a srovnání dosažených výsledků s literaturou. V diskusi jsou na základě experimentů uvedena doporučení.

# Literatura

- [1] *CS231n: Convolutional Neural Networks for Visual Recognition*. Online. Dostupné z: <https://cs231n.github.io/>. [cit. 2023-12-10].
- [2] KHAN, Muhammad Aamir, Aunsia KHAN, Muhammad Nasir KHAN a Sajid ANWAR. *A novel learning method to classify data streams in the internet of things*. In: 2014 National Software Engineering Conference. Online. IEEE, 2014, s. 61-66 [cit. 2023-12-05]. ISBN 978-1-4799-6161-0. Dostupné z: doi:10.1109/NSEC.2014.6998242
- [3] MOHRI, Mehryar, Afshin ROSTAMIZADEH a Ameet TALWALKAR. *Foundations of machine learning*. Cambridge: MIT Press, c2012. Adaptive computation and machine learning series. ISBN 9780262018258. [cit. 2023-12-14]
- [4] MARSLAND, Stephen. *Machine learning: an algorithmic perspective*. 2nd ed. Boca Raton: Chapman & Hall/CRC, [2015]. Chapman & Hall/CRC machine learning & pattern recognition series. ISBN 978-1-4665-8328-3. [cit. 2023-12-14]
- [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. [cit. 2023-12-14]
- [6] KOZUMPLÍK, Jiří; PROVAZNÍK, Ivo. *Umělá inteligence v medicíně. Umělá inteligence v medicíně..* Brno: FEKT VUT, 2007. [cit. 2023-11-30].
- [7] DA SILVA, Ivan Nunes, Rogerio ANDRADE FLAUZINO, Silas Franco DOS REIS ALVES, Danilo HERNANE SPATTI a Luisa Helena Bartocci LIBONI. *Artificial Neural Networks: A Practical Course*. Imprint: Springer, 2017. ISBN 978-3-319-43161-1. [cit. 2023-11-30].
- [8] FIALA, Pavel a Jiří VALENTA. *Přehled anatomie centrálního nervového systému*. Praha: Univerzita Karlova, nakladatelství Karolinum, 2020. ISBN 9788024644776. [cit. 2023-12-14]
- [9] AGGARWAL, Charu C. *Neural networks and deep learning: a textbook*. Cham: Springer, 2018. ISBN 978-3-319-94462-3. [cit. 2023-11-30].
- [10] MULLA, Mohammed Zeeshan. *Cost, Activation, Loss Function// Neural Network// Deep Learning. What are these?* Online. Dostupné z: <https://medium.com/@zeeshanmulla/cost-activation-loss-function-neural-network-deep-learning-what-are-these-91> [cit. 2023-12-10]

- [11] TERVEN, Juan; CORDOVA-ESPARZA, Diana-Margarita Cordova-Esparza; RAMIREZ-PEDRAZA, Alfonso a CHÁVEZ URBIOLA, Edgar Arturo. *Loss Functions and Metrics in Deep Learning*. Online. Dostupné z: <https://doi.org/10.48550/arXiv.2307.02694>. [cit. 2023-12-13].
- [12] OMAR, Amri, Fri MOHAMED, Msaaf MOHAMMED a Belmajdoub FOUAD. *The commonly used algorithms to optimize a neural network in supervised learning: Overview, and comparative study*. In: 2021 International Conference on Digital Age & Technological Advances for Sustainable Development (ICDATA) Online. IEEE, 2021, s. 31-38 [cit. 2023-12-14]. ISBN 978-1-6654-2901-6. Dostupné z: [doi:10.1109/ICDATA52997.2021.00015](https://doi.org/10.1109/ICDATA52997.2021.00015)
- [13] ABDULKADIROV, Ruslan; LYAKHOV, Pavel a NAGORNOV, Nikolay, 2023. *Survey of Optimization Algorithms in Modern Neural Networks*. Online. Mathematics. Roč. 2023, č. 11, s. 1-37. Dostupné z: <https://doi.org/https://doi.org/10.3390/math11112466> [cit. 2023-12-14].
- [14] RUDER, Sebastian. *An overview of gradient descent optimization algorithms*. Online. CoRR. Č. abs/1609.04747, article 1609.04747. Dostupné z: <https://doi.org/https://doi.org/10.48550/arXiv.1609.04747>. [cit. 2023-12-15].
- [15] NIELSEN, Michael Aaron. *Neural Networks and Deep Learning* Online. Determination Press, 2015 [cit. 2023-11-30]. Dostupné z: <http://neuralnetworksanddeeplearning.com/>
- [16] GOODFELLOW, Ian; BENGIO, Yoshua; COURVILLE, Aaron *Deep Learning*. Online. Cambridge, MA, USA: MIT Press, 2016. Dostupné z: <http://www.deeplearningbook.org> [cit. 2023-11-30].
- [17] MISHRA, Maynak. *Convolutional Neural Networks, Explained* Online. Dostupné z: <https://towardsdatascience.com/convolutional-neural-networks-explained-9cc5188c4939>. [cit. 2023-12-10].
- [18] AGUIRRE, Diego, 2019. *A Novel Set Of Weight Initialization Techniques For Deep Learning Architectures*. Online, Disertace, vedoucí Olac Fuentes. University of Texas at El Paso. Dostupné z: [https://scholarworks.utep.edu/open\\_etd/2822](https://scholarworks.utep.edu/open_etd/2822). [cit. 2023-12-29].
- [19] NARKHEDE, Meenal V.; BARTAKKE, Prashant P. a SUTAONE, Mukul S., 2022. *A review on weight initialization strategies for neural networks*. Online. Artificial Intelligence Review. Roč. 55, č. 1, s. 291-322. ISSN 0269-2821. Dostupné z: <https://doi.org/10.1007/s10462-021-10033-z> [cit. 2023-08-14].

- [20] WANCHEN, Li, 2020. *Analysis on the Weight initialization Problem in Fully-connected Multi-layer Perceptron Neural Network*. Online. In: 2020 International Conference on Artificial Intelligence and Computer Engineering (ICAICE). IEEE, s. 150-153. ISBN 978-1-7281-9146-1. Dostupné z: <https://doi.org/10.1109/ICAICE51518.2020.00035>. [cit. 2024-04-12].
- [21] LI, Huimin; Perin, Guilherme; KRČEK, Marina *A Comparison of Weight Initializers in Deep Learning-Based Side-Channel Analysis* Online. Applied Cryptography and Network Security Workshops. ACNS 2020. Lecture Notes in Computer Science(), 12418. Springer Cham. ISBN 978-3-030-61638-0. Dostupné z: [https://doi.org/10.1007/978-3-030-61638-0\\_8](https://doi.org/10.1007/978-3-030-61638-0_8) [cit. 2023-12-29]
- [22] NGUYEN, D. a WIDROW, B., 1990. *Improving the learning speed of 2-layer neural networks by choosing initial values of the adaptive weights*. Online. In: 1990 IJCNN International Joint Conference on Neural Networks. IEEE, 21-26 vol.3. Dostupné z: <https://doi.org/10.1109/IJCNN.1990.137819>. [cit. 2024-04-22].
- [23] HERNANDEZ-ESPINOSA, C. a FERNANDEZ-REDONDO, M., 2001. *Multilayer feedforward weight initialization*. Online. In: IJCNN'01. International Joint Conference on Neural Networks. Proceedings (Cat. No.01CH37222). IEEE, s. 166-170. ISBN 0-7803-7044-9. Dostupné z: <https://doi.org/10.1109/IJCNN.2001.939011>. [cit. 2024-04-13].
- [24] LIU, Jingjing; LIU, Yefeng a ZHANG, Qichun, 2022. *A weight initialization method based on neural network with asymmetric activation function*. Online. Neurocomputing. Roč. 483, s. 171-182. ISSN 09252312. Dostupné z: <https://doi.org/10.1016/j.neucom.2022.01.088>. [cit. 2024-04-13].
- [25] QIAO, Junfei; LI, Sanyi a LI, Wenjing, 2016. *Mutual information based weight initialization method for sigmoidal feedforward neural networks*. Online. Neurocomputing. Roč. 207, s. 676-683. ISSN 09252312. Dostupné z: <https://doi.org/10.1016/j.neucom.2016.05.054>. [cit. 2024-04-13].
- [26] THIMM, G. a FIESLER, E., 1997. *High-order and multilayer perceptron initialization*. Online. *IEEE Transactions on Neural Networks*. Roč. 8, č. 2, s. 349-359. ISSN 1045-9227. Dostupné z: <https://doi.org/10.1109/72.557673>. [cit. 2024-04-22].
- [27] BOULILA, Wadii; DRISS, Maha; ALSHANQITI, Eman; AL-SAREM, Mohamed; SAEED, Faisal et al., 2022. *Weight Initialization Techniques for Deep*

- Learning Algorithms in Remote Sensing: Recent Trends and Future Perspectives*. Online. In: *Advances on Smart and Soft Computing. Advances in Intelligent Systems and Computing*. 1399. Springer Singapore, s. 477–484. ISBN 978-981-16-5559-3. Dostupné z: [https://doi.org/10.1007/978-981-16-5559-3\\_39](https://doi.org/10.1007/978-981-16-5559-3_39) [cit. 2023-12-29]
- [28] KAPOOR, Namrata. <https://www.numpyninja.com/post/weight-initialization-techniques>. Online. Dostupné z: <https://www.numpyninja.com/post/weight-initialization-techniques>. [cit. 2023-12-10].
- [29] GLOT, Xavier a BENGIO, Yoshua, 2010. *Understanding the difficulty of training deep feedforward neural networks*. Online. In: TEH, Yee Whye a TITTERINGTON, Mike. *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*. 9. Chia Laguna Resort, Sardinia, Italy: PMLR, s. 249-256. Dostupné z: <http://proceedings.mlr.press/v9/glorot10a/glorot10a.pdf>. [cit. 2024-04-13].
- [30] HE, Kaiming; ZHANG, Xiangyu; REN, Shaoqing a SUN, Jian, 2015. *Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification*. Online. In: *2015 IEEE International Conference on Computer Vision (ICCV)*. IEEE, s. 1026-1034. ISBN 978-1-4673-8391-2. Dostupné z: <https://doi.org/10.1109/ICCV.2015.123>. [cit. 2024-04-22].
- [31] SAXE, Andrew M.; MCCLELLAND, James L. a GANGULI, Surya. *Exact solutions to the nonlinear dynamics of learning in deep linear neural networks*. Online. In: *CoRR*. Abs/1312.6120. Dostupné z: <https://doi.org/10.48550/arXiv.1312.6120>. [cit. 2024-04-23].
- [32] MISHKIN, Dmytro a MATAS, Jiří. *All you need is good init*. Online. In: *4th International Conference On Learning Representations*. San Juan, Porto Rico. Dostupné z: <http://arxiv.org/abs/1511.06422>. [cit. 2024-04-28].
- [33] TIRUMALA, Sreenivas Sremath. *A Novel Weights of Weights Approach for Efficient Transfer Learning in Artificial Neural Networks*. Online. *Procedia Computer Science*. Roč. 2022, č. 212, s. 295-303. Dostupné z: <https://doi.org/https://doi.org/10.1016/j.procs.2022.11.013>. [cit. 2024-01-02].
- [34] NEYSHABUR, Behnam; HANIE, Sedghi a CHIYUAN, Zhang. *What is being transferred in transfer learning?* Online. *Proceedings of the 34th International Conference on Neural Information Processing Systems*. 44. Roč. 2020, č. NIPS '20, article 44, s. 512–523. ISBN 9781713829546. Dostupné z: <https://doi.org/10.48550/arXiv.2008.11687>. [cit. 2024-04-24].

- [35] FOSS, Vicki, 2018. *Multiclass Classification Model Evaluation An overview of evaluation metrics for a multiclass machine-learning model*. Online. Aktualizováno 24.8.2019. Dostupné z: <https://parasite.id/blog/2018-12-13-model-evaluation/>. [cit. 2024-01-03].
- [36] SAXENA, Saurabh, 2022. *Multi-class Model Evaluation with Confusion Matrix and Classification Report*. Online. Aktualizováno 30.9.2022. Dostupné z: <https://towardsai.net/p/1/multi-class-model-evaluation-with-confusion-matrix-and-classification-report> [cit. 2024-01-03].
- [37] GRANDINI, Margherita; BAGLI, Enrico a VISANI, Giorgio, 2020. *Metrics for Multi-Class Classification: an Overview*, Online. In: ArXiv. Abs/2008.05756. Dostupné z: <https://arxiv.org/pdf/2008.05756>. [cit. 2024-01-03].
- [38] WONG, Kit; DORNBERGER, Rolf a HANNE, Thomas. *An analysis of weight initialization methods in connection with different activation functions for feedforward neural networks*. Online. Evolutionary Intelligence. Roč. 2022. Dostupné z: <https://doi.org/https://doi.org/10.1007/s12065-022-00795-y> [cit. 2023-12-29].
- [39] LECUN, Yann; CORTES, Corinna; BURGESS, Chris. *MNIST handwritten digit database* Dostupné z: <http://yann.lecun.org/exdb/mnist/>. [cit. 2023-12-29]
- [40] ChatGPT, verze z 13.5., OpenAI, 10. 5. 2024. Dostupné z: <https://chatgpt.com/share/eb970354-4f40-4225-ac5b-b4aaf371f6a2>. [cit. 2024-05-18].
- [41] KRIZHEVSKY, Alex a HINTON, Geoffrey, 2009. *Learning multiple layers of features from tiny images*. Online. In: . Toronto, Ontario: Technical report, University of Toronto. Dostupné z: <https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf>. [cit. 2024-05-11].
- [42] The CIFAR-10 dataset. Online. Dostupné z: <https://www.cs.toronto.edu/~kriz/cifar.html> [cit. 2024-05-11].
- [43] LI DENG, 2012. *The MNIST Database of Handwritten Digit Images for Machine Learning Research* [Best of the Web]. Online. IEEE Signal Processing Magazine. Roč. 29, č. 6, s. 141-142. ISSN 1053-5888. Dostupné z: <https://doi.org/10.1109/MSP.2012.2211477>. [cit. 2024-05-17].
- [44] *Datasets*. Online. <https://paperswithcode.com/datasets> [cit. 2024-05-11].

- [45] HE, Kaiming; ZHANG, Xiangyu; REN, Shaoqing a SUN, Jian, 2015. *Deep Residual Learning for Image Recognition* . Online. Dostupné z: <https://doi.org/10.48550/arXiv.1512.03385>. [cit. 2024-05-18].
- [46] *ResNet18-Pytorch*. Online. Dostupné z: <https://github.com/samcw/ResNet18-Pytorch> [cit. 2024-05-18].
- [47] ChatGPT, verze z 13.5., OpenAI, 10. 5. 2024, Dostupné z: <https://chatgpt.com/share/668ba22e-fb7f-4114-8fe7-77974fee421d>. [cit. 2024-05-18].
- [48] KINGMA, Diederik P. a BA, Jimmy, 2017. *Adam: A Method for Stochastic Optimization* . Online. In: . Abs/1412.6980. ArXiv. Dostupné z: <https://doi.org/10.48550/arXiv.1412.6980>. [cit. 2024-05-06].
- [49] MISHKIN, Dmytrom, 2017 *LSUV-keras*. Online. Dostupné z: <https://github.com/ducha-aiki/LSUV-keras> [cit. 2024-05-18].
- [50] PEDREGOSA, Fabian; VAROQUAUX, Gaël; GRAMFORT, Alexandre; MICHEL, Vincent; THIRION, Bertrand et al., 2011. *Scikit-learn: Machine Learning in Python*. In: Journal of Machine Learning Research. 12. S. 2825-2830. [cit. 2024-05-19].
- [51] ChatGPT, verze z 13.5., OpenAI, 10. 5. 2024, Dostupné z: <https://chatgpt.com/share/bede2764-05c6-4198-8610-24695254469d>. [cit. 2024-05-19].
- [52] DESAI, Chitra, 2024. *Impact of Weight Initialization Techniques on Neural Network Efficiency and Performance: A Case Study with MNIST Dataset*. Online. In: International Journal of Engineering and Computer Science. 13(04). S. 26115–26120. Dostupné z: <https://doi.org/10.18535/ijecs/v13i04.4809>. [cit. 2024-05-27].
- [53] MEHROTRA, Kishan, Chilukuri K. MOHAN a Sanjay RANKA. *Elements of artificial neural networks*. Cambridge, Mass.: MIT Press, c1997. ISBN 0-262-13328-8. [cit. 2023-11-30].

## Seznam symbolů a zkratek

<b>ReLU</b>	Rectified Linear Unit
<b>CCE</b>	Categorical Cross Entropy
<b>SGD</b>	Stochastic Gradient Descent
<b>Adam</b>	Adaptive moment estimate
<b>CNN</b>	Convolutional Neural Network
<b>LSUV</b>	Layer-Sequential Unit-Variance
<b>TP</b>	True Positive
<b>TN</b>	True Negative
<b>FP</b>	False Positive
<b>FN</b>	False Negative
<b>MNIST</b>	Modified National Institute of Standards and Technology database
<b>CIFAR</b>	Canadian Institute for Advanced Research



# A Výsledky pro databázi MNIST

## A.1 Mělká vícevrstvá perceptronová síť

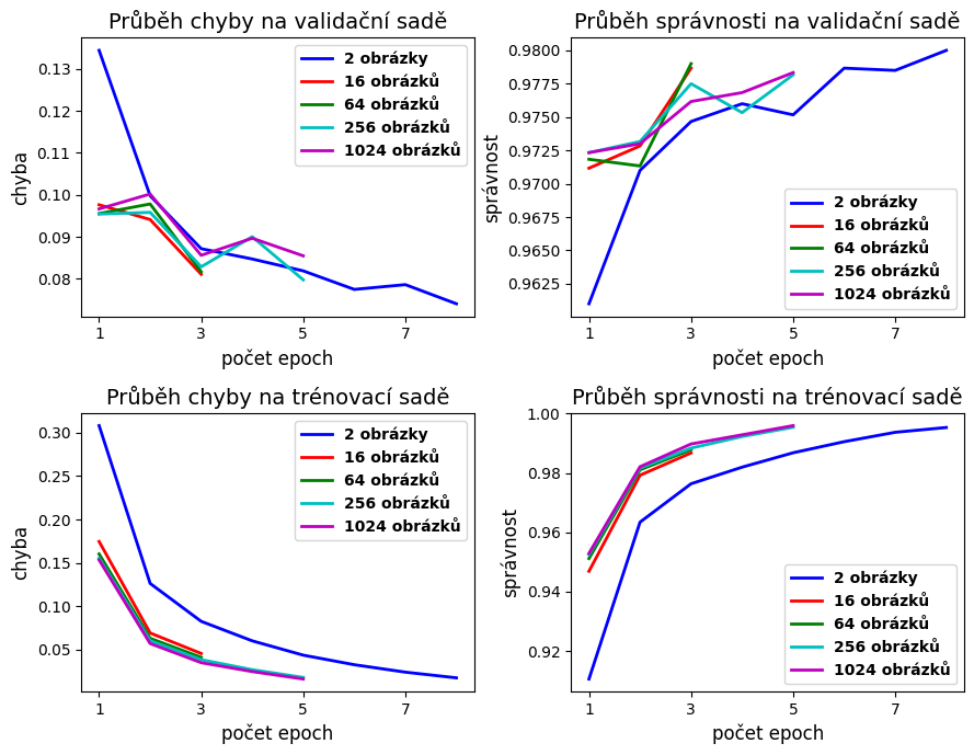
### A.1.1 LSUV

Tab. A.1: Výsledky ohodnocující učení mělké vícevrstvé perceptronové sítě, která byla inicializována metodou LSUV v různých konfiguracích a poté učena na trénovací sadě databáze MNIST. Hodnoty jsou získány pro testovací sadu databáze MNIST. Šestý sloupec vyjadřuje počet epoch k dosažení optimálního naučení sítě. Žluté podbarvení vyjadřuje nejvyšší dosaženou správnost a oranžové podbarvení nejnižší počet epoch.

Metoda LSUV					
typ konfigurace	správnost	průměrná vážená přesnost	průměrná vážená výtěžnost	průměrné vážené F1-skóre	počet epoch
2 obrázky	<b>98,030 %</b>	0,98	0,98	0,98	8
16 obrázků	97,730 %	0,98	0,98	0,98	<b>3</b>
64 obrázků	97,620 %	0,98	0,98	0,98	<b>3</b>
256 obrázků	97,760 %	0,98	0,98	0,98	5
1024 obrázků	97,710 %	0,98	0,98	0,98	5

Tab. A.2: Popis počtu epoch k dosažení 85 %, 90 %, 95 %, 96 %, 97 %, 98 % a 99 % správnosti na validační sadě při učení mělké vícevrstvé perceptronové sítě, která byla inicializována metodou LSUV. Hodnoty počtu epoch jsou vyobrazeny pro 5 různých konfigurací metody. Znak "-" značí nedosažení dané správnosti. Nejlepší výsledek je podbarven modrou.

typ konfigurace	Počet epoch k dosažení určité správnosti na validační sadě						
	85 %	90 %	95 %	96 %	97 %	98 %	99 %
2 obrázky	1	1	1	1	2	8	-
16 obrázků	1	1	1	1	1	13	-
64 obrázků	1	1	1	1	1	-	-
256 obrázků	1	1	1	1	1	-	-
1024 obrázků	1	1	1	1	1	11	-



Obr. A.1: Průběhy chyby a správnosti na validační a trénovací sadě databáze MNIST při učení mělké vícevrstvé perceptronové sítě pro 5 různých konfigurací metody inicializace vah LSUV. Průběhy jsou zobrazeny do epochy optimálního naučení.

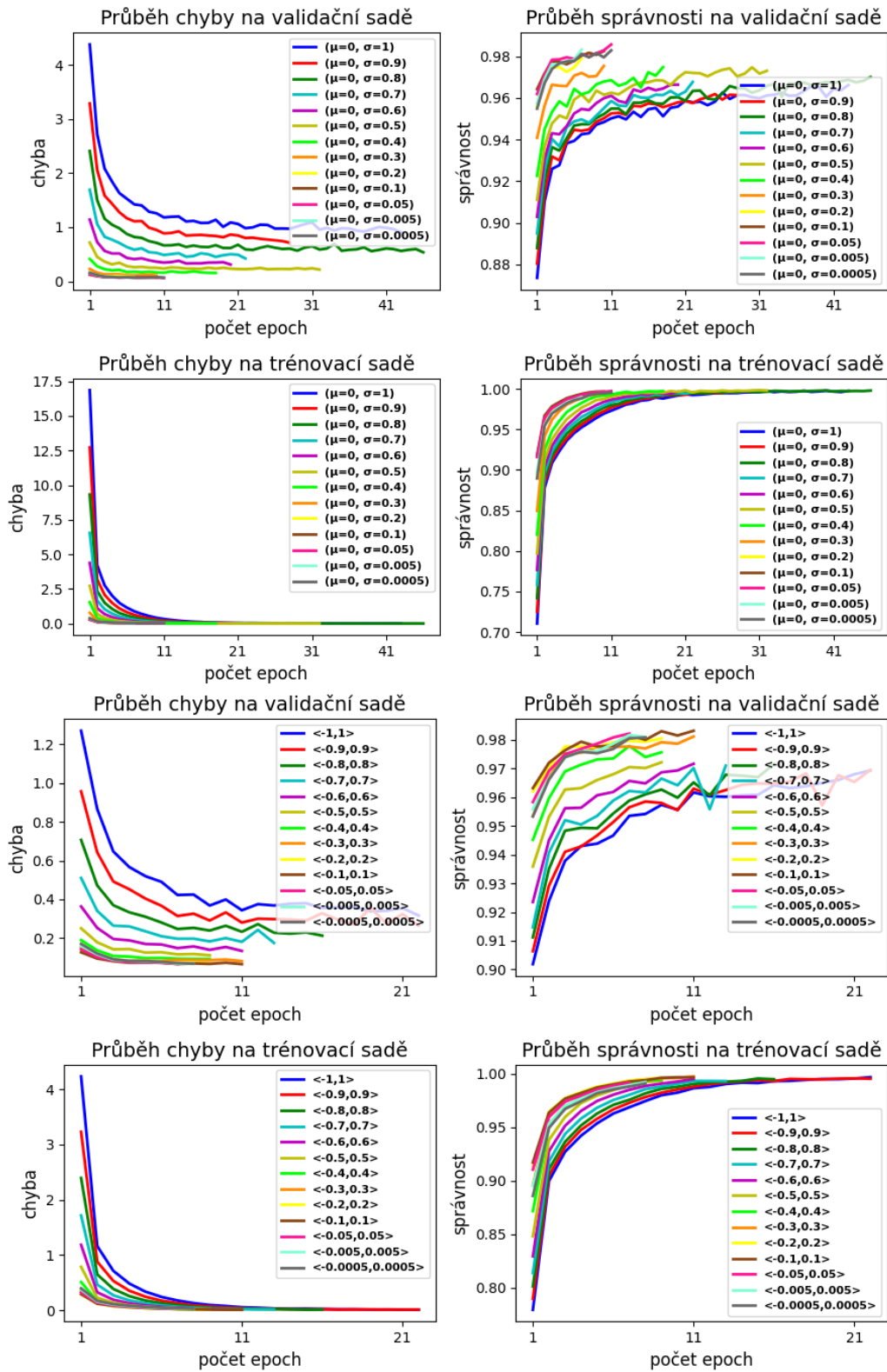
## A.1.2 Náhodná inicializace vah z normálního a rovnoměrného pravděpodobnostního rozdělení

Tab. A.3: Výsledky ohodnocující učení mělké vícevrstvé perceptronové sítě. Zvolená inicializace vah byla inicializace vah z normálního pravděpodobnostního rozdělení v různých konfiguracích. Síť byla učena na trénovací sadě databáze MNIST. Hodnoty jsou získány pro testovací sadu databáze MNIST. Šestý sloupec vyjadřuje počet epoch k dosažení optimálního naučení sítě. Žluté podbarvení vyjadřuje nejvyšší dosaženou správnost a oranžové podbarvení nejnižší počet epoch.

Metoda inicializace z normálního pravděpodobnostního rozdělení					
typ konfigurace	správnost	průměrná vážená přesnost	průměrná vážená výtěžnost	průměrné vážené F1-skóre	počet epoch
$(\mu = 0, \sigma = 1)$	96,230 %	0,96	0,96	0,96	43
$(\mu = 0, \sigma = 0,9)$	95,780 %	0,96	0,96	0,96	28
$(\mu = 0, \sigma = 0,8)$	96,480 %	0,97	0,96	0,96	46
$(\mu = 0, \sigma = 0,7)$	95,830 %	0,96	0,96	0,96	22
$(\mu = 0, \sigma = 0,6)$	96,180 %	0,96	0,96	0,96	20
$(\mu = 0, \sigma = 0,5)$	97,030 %	0,97	0,97	0,97	32
$(\mu = 0, \sigma = 0,4)$	97,009 %	0,97	0,97	0,97	18
$(\mu = 0, \sigma = 0,3)$	97,120 %	0,97	0,97	0,97	10
$(\mu = 0, \sigma = 0,2)$	97,200 %	0,97	0,97	0,97	<b>7</b>
$(\mu = 0, \sigma = 0,1)$	98,020 %	0,98	0,98	0,98	10
$(\mu = 0, \sigma = 0,05)$	<b>98,100 %</b>	0,98	0,98	0,98	<b>11</b>
$(\mu=0, \sigma=0,005)$	97,800 %	0,98	0,98	0,98	<b>7</b>
$(\mu = 0, \sigma = 0,0005)$	97,850 %	0,98	0,98	0,98	11

Tab. A.4: Popis počtu epoch k dosažení určité správnosti na validační sadě při učení mělké vícevrstvé perceptronové sítě, inicializované metodou náhodného inicializace vah z normálního pravděpodobnostního rozdělení v různých konfiguracích. Hodnoty počtu epoch jsou vyobrazeny pro 13 různých konfigurací metody. Znak "-" značí nedosažení dané správnosti. Nejlepší výsledek je podbarven modrou.

typ konfigurace	Počet epoch k dosažení určité správnosti na validační sadě						
	85 %	90 %	95 %	96 %	97 %	98 %	99 %
$(\mu = 0, \sigma = 1)$	1	2	11	26	-	-	-
$(\mu = 0, \sigma = 0,9)$	1	2	10	24	-	-	-
$(\mu = 0, \sigma = 0,8)$	1	2	9	19	46	-	-
$(\mu = 0, \sigma = 0,7)$	1	2	9	14	-	-	-
$(\mu = 0, \sigma = 0,6)$	1	1	6	11	30	-	-
$(\mu = 0, \sigma = 0,5)$	1	1	4	7	21	-	-
$(\mu = 0, \sigma = 0,4)$	1	1	3	6	16	-	-
$(\mu = 0, \sigma = 0,3)$	1	1	2	3	6	-	-
$(\mu = 0, \sigma = 0,2)$	1	1	1	2	3	-	-
$(\mu = 0, \sigma = 0,1)$	1	1	1	1	2	7	-
$(\mu = 0, \sigma = 0,05)$	1	1	1	1	2	7	-
$(\mu = 0, \sigma = 0,005)$	1	1	1	2	3	7	-
$(\mu = 0, \sigma = 0,0005)$	1	1	1	2	3	7	-



Obr. A.2: Průběhy chyby a správnosti na validační a trénovací sadě databáze MNIST při učení mělké vícevrstvé perceptronové sítě pro konfigurace metod inicializace vah z normálního a rovnoměrného pravděpodobnostního rozdělení. Horní čtyři grafy vyjadřují průběhy pro inicializace vah z normálního pravděpodobnostního rozdělení a čtyři spodní grafy vyjadřují průběhy pro inicializace vah z rovnoměrného pravděpodobnostního rozdělení. Průběhy jsou zobrazeny do epochy optimálního naučení.

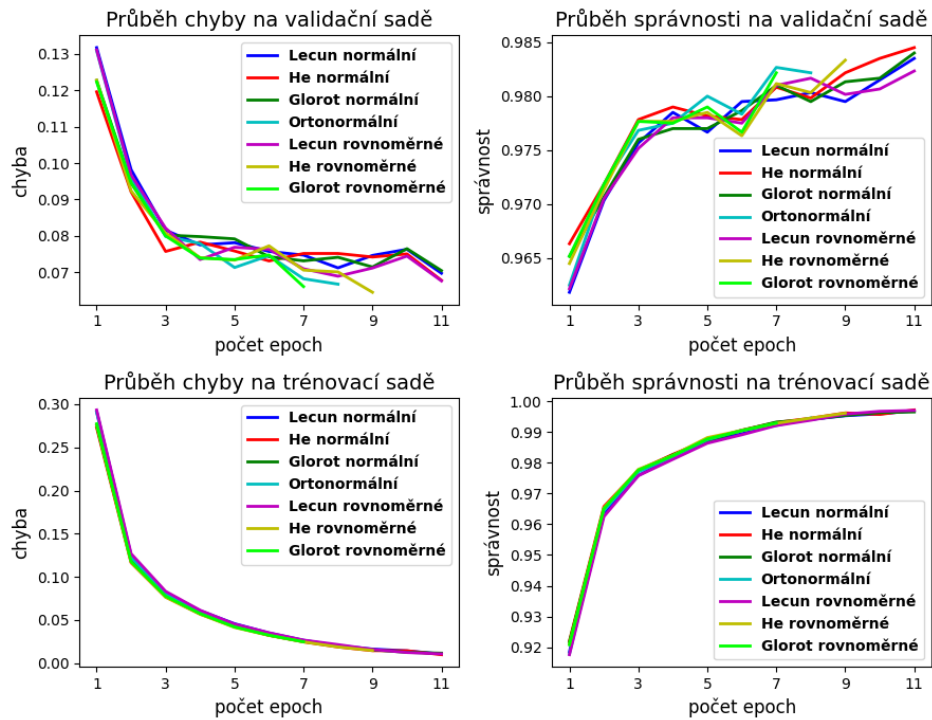
Tab. A.5: Výsledky ohodnocující učení mělké vícevrstvé perceptronové sítě. Zvolená inicializace vah byla inicializace vah z rovnoměrného pravděpodobnostního rozdělení v různých konfiguracích. Síť byla učena na trénovací sadě databáze MNIST. Hodnoty jsou získány pro testovací sadu databáze MNIST. Šestý sloupec vyjadřuje počet epoch k dosažení optimálního naučení sítě. Žluté podbarvení vyjadřuje nejvyšší dosaženou správnost a oranžové podbarvení nejnižší počet epoch.

Metoda inicializace z rovnoměrného pravděpodobnostního rozdělení					
typ konfigurace	správnost	průměrná vážená přesnost	průměrná vážená výtěžnost	průměrné vážené F1-skóre	počet epoch
<-1,1>	96,720 %	0,97	0,97	0,97	22
<-0,9;0,9>	96,830 %	0,97	0,97	0,97	22
<-0,8;0,8>	96,720 %	0,97	0,97	0,97	16
<-0,7;0,7>	96,820 %	0,97	0,97	0,97	13
<-0,6;0,6>	96,800 %	0,97	0,97	0,97	11
<-0,5;0,5>	97,220 %	0,97	0,97	0,97	9
<-0,4;0,4>	97,520 %	0,97	0,97	0,97	9
<-0,3;0,3>	97,870 %	0,98	0,98	0,98	11
<-0,2;0,2>	97,940 %	0,98	0,98	0,98	9
<-0,1;0,1>	97,880 %	0,98	0,98	0,98	11
<-0,05;0,05>	97,930 %	0,98	0,98	0,98	7
<-0,005;0,005>	97,960 %	0,98	0,98	0,98	8
<-0,0005;0,0005>	97,950 %	0,98	0,98	0,98	8

Tab. A.6: Popis počtu epoch k dosažení určité správnosti na validační sadě při učení mělké vícevrstvé perceptronové sítě, inicializované metodou náhodného inicializace vah z rovnoměrného pravděpodobnostního rozdělení v různých konfiguracích. Hodnoty počtu epoch jsou vyobrazeny pro 13 různých konfigurací metody. Znak "-" značí nedosažení dané správnosti. Nejlepší výsledek je podbarven modrou.

typ konfigurace	Počet epoch k dosažení určité správnosti na validační sadě						
	85 %	90 %	95 %	96 %	97 %	98 %	99 %
<-1,1>	1	1	7	11	29	-	-
<-0,9;0,9>	1	1	6	11	24	-	-
<-0,8;0,8>	1	1	6	8	16	-	-
<-0,7;0,7>	1	1	3	7	11	-	-
<-0,6;0,6>	1	1	3	5	11	-	-
<-0,5;0,5>	1	1	2	3	7	-	-
<-0,4;0,4>	1	1	2	3	4	19	-
<-0,3;0,3>	1	1	1	2	3	11	-
<-0,2;0,2>	1	1	1	1	2	9	-
<-0,1;0,1>	1	1	1	1	2	7	-
<-0,05;0,05>	1	1	1	2	3	6	-
<-0,005;0,005>	1	1	1	2	3	7	-
<-0,0005;0,0005>	1	1	1	2	3	7	-

### A.1.3 Ostatní metody náhodné inicializace



Obr. A.3: Průběhy chyby a správnosti na validační a trénovací sadě databáze MNIST při učení mělké vícevrstvé perceptronové sítě, která byla inicializována vybranými metodami náhodné inicializace. Zvolené inicializace vah byly LeCun, He, Glorot, které se dále dělí podle toho, z jakého pravděpodobnostního rozdělení jsou váhy vzorkovány na rovnoměrné a normální. Dále byla použita metoda inicializaci vah pomocí rozkladu na ortonormální bázi. Průběhy jsou zobrazeny do epochy optimálního naučení.

Tab. A.7: Výsledky ohodnocující učení mělké vícevrstvé perceptronové sítě. Zvolené inicializace vah byly LeCun, He, Glorot, které se dále dělí podle toho, z jakého pravděpodobnostního rozdělení jsou váhy vzorkovány na rovnoměrné a normální. Dále byla použita s metodou inicializaci vah pomocí rozkladu na ortonormální bázi. Sít byla učena na trénovací sadě databáze MNIST. Hodnoty jsou získány pro testovací sadu databáze MNIST. Šestý sloupec vyjadřuje počet epoch k dosažení optimálního naučení sítě. Žluté podbarvení vyjadřuje nejvyšší dosaženou správnost a oranžové podbarvení nejnižší počet epoch.

Ostatní metody náhodné inicializace					
typ metody	správnost	průměrná vážená přesnost	průměrná vážená výtěžnost	průměrné vážené F1-skóre	počet epoch
LeCun rovnoměrné	98,020 %	0,98	0,98	0,98	11
LeCun normální	98,100 %	0,98	0,98	0,98	11
He rovnoměrné	97,880 %	0,98	0,98	0,98	9
He normální	<b>98,140 %</b>	0,98	0,98	0,98	11
Glorot rovnoměrné	97,860 %	0,99	0,98	0,98	<b>7</b>
Glorot normální	98,100 %	0,98	0,98	0,98	11
Ortonormální	97,940 %	0,98	0,98	0,98	8

Tab. A.8: Popis počtu epoch k dosažení určité správnosti na validační sadě při učení mělké vícevrstvé perceptronové sítě, která byla inicializována vybranými metodami náhodné inicializace. Zvolené inicializace vah byly LeCun, He, Glorot, které se dále dělí podle toho, z jakého pravděpodobnostního rozdělení jsou váhy vzorkovány na rovnoměrné a normální. Dále byla použita metoda inicializaci vah pomocí rozkladu na ortonormální bázi. Hodnoty počtu epoch jsou vyobrazeny pro 4 metody a jejich případné konfigurace. Znak "-" značí nedosažení dané správnosti. Nejlepší výsledek je podbarven modrou.

typ konfigurace	Počet epoch k dosažení určité správnosti na validační sadě						
	85 %	90 %	95 %	96 %	97 %	98 %	99 %
LeCun rovnoměrné	1	1	1	1	2	7	-
LeCun normální	1	1	1	1	2	8	-
He rovnoměrné	1	1	1	1	2	7	-
He normální	1	1	1	1	2	7	-
Glorot rovnoměrné	1	1	1	1	2	7	-
Glorot normální	1	1	1	1	2	7	-
Ortonormální	1	1	1	1	2	5	-

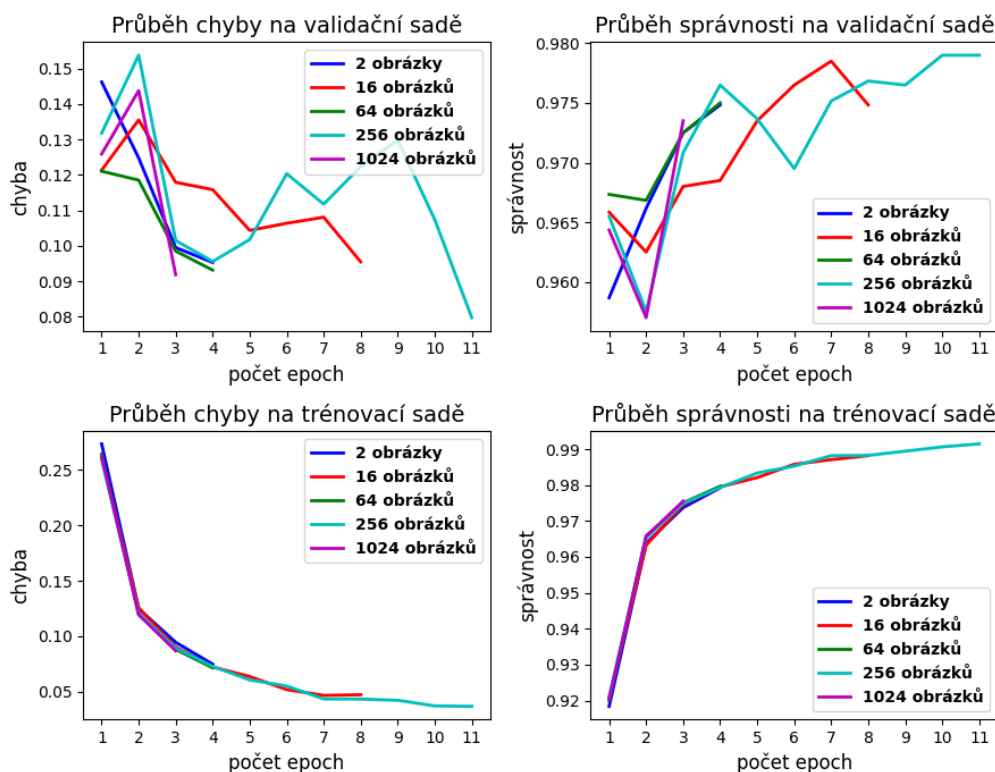
## A.2 Hluboká vícevrstvá perceptronová síť

### A.2.1 LSUV

Tab. A.9: Výsledky ohodnocující učení hluboké vícevrstvé perceptronové sítě, která byla inicializovaná metodou LSUV v různých konfiguracích a poté učena na trénovací sadě databáze MNIST. Hodnoty jsou získány pro testovací sadu databáze MNIST. Šestý sloupec vyjadřuje počet epoch k dosažení optimálního naučení sítě. Žluté podbarvení vyjadřuje nejvyšší dosaženou správnost a oranžové podbarvení nejnižší počet epoch.

Metoda LSUV					
typ konfigurace	správnost	průměrná vážená přesnost	průměrná vážená výtěžnost	průměrné vážené F1-skóre	počet epoch
2 obrázky	97,240 %	0,97	0,97	0,97	4
16 obrázků	<b>97,690 %</b>	0,98	0,98	0,98	8
64 obrázků	97,270 %	0,97	0,97	0,97	4
256 obrázků	97,540 %	0,98	0,98	0,98	11
1024 obrázků	97,180 %	0,97	0,97	0,97	<b>3</b>





Obr. A.4: Průběhy chyby a správnosti na validační a trénovací sadě databáze MNIST při učení hluboké vícevrstvé perceptronové sítě pro 5 různých konfigurací metody inicializace vah LSUV. Průběhy jsou zobrazeny do epochy optimálního naučení.

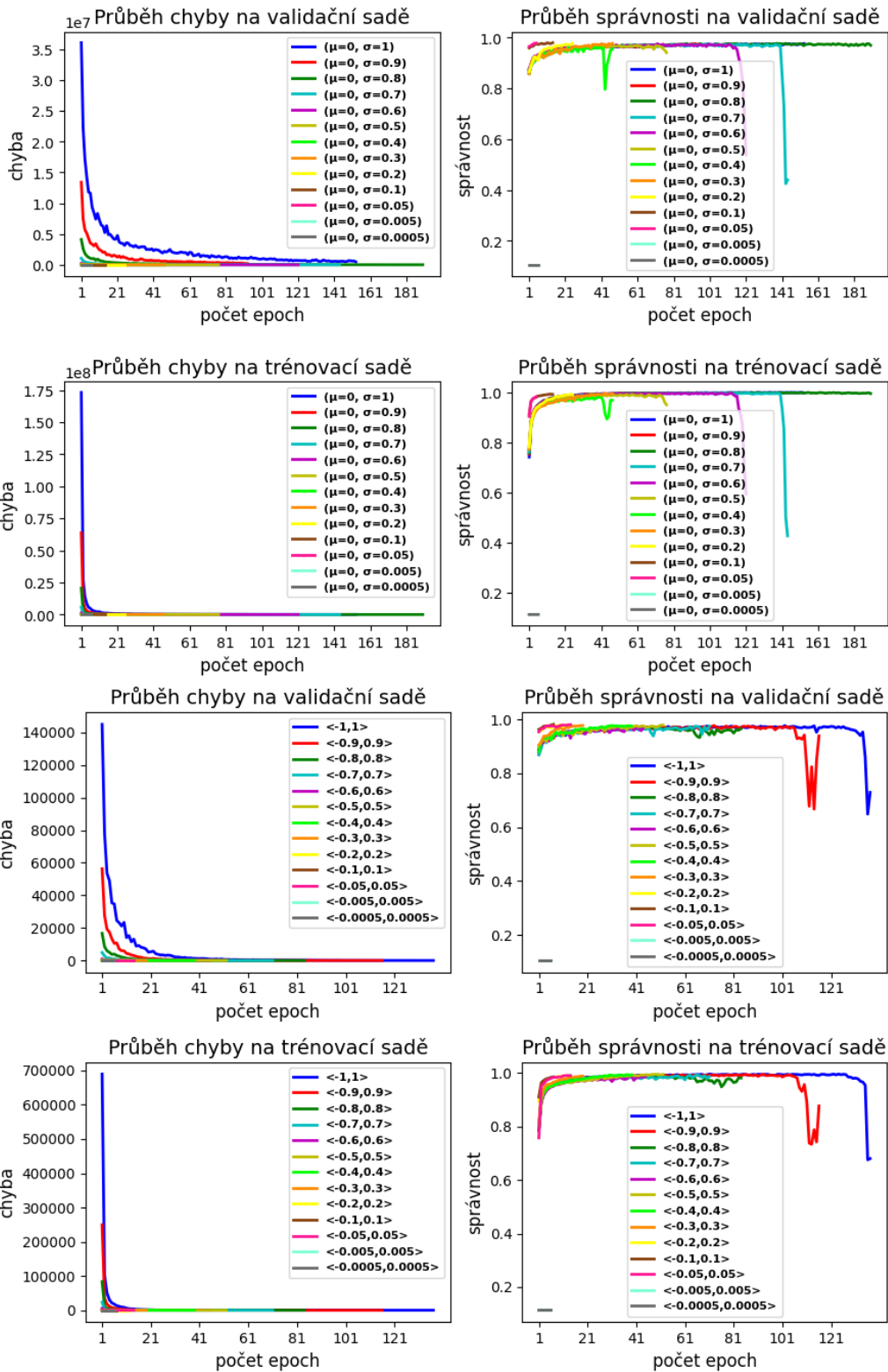
Tab. A.10: Popis počtu epoch k dosažení 85 %, 90 %, 95 %, 96 %, 97 %, 98 % a 99 % správnosti na validační sadě při učení hluboké vícevrstvé perceptronové sítě, které byla inicializovaná metodou LSUV. Hodnoty počtu epoch jsou vyobrazeny pro 5 různých konfigurací metody. Znak "-" značí nedosažení dané správnosti. Nejlepší výsledek je podbarven modrou.

typ konfigurace	Počet epoch k dosažení určité správnosti na validační sadě						
	85 %	90 %	95 %	96 %	97 %	98 %	99 %
2 obrázky	1	1	1	2	3	-	-
16 obrázků	1	1	1	1	5	-	-
64 obrázků	1	1	1	1	3	12	-
256 obrázků	1	1	1	1	3	14	-
1024 obrázků	1	1	1	1	3	-	-

## A.2.2 Náhodná inicializace vah z normálního a rovnoměrného pravděpodobnostního rozdělení

Tab. A.11: Výsledky ohodnocující učení hluboké vícevrstvé perceptronové sítě. Zvolená inicializace vah byla inicializace vah z normálního pravděpodobnostního rozložení v různých konfiguracích. Síť byla učena na trénovací sadě databáze MNIST. Hodnoty jsou získány pro testovací sadu databáze MNIST. Šestý sloupec vyjadřuje počet epoch k dosažení optimálního naučení sítě. Žluté podbarvení vyjadřuje nejvyšší dosaženou správnost a oranžové podbarvení nejnižší počet epoch.

Metoda inicializace z normálního pravděpodobnostního rozložení					
typ konfigurace	správnost	průměrná vážená přesnost	průměrná vážená výtěžnost	průměrné vážené F1-skóre	počet epoch
$(\mu = 0, \sigma = 1)$	97,380 %	0,97	0,97	0,97	153
$(\mu = 0, \sigma = 0,9)$	97,170 %	0,97	0,97	0,97	93
$(\mu = 0, \sigma = 0,8)$	96,770 %	0,97	0,97	0,97	200
$(\mu = 0, \sigma = 0,7)$	41,670 %	0,82	0,42	0,44	144
$(\mu = 0, \sigma = 0,6)$	53,170 %	0,70	0,53	0,52	121
$(\mu = 0, \sigma = 0,5)$	93,050 %	0,94	0,93	0,93	77
$(\mu = 0, \sigma = 0,4)$	94,660 %	0,95	0,95	0,95	47
$(\mu = 0, \sigma = 0,3)$	97,410 %	0,97	0,97	0,97	47
$(\mu = 0, \sigma = 0,2)$	97,500 %	0,97	0,97	0,97	25
$(\mu = 0, \sigma = 0,1)$	<b>97,850 %</b>	0,98	0,98	0,98	14
$(\mu = 0, \sigma = 0,05)$	97,550 %	0,98	0,98	0,98	<b>5</b>
$(\mu=0, \sigma=0,005)$	11,350 %	0,01	0,11	0,02	6
$(\mu = 0, \sigma = 0,0005)$	11,350 %	0,01	0,11	0,02	6



Obr. A.5: Průběhy chyby a správnosti na validační a trénovací sadě databáze MNIST při učení hluboké vícevrstvé perceptronové sítě pro konfigurace metod inicializace vah z normálního a rovnoměrného pravděpodobnostního rozložení. Horní čtyři grafy vyjadřují průběhy pro inicializace vah z normálního pravděpodobnostního rozložení a čtyři spodní grafy vyjadřují průběhy pro inicializace vah z rovnoměrného pravděpodobnostního rozložení. Průběhy jsou zobrazeny do epochy optimálního naučení.

Tab. A.12: Popis počtu epoch k dosažení 85 %, 90 %, 95 %, 96 %, 97 %, 98 % a 99 % správnosti na validační sadě při učení hluboké vícevrstvé perceptronové sítě, inicializované metodou náhodného inicializace vah z normálního pravděpodobnostního rozložení v různých konfiguracích. Hodnoty počtu epoch jsou vyobrazeny pro 13 různých konfigurací metody. Znak "-" značí nedosažení dané správnosti. Nejlepší výsledek je podbarven modrou.

typ konfigurace	Počet epoch k dosažení určité správnosti na validační sadě						
	85 %	90 %	95 %	96 %	97 %	98 %	99 %
$(\mu = 0, \sigma = 1)$	1	4	16	22	60	-	-
$(\mu = 0, \sigma = 0,9)$	1	3	15	21	49	-	-
$(\mu = 0, \sigma = 0,8)$	1	3	14	23	53	-	-
$(\mu = 0, \sigma = 0,7)$	1	3	14	22	58	-	-
$(\mu = 0, \sigma = 0,6)$	1	3	14	21	59	-	-
$(\mu = 0, \sigma = 0,5)$	1	3	18	27	-	-	-
$(\mu = 0, \sigma = 0,4)$	1	3	19	26	-	-	-
$(\mu = 0, \sigma = 0,3)$	1	3	16	25	34	-	-
$(\mu = 0, \sigma = 0,2)$	1	3	9	12	20	-	-
$(\mu = 0, \sigma = 0,1)$	1	1	1	2	4	14	-
$(\mu = 0, \sigma = 0,05)$	1	1	1	1	3	11	-
$(\mu = 0, \sigma = 0,005)$	-	-	-	-	-	-	-
$(\mu = 0, \sigma = 0,0005)$	-	-	-	-	-	-	-

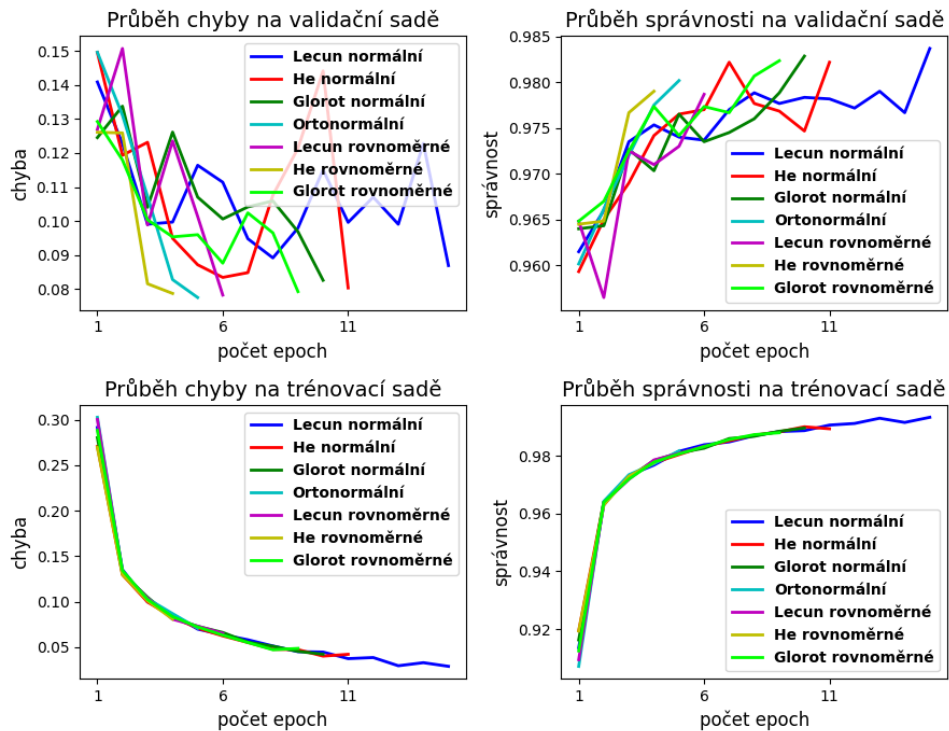
Tab. A.13: Výsledky ohodnocující učení hluboké vícevrstvé perceptronových sítí. Zvolená inicializace vah byla inicializace vah z rovnoměrného pravděpodobnostního rozložení v různých konfiguracích. Síť byla učena na trénovací sadě databáze MNIST. Hodnoty jsou získány pro testovací sadu databáze MNIST. Šestý sloupec vyjadřuje počet epoch k dosažení optimálního naučení sítě. Žluté podbarvení vyjadřuje nejvyšší dosaženou správnost a oranžové podbarvení nejnižší počet epoch.

Metoda inicializace z rovnoměrného pravděpodobnostního rozložení					
typ konfigurace	správnost	průměrná vážená přesnost	průměrná vážená výtěžnost	průměrné vážené F1-skóre	počet epoch
<-1,1>	69,760 %	0,82	0,70	0,72	137
<-0,9;0,9>	93,730 %	0,94	0,94	0,94	116
<-0,8;0,8>	96,270 %	0,96	0,96	0,96	84
<-0,7;0,7>	97,130 %	0,97	0,97	0,97	71
<-0,6;0,6>	96,950 %	0,97	0,97	0,97	45
<-0,5;0,5>	97,660 %	0,98	0,98	0,98	52
<-0,4;0,4>	97,630 %	0,98	0,98	0,98	39
<-0,3;0,3>	97,350 %	0,97	0,97	0,97	19
<-0,2;0,2>	<b>97,780 %</b>	0,98	0,98	0,98	12
<-0,1;0,1>	97,560 %	0,98	0,98	0,98	<b>7</b>
<-0,05;0,05>	97,670 %	0,98	0,98	0,98	14
<-0,005;0,005>	11,350 %	0,01	0,11	0,02	6
<-0,0005;0,0005>	11,350 %	0,01	0,11	0,02	6

Tab. A.14: Popis počtu epoch k dosažení určité správnosti na validační sadě při učení hluboké vícevrstvé perceptronové sítě, inicializované metodou náhodného inicializace vah z rovnoměrného pravděpodobnostního rozložení v různých konfiguracích. Hodnoty počtu epoch jsou vyobrazeny pro 13 různých konfigurací metody. Znak "-" značí nedosažení dané správnosti. Nejlepší výsledek je podbarven modrou.

typ konfigurace	Počet epoch k dosažení určité správnosti na validační sadě						
	85 %	90 %	95 %	96 %	97 %	98 %	99 %
<-1,1>	1	2	11	23	36	-	-
<-0,9;0,9>	1	2	11	17	32	-	-
<-0,8;0,8>	1	2	11	19	47	-	-
<-0,7;0,7>	1	3	12	16	57	-	-
<-0,6;0,6>	1	2	13	22	41	-	-
<-0,5;0,5>	1	2	10	20	31	-	-
<-0,4;0,4>	1	2	8	16	26	43	-
<-0,3;0,3>	1	1	6	8	10	25	-
<-0,2;0,2>	1	1	1	3	4	10	-
<-0,1;0,1>	1	1	1	1	3	7	-
<-0,05;0,05>	1	1	1	2	4	14	-
<-0,005;0,005>	-	-	-	-	-	-	-
<-0,0005;0,0005>	-	-	-	-	-	-	-

### A.2.3 Ostatní metody náhodné inicializace



Obr. A.6: Průběhy chyby a správnosti na validační a trénovací sadě databáze MNIST při učení hluboké vícevrstvé perceptronovské sítě, která byla inicializována vybranými metodami náhodné inicializace. Zvolené inicializace vah byly LeCun, He, Glorot, které se dále dělí podle toho, z jakého pravděpodobnostního rozložení jsou váhy vzorkovány na rovnoměrné a normální. Dále byla použita metoda inicializaci vah pomocí rozkladu na ortonormální bázi. Průběhy jsou zobrazeny do epochy optimálního naučení.

Tab. A.15: Výsledky ohodnocující učení hluboké vícevrstvé perceptronové sítě. Zvolené inicializace vah byly LeCun, He, Glorot, které se dále dělí podle toho, z jakého pravděpodobnostního rozložení jsou váhy vzorkovány na rovnoměrné a normální. Dále byla použita metoda inicializaci vah pomocí rozkladu na ortonormální bázi. Síť byla učena na trénovací sadě databáze MNIST. Hodnoty jsou získány pro testovací sadu databáze MNIST. Šestý sloupec vyjadřuje počet epoch k dosažení optimálního naučení sítě. Žluté podbarvení vyjadřuje nejvyšší dosaženou správnost a oranžové podbarvení nejnižší počet epoch.

Ostatní metody náhodné inicializace					
typ metody	správnost	průměrná vážená přesnost	průměrná vážená výtežnost	průměrné vážené F1-skóre	počet epoch
LeCun rovnoměrné	97,720 %	0,98	0,98	0,98	6
LeCun normální	<b>98,180 %</b>	0,98	0,98	0,98	15
He rovnoměrné	97,450 %	0,97	0,97	0,97	4
He normální	97,770 %	0,98	0,98	0,98	11
Glorot rovnoměrné	97,910 %	0,99	0,98	0,98	9
Glorot normální	97,970 %	0,98	0,98	0,98	10
Ortonormální	97,450 %	0,97	0,97	0,97	5

Tab. A.16: Popis počtu epoch k dosažení určité správnosti na validační sadě při učení hluboké vícevrstvé perceptronové sítě, která byla inicializována vybranými metodami náhodné inicializace. Zvolené inicializace vah byly LeCun, He, Glorot, které se dále dělí podle toho, z jakého pravděpodobnostního rozložení jsou váhy vzorkovány na rovnoměrné a normální. Dále byla použita metoda inicializaci vah pomocí rozkladu na ortonormální bázi. Hodnoty počtu epoch jsou vyobrazeny pro 4 metody a jejich případné konfigurace. Znak "-" značí nedosažení dané správnosti. Nejlepší výsledek je podbarven modrou.

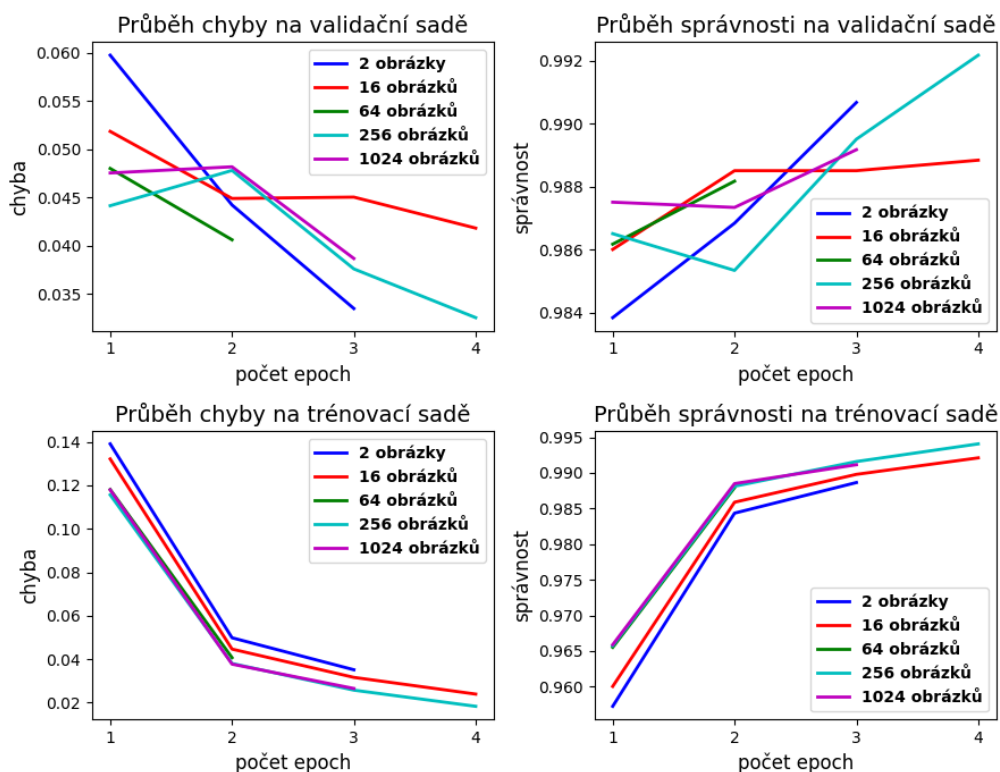
typ konfigurace	Počet epoch k dosažení určité správnosti na validační sadě						
	85 %	90 %	95 %	96 %	97 %	98 %	99 %
LeCun rovnoměrné	1	1	1	1	3	9	-
LeCun normální	1	1	1	1	3	15	-
He rovnoměrné	1	1	1	1	2	7	-
He normální	1	1	1	2	4	7	-
Glorot rovnoměrné	1	1	1	1	3	8	-
Glorot normální	1	1	1	1	3	10	-
Ortonormální	1	1	1	1	3	5	-

## A.3 Jednoduchá konvoluční neuronová síť

Tab. A.17: Výsledky ohodnocující učení jednoduché konvoluční neuronové sítě, která byla inicializovaná metodou LSUV v různých konfiguracích a poté učena na trénovací sadě databáze MNIST. Hodnoty jsou získány pro testovací sadu databáze MNIST. Šestý sloupec vyjadřuje počet epoch k dosažení optimálního naučení sítě. Žluté podbarvení vyjadřuje nejvyšší dosaženou správnost a oranžové podbarvení nejnižší počet epoch.

Metoda LSUV					
typ konfigurace	správnost	průměrná vážená přesnost	průměrná vážená výtěžnost	průměrné vážené F1-skóre	počet epoch
2 obrázky	98,850 %	0,99	0,99	0,99	3
16 obrázků	98,870 %	0,99	0,99	0,99	4
64 obrázků	98,700 %	0,99	0,99	0,99	<b>2</b>
256 obrázků	98,920 %	0,99	0,99	0,99	4
1024 obrázků	<b>98,930 %</b>	0,99	0,99	0,99	3





Obr. A.7: Průběhy chyby a správnosti na validační a trénovací sadě databáze MNIST při učení jednoduché konvoluční neuronové sítě pro 5 různých konfigurací metody inicializace vah LSUV. Průběhy jsou zobrazeny do epochy optimálního naučení.

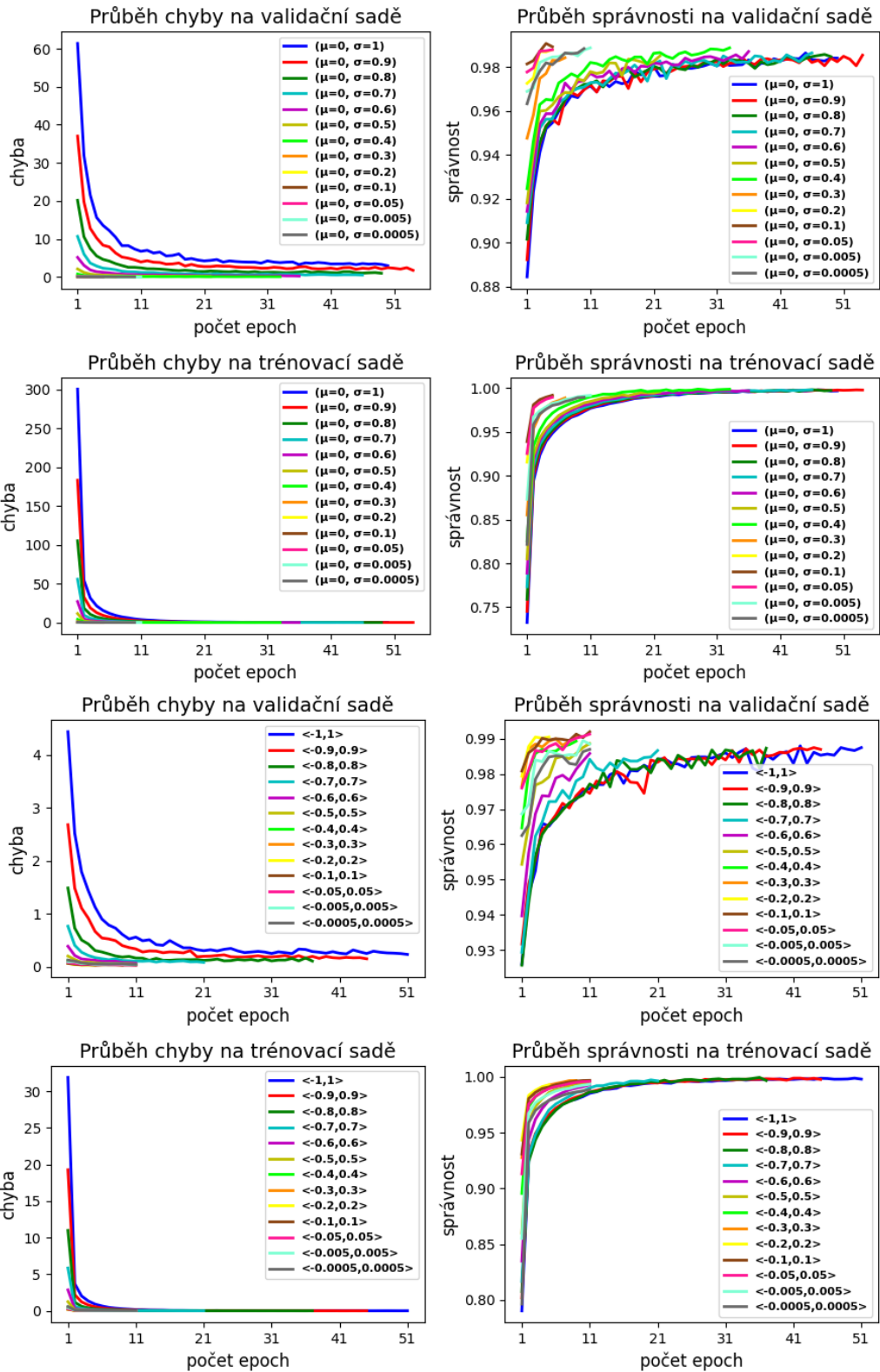
Tab. A.18: Popis počtu epoch k dosažení 85 %, 90 %, 95 %, 96 %, 97 %, 98 % a 99 % správnosti na validační sadě při učení jednoduché konvoluční neuronové sítě, které byla inicializovaná metodou LSUV. Hodnoty počtu epoch jsou vyobrazeny pro 5 různých konfigurací metody. Znak "-" značí nedosažení dané správnosti. Nejlepší výsledek je podbarven modrou.

typ konfigurace	Počet epoch k dosažení určité správnosti na validační sadě						
	85 %	90 %	95 %	96 %	97 %	98 %	99 %
2 obrázky	1	1	1	1	1	1	3
16 obrázků	1	1	1	1	1	1	6
64 obrázků	1	1	1	1	1	1	7
256 obrázků	1	1	1	1	1	1	4
1024 obrázků	1	1	1	1	1	1	4

### A.3.1 Náhodná inicializace vah z normálního a rovnoměrného pravděpodobnostního rozdělení

Tab. A.19: Výsledky ohodnocující učení jednoduché konvoluční neuronové sítě. Zvolená inicializace vah byla inicializace vah z normálního pravděpodobnostního rozložení v různých konfiguracích. Síť byla učena na trénovací sadě databáze MNIST. Hodnoty jsou získány pro testovací sadu databáze MNIST. Šestý sloupec vyjadřuje počet epoch k dosažení optimálního naučení sítě. Žluté podbarvení vyjadřuje nejvyšší dosaženou správnost a oranžové podbarvení nejmenší počet epoch.

Metoda inicializace z normálního pravděpodobnostního rozložení					
typ konfigurace	správnost	průměrná vážená přesnost	průměrná vážená výtěžnost	průměrné vážené F1-skóre	počet epoch
$(\mu = 0, \sigma = 1)$	98,090 %	0,98	0,98	0,98	50
$(\mu = 0, \sigma = 0,9)$	98,410 %	0,98	0,98	0,98	54
$(\mu = 0, \sigma = 0,8)$	98,370 %	0,98	0,98	0,98	49
$(\mu = 0, \sigma = 0,7)$	98,490 %	0,98	0,98	0,98	46
$(\mu = 0, \sigma = 0,6)$	98,470 %	0,98	0,98	0,98	36
$(\mu = 0, \sigma = 0,5)$	98,140 %	0,98	0,98	0,98	22
$(\mu = 0, \sigma = 0,4)$	98,790 %	0,99	0,99	0,99	33
$(\mu = 0, \sigma = 0,3)$	98,130 %	0,98	0,98	0,98	7
$(\mu = 0, \sigma = 0,2)$	98,900 %	0,99	0,99	0,99	5
$(\mu = 0, \sigma = 0,1)$	<b>99,080 %</b>	0,99	0,99	0,99	5
$(\mu = 0, \sigma = 0,05)$	98,780 %	0,99	0,99	0,99	5
$(\mu=0, \sigma=0,005)$	98,690 %	0,99	0,99	0,99	11
$(\mu = 0, \sigma = 0,0005)$	98,790 %	0,99	0,99	0,99	10



Obr. A.8: Průběhy chyby a správnosti na validační a trénovací sadě databáze MNIST při učení jednoduché konvoluční neuronové sítě pro konfigurace metod inicializace vah z normálního a rovnoměrného pravděpodobnostního rozložení. Horní čtyři grafy vyjadřují průběhy pro inicializace vah z normálního pravděpodobnostního rozložení a čtyři spodní grafy vyjadřují průběhy pro inicializace vah z rovnoměrného pravděpodobnostního rozložení. Průběhy jsou zobrazeny do epochy optimálního naučení.

Tab. A.20: Popis počtu epoch k dosažení 85 %, 90 %, 95 %, 96 %, 97 %, 98 % a 99 % správnosti na validační sadě při učení jednoduché konvoluční neuronové sítě, inicializované metodou náhodného inicializace vah z normálního pravděpodobnostního rozložení v různých konfiguracích. Hodnoty počtu epoch jsou vyobrazeny pro 13 různých konfigurací metody. Znak "-" značí nedosažení dané správnosti. Nejlepší výsledek je podbarven modrou.

typ konfigurace	Počet epoch k dosažení určité správnosti na validační sadě						
	85 %	90 %	95 %	96 %	97 %	98 %	99 %
$(\mu = 0, \sigma = 1)$	1	2	4	7	10	21	-
$(\mu = 0, \sigma = 0,9)$	1	2	4	7	11	25	-
$(\mu = 0, \sigma = 0,8)$	1	1	4	6	9	21	-
$(\mu = 0, \sigma = 0,7)$	1	1	3	6	9	18	-
$(\mu = 0, \sigma = 0,6)$	1	1	3	6	7	17	-
$(\mu = 0, \sigma = 0,5)$	1	1	3	4	9	14	-
$(\mu = 0, \sigma = 0,4)$	1	1	3	3	6	11	-
$(\mu = 0, \sigma = 0,3)$	1	1	2	3	3	5	-
$(\mu = 0, \sigma = 0,2)$	1	1	1	1	1	3	-
$(\mu = 0, \sigma = 0,1)$	1	1	1	1	1	1	4
$(\mu = 0, \sigma = 0,05)$	1	1	1	1	1	3	10
$(\mu = 0, \sigma = 0,005)$	-	1	1	1	2	3	-
$(\mu = 0, \sigma = 0,0005)$	1	1	1	1	2	4	-

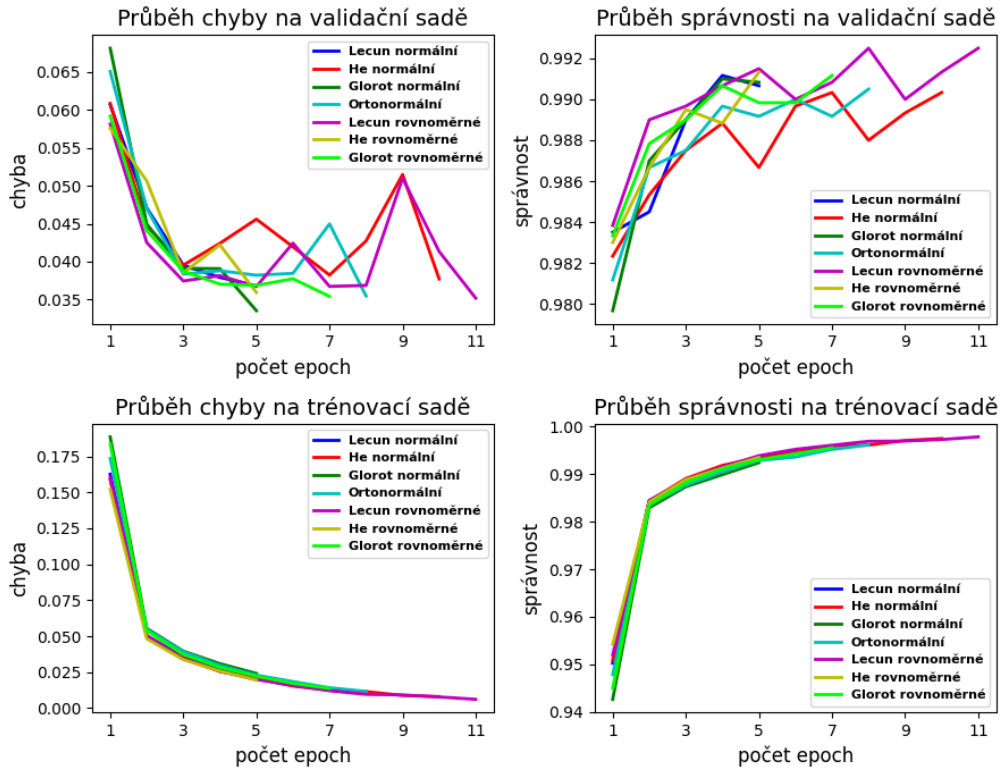
Tab. A.21: Výsledky ohodnocující učení jednoduché konvoluční neuronové sítě. Zvolená inicializace vah byla inicializace vah z rovnoměrného pravděpodobnostního rozložení v různých konfiguracích. Síť byla učena na trénovací sadě databáze MNIST. Hodnoty jsou získány pro testovací sadu databáze MNIST. Šestý sloupec vyjadřuje počet epoch k dosažení optimálního naučení sítě. Žluté podbarvení vyjadřuje nejvyšší dosaženou správnost a oranžové podbarvení nejnižší počet epoch.

Metoda inicializace z rovnoměrného pravděpodobnostního rozložení					
typ konfigurace	správnost	průměrná vážená přesnost	průměrná vážená výtěžnost	průměrné vážené F1-skóre	počet epoch
<-1,1>	98,600 %	0,99	0,99	0,99	51
<-0,9;0,9>	98,490 %	0,98	0,98	0,98	45
<-0,8;0,8>	98,620 %	0,99	0,99	0,99	37
<-0,7;0,7>	98,520 %	0,99	0,99	0,99	21
<-0,6;0,6>	98,600 %	0,99	0,99	0,99	11
<-0,5;0,5>	98,580 %	0,99	0,99	0,99	11
<-0,4;0,4>	98,800 %	0,99	0,99	0,99	9
<-0,3;0,3>	99,040 %	0,99	0,99	0,99	9
<-0,2;0,2>	99,090 %	0,99	0,99	0,99	5
<-0,1;0,1>	99,100 %	0,99	0,99	0,99	11
<-0,05;0,05>	99,110 %	0,99	0,99	0,99	11
<-0,005;0,005>	98,860 %	0,99	0,99	0,99	11
<-0,0005;0,0005>	98,480 %	0,98	0,98	0,98	11

Tab. A.22: Popis počtu epoch k dosažení určité správnosti na validační sadě při učení jednoduché konvoluční neuronové sítě, inicializované metodou náhodného inicializace vah z rovnoměrného pravděpodobnostního rozložení v různých konfiguracích. Hodnoty počtu epoch jsou vyobrazeny pro 13 různých konfigurací metody. Znak "-" značí nedosažení dané správnosti. Nejlepší výsledek je podbarven modrou.

typ konfigurace	Počet epoch k dosažení určité správnosti na validační sadě						
	85 %	90 %	95 %	96 %	97 %	98 %	99 %
<-1,1>	1	1	3	4	7	15	-
<-0,9;0,9>	1	1	3	4	7	15	-
<-0,8;0,8>	1	1	3	4	8	13	-
<-0,7;0,7>	1	1	3	3	5	11	-
<-0,6;0,6>	1	1	2	3	4	9	-
<-0,5;0,5>	1	1	1	2	3	6	-
<-0,4;0,4>	1	1	1	1	2	2	15
<-0,3;0,3>	1	1	1	1	1	2	6
<-0,2;0,2>	1	1	1	1	1	2	3
<-0,1;0,1>	1	1	1	1	1	1	4
<-0,05;0,05>	1	1	1	1	1	2	9
<-0,005;0,005>	1	1	1	1	2	3	21
<-0,0005;0,0005>	1	1	1	1	3	4	-

### A.3.2 Ostatní metody náhodné inicializace



Obr. A.9: Průběhy chyby a správnosti na validační a trénovací sadě databáze MNIST při učení jednoduché konvoluční neuronové sítě, která byla inicializována vybranými metodami náhodné inicializace. Zvolené inicializace vah byly LeCun, He, Glorot, které se dále dělí podle toho, z jakého pravděpodobnostního rozložení jsou váhy vzorkovány na rovnoměrné a normální. Dále byla použita metoda inicializaci vah pomocí rozkladu na ortonormální bázi. Průběhy jsou zobrazeny do epochy optimálního naučení.

Tab. A.23: Výsledky ohodnocující učení jednoduché konvoluční neuronové sítě. Zvolené inicializace vah byly LeCun, He, Glorot, které se dále dělí podle toho, z jakého pravděpodobnostního rozložení jsou váhy vzorkovány na rovnoměrné a normální. Dále byla použita metoda inicializaci vah pomocí rozkladu na ortonormální bázi. Síť byla učena na trénovací sadě databáze MNIST. Hodnoty jsou získány pro testovací sadu databáze MNIST. Šestý sloupec vyjadřuje počet epoch k dosažení optimálního naučení sítě. Žluté podbarvení vyjadřuje nejvyšší dosaženou správnost a oranžové podbarvení nejnižší počet epoch.

Ostatní metody náhodné inicializace					
typ metody	správnost	průměrná vážená přesnost	průměrná vážená výtežnost	průměrné vážené F1-skóre	počet epoch
LeCun rovnoměrné	99,200 %	0,99	0,99	0,99	11
LeCun normální	99,150 %	0,99	0,99	0,99	<b>5</b>
He rovnoměrné	99,110 %	0,99	0,99	0,99	<b>5</b>
He normální	99,070 %	0,99	0,99	0,99	10
Glorot rovnoměrné	99,090 %	0,99	0,99	0,99	7
Glorot normální	<b>99,190 %</b>	0,99	0,99	0,99	<b>5</b>
Ortonormální	<b>99,190 %</b>	0,99	0,99	0,99	8

Tab. A.24: Popis počtu epoch k dosažení určité správnosti na validační sadě při učení jednoduché konvoluční neuronové sítě, která byla inicializována vybranými metodami náhodné inicializace. Zvolené inicializace vah byly LeCun, He, Glorot, které se dále dělí podle toho, z jakého pravděpodobnostního rozložení jsou váhy vzorkovány na rovnoměrné a normální. Dále byla použita metoda inicializaci vah pomocí rozkladu na ortonormální bázi. Hodnoty počtu epoch jsou vyobrazeny pro 4 metody a jejich případné konfigurace. Znak "-" značí nedosažení dané správnosti. Nejlepší výsledek je podbarven modrou.

typ konfigurace	Počet epoch k dosažení určité správnosti na validační sadě						
	85 %	90 %	95 %	96 %	97 %	98 %	99 %
LeCun rovnoměrné	1	1	1	1	1	1	4
LeCun normální	1	1	1	1	1	1	4
He rovnoměrné	1	1	1	1	1	1	5
He normální	1	1	1	1	1	1	7
Glorot rovnoměrné	1	1	1	1	1	1	4
Glorot normální	1	1	1	1	1	2	4
Ortonormální	1	1	1	1	1	1	6

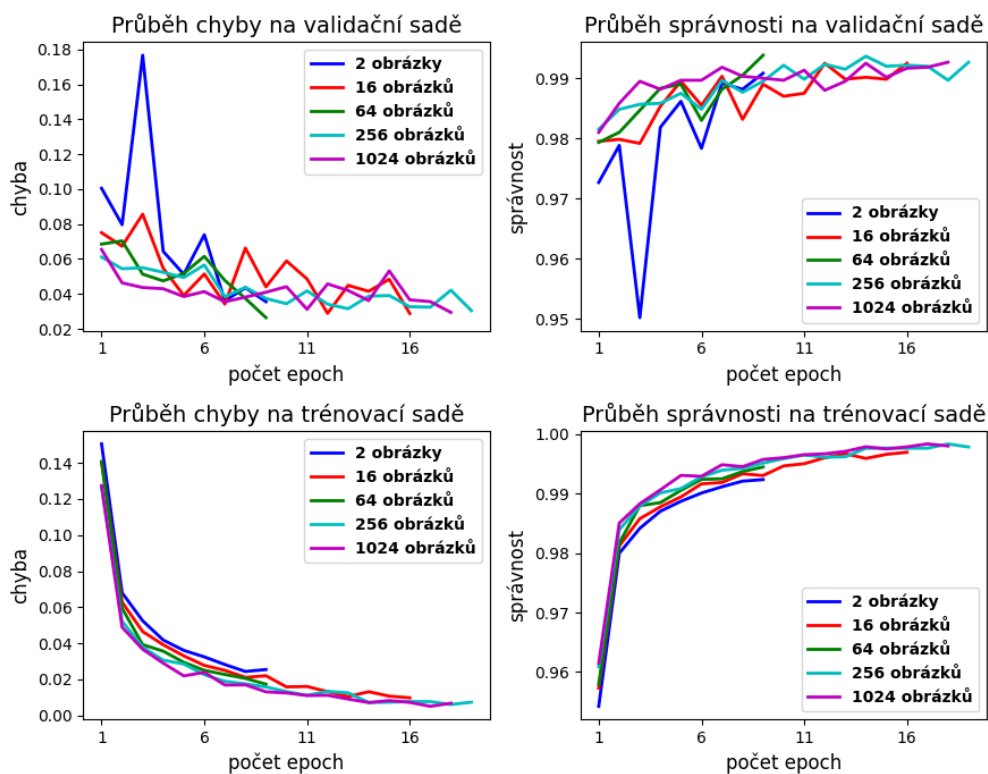
## A.4 Hluboká konvoluční neuronová síť – ResNet18

### A.4.1 LSUV

Tab. A.25: Výsledky ohodnocující učení hluboké konvoluční neuronové sítě – ResNet18, která byla inicializovaná metodou LSUV v různých konfiguracích a poté učena na trénovací sadě databáze MNIST. Hodnoty jsou získány pro testovací sadu databáze MNIST. Šestý sloupec vyjadřuje počet epoch k dosažení optimálního naučení sítě. Žluté podbarvení vyjadřuje nejvyšší dosaženou správnost a oranžové podbarvení nejnižší počet epoch.

Metoda LSUV					
typ konfigurace	správnost	průměrná vážená přesnost	průměrná vážená výtěžnost	průměrné vážené F1-skóre	počet epoch
2 obrázky	99,050 %	0,99	0,99	0,99	<b>9</b>
16 obrázků	<b>99,220 %</b>	0,99	0,99	0,99	16
64 obrázků	99,070 %	0,99	0,99	0,99	<b>9</b>
256 obrázků	99,210 %	0,99	0,99	0,99	19
1024 obrázků	99,080 %	0,99	0,99	0,99	18





Obr. A.10: Průběhy chyby a správnosti na validační a trénovací sadě databáze MNIST při učení hluboké konvoluční neuronové sítě – ResNet18, pro 5 různých konfigurací metody inicializace vah LSUV. Průběhy jsou zobrazeny do epochy optimálního naučení.

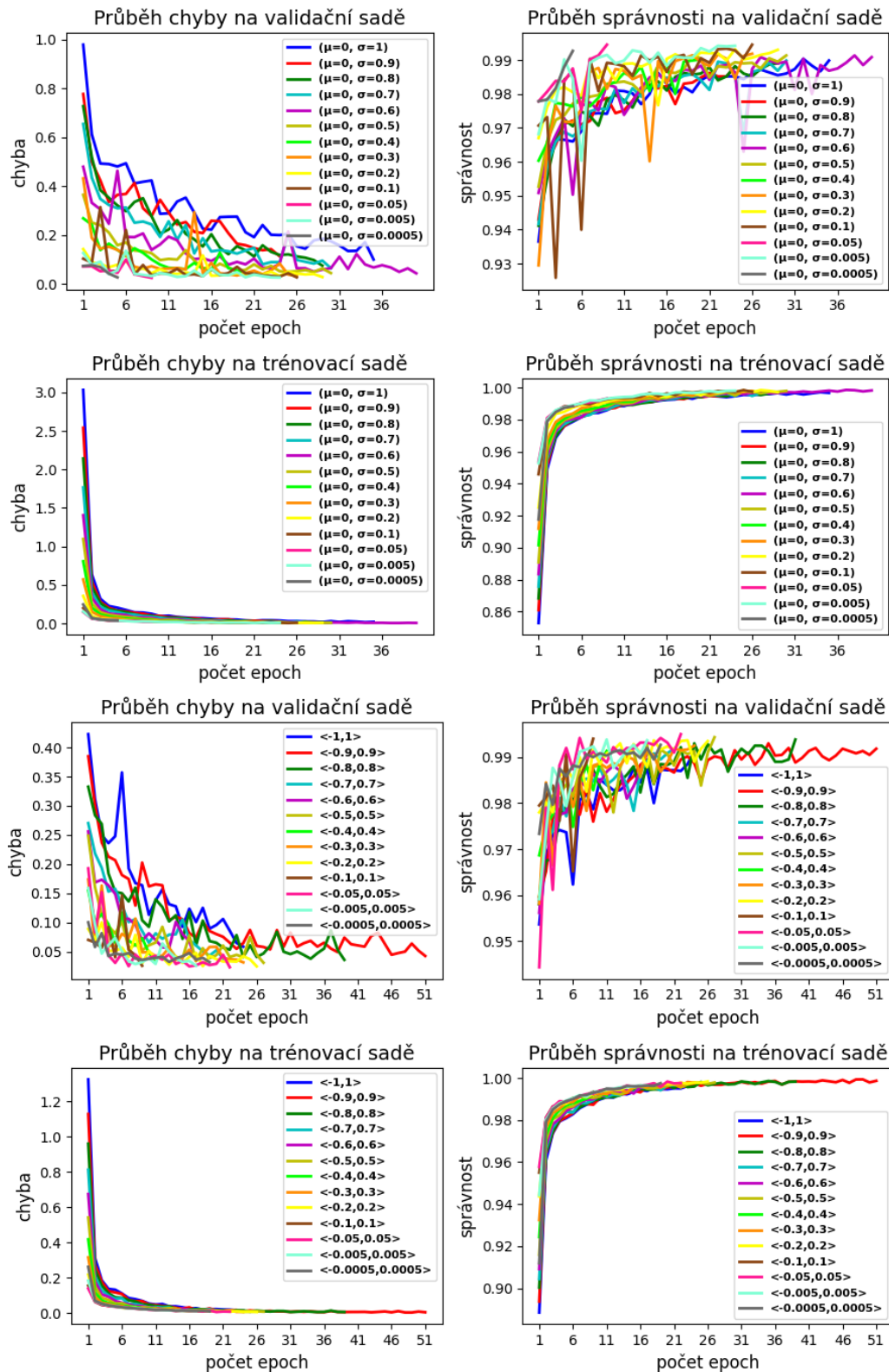
Tab. A.26: Popis počtu epoch k dosažení 85 %, 90 %, 95 %, 96 %, 97 %, 98 % a 99 % správnosti na validační sadě při učení hluboké konvoluční neuronové sítě – ResNet18, které byla inicializovaná metodou LSUV. Hodnoty počtu epoch jsou vyobrazeny pro 5 různých konfigurací metody. Znak "-" značí nedosažení dané správnosti. Nejlepší výsledek je podbarven modrou.

typ konfigurace	Počet epoch k dosažení určité správnosti na validační sadě						
	85 %	90 %	95 %	96 %	97 %	98 %	99 %
2 obrázky	1	1	1	1	1	4	9
16 obrázků	1	1	1	1	1	4	7
64 obrázků	1	1	1	1	1	2	8
256 obrázků	1	1	1	1	1	1	10
1024 obrázků	1	1	1	1	1	1	7

## A.4.2 Náhodná inicializace vah z normálního a rovnoměrného pravděpodobnostního rozdělení

Tab. A.27: Výsledky ohodnocující učení hluboké konvoluční neuronové sítě – ResNet18. Zvolená inicializace vah byla inicializace vah z normálního pravděpodobnostního rozložení v různých konfiguracích. Síť byla učena na trénovací sadě databáze MNIST. Hodnoty jsou získány pro testovací sadu databáze MNIST. Šestý sloupec vyjadřuje počet epoch k dosažení optimálního naučení sítě. Žluté podbarvení vyjadřuje nejvyšší dosaženou správnost a oranžové podbarvení nejnižší počet epoch.

Metoda inicializace z normálního pravděpodobnostního rozložení					
typ konfigurace	správnost	průměrná vážená přesnost	průměrná vážená výtěžnost	průměrné vážené F1-skóre	počet epoch
$(\mu = 0, \sigma = 1)$	98,500 %	0,98	0,98	0,98	35
$(\mu = 0, \sigma = 0,9)$	98,710 %	0,99	0,99	0,99	24
$(\mu = 0, \sigma = 0,8)$	98,660 %	0,99	0,99	0,99	29
$(\mu = 0, \sigma = 0,7)$	98,900 %	0,99	0,99	0,99	30
$(\mu = 0, \sigma = 0,6)$	99,240 %	0,99	0,99	0,99	40
$(\mu = 0, \sigma = 0,5)$	99,170 %	0,99	0,99	0,99	30
$(\mu = 0, \sigma = 0,4)$	98,730 %	0,99	0,99	0,99	20
$(\mu = 0, \sigma = 0,3)$	99,230 %	0,99	0,99	0,99	26
$(\mu = 0, \sigma = 0,2)$	99,250 %	0,99	0,99	0,99	29
$(\mu = 0, \sigma = 0,1)$	<b>99,350 %</b>	0,99	0,99	0,99	26
$(\mu = 0, \sigma = 0,05)$	99,240 %	0,99	0,99	0,99	9
$(\mu=0, \sigma=0,005)$	99,330 %	0,99	0,99	0,99	24
$(\mu = 0, \sigma = 0,0005)$	98,990 %	0,99	0,99	0,99	<b>5</b>



Obr. A.11: Průběhy chyby a správnosti na validační a trénovací sadě databáze MNIST při učení hluboké konvoluční neuronové sítě – ResNet18 pro konfigurace metod inicializace vah z normálního a rovnoměrného pravděpodobnostního rozložení. Horní čtyři grafy vyjadřují průběhy pro inicializace vah z normálního pravděpodobnostního rozložení a čtyři spodní grafy vyjadřují průběhy pro inicializace vah z rovnoměrného pravděpodobnostního rozložení. Průběhy jsou zobrazeny do epochy optimálního naučení.

Tab. A.28: Popis počtu epoch k dosažení 85 %, 90 %, 95 %, 96 %, 97 %, 98 % a 99 % správnosti na validační sadě při učení hluboké konvoluční neuronové sítě – ResNet18 inicializované metodou náhodného inicializace vah z normálního pravděpodobnostního rozložení v různých konfiguracích. Hodnoty počtu epoch jsou vyobrazeny pro 13 různých konfigurací metody. Znak "-" značí nedosažení dané správnosti. Nejlepší výsledek je podbarven modrou.

typ konfigurace	Počet epoch k dosažení určité správnosti na validační sadě						
	85 %	90 %	95 %	96 %	97 %	98 %	99 %
$(\mu = 0, \sigma = 1)$	1	1	2	3	7	10	32
$(\mu = 0, \sigma = 0,9)$	1	1	2	2	4	13	-
$(\mu = 0, \sigma = 0,8)$	1	1	2	3	4	13	-
$(\mu = 0, \sigma = 0,7)$	1	1	2	3	7	10	37
$(\mu = 0, \sigma = 0,6)$	1	1	1	2	4	9	27
$(\mu = 0, \sigma = 0,5)$	1	1	1	2	3	9	19
$(\mu = 0, \sigma = 0,4)$	1	1	1	1	3	9	20
$(\mu = 0, \sigma = 0,3)$	1	1	1	2	3	7	18
$(\mu = 0, \sigma = 0,2)$	1	1	1	1	2	3	16
$(\mu = 0, \sigma = 0,1)$	1	1	1	1	1	4	7
$(\mu = 0, \sigma = 0,05)$	1	1	1	1	1	2	7
$(\mu = 0, \sigma = 0,005)$	1	1	1	1	2	4	4
$(\mu = 0, \sigma = 0,0005)$	1	1	1	1	1	3	5

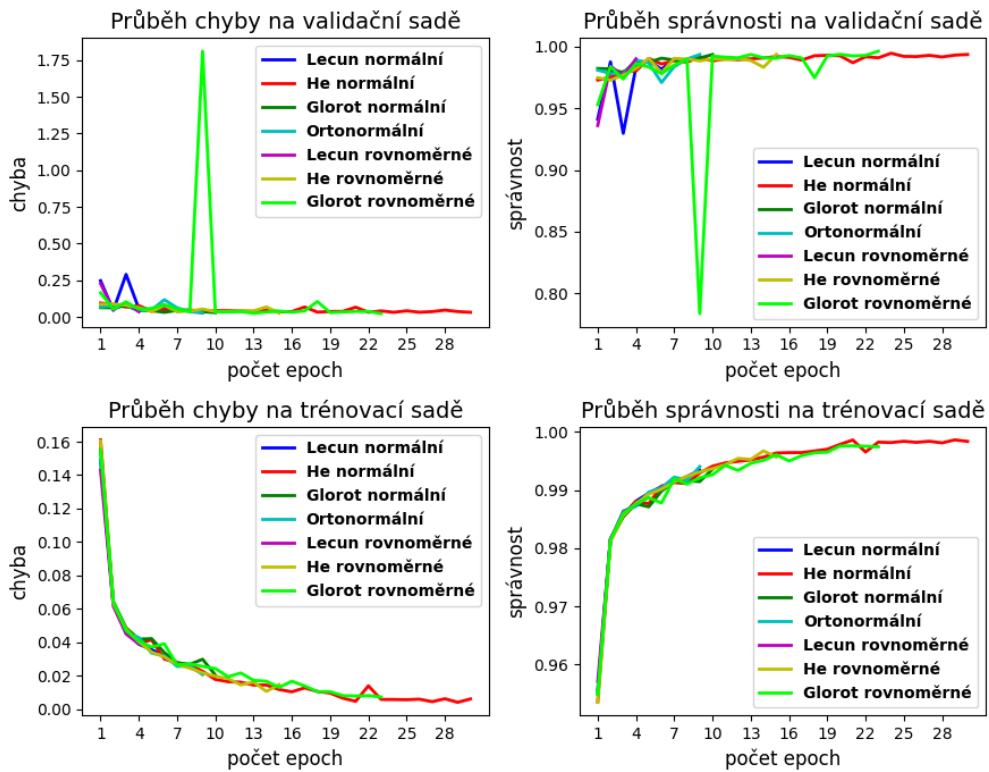
Tab. A.29: Výsledky ohodnocující učení hluboké konvoluční neuronové sítě – ResNet18. Zvolená inicializace vah byla inicializace vah z rovnoměrného pravděpodobnostního rozložení v různých konfiguracích. Síť byla učena na trénovací sadě databáze MNIST. Hodnoty jsou získány pro testovací sadu databáze MNIST. Šestý sloupec vyjadřuje počet epoch k dosažení optimálního naučení sítě. Žluté podbarvení vyjadřuje nejvyšší dosaženou správnost a oranžové podbarvení nejnižší počet epoch.

Metoda inicializace z rovnoměrného pravděpodobnostního rozložení					
typ konfigurace	správnost	průměrná vážená přesnost	průměrná vážená výtěžnost	průměrné vážené F1-skóre	počet epoch
<-1,1>	98,950 %	0,99	0,99	0,99	24
<-0,9;0,9>	99,140 %	0,99	0,99	0,99	21
<-0,8;0,8>	99,170 %	0,99	0,99	0,99	39
<-0,7;0,7>	99,200 %	0,99	0,99	0,99	21
<-0,6;0,6>	98,960 %	0,99	0,99	0,99	17
<-0,5;0,5>	99,250 %	0,99	0,99	0,99	27
<-0,4;0,4>	98,950 %	0,99	0,99	0,99	11
<-0,3;0,3>	99,090 %	0,99	0,99	0,99	24
<-0,2;0,2>	<b>99,400 %</b>	0,99	0,99	0,99	26
<-0,1;0,1>	99,290 %	0,99	0,99	0,99	<b>9</b>
<-0,05;0,05>	99,320 %	0,99	0,99	0,99	22
<-0,005;0,005>	99,090 %	0,99	0,99	0,99	17
<-0,0005;0,0005>	99,310 %	0,99	0,99	0,99	19

Tab. A.30: Popis počtu epoch k dosažení určité správnosti na validační sadě při učení hluboké konvoluční neuronové sítě – ResNet18, inicializované metodou náhodného inicializace vah z rovnoměrného pravděpodobnostního rozložení v různých konfiguracích. Hodnoty počtu epoch jsou vyobrazeny pro 13 různých konfigurací metody. Znak "-" značí nedosažení dané správnosti. Nejlepší výsledek je podbarven modrou.

typ konfigurace	Počet epoch k dosažení určité správnosti na validační sadě						
	85 %	90 %	95 %	96 %	97 %	98 %	99 %
<-1,1>	1	1	1	2	3	7	24
<-0,9;0,9>	1	1	1	2	3	6	23
<-0,8;0,8>	1	1	1	2	4	6	22
<-0,7;0,7>	1	1	1	2	2	4	18
<-0,6;0,6>	1	1	1	2	2	6	17
<-0,5;0,5>	1	1	1	2	2	3	14
<-0,4;0,4>	1	1	1	1	2	4	11
<-0,3;0,3>	1	1	1	2	2	2	14
<-0,2;0,2>	1	1	1	1	1	2	7
<-0,1;0,1>	1	1	1	1	1	2	5
<-0,05;0,05>	1	1	1	2	2	4	5
<-0,005;0,005>	1	1	1	2	2	3	7
<-0,0005;0,0005>	1	1	1	1	1	2	8

### A.4.3 Ostatní metody náhodné inicializace



Obr. A.12: Průběhy chyby a správnosti na validační a trénovací sadě databáze MNIST při učení hluboké konvoluční neuronové sítě – ResNet18, která byla inicializována vybranými metodami náhodné inicializace. Zvolené inicializace vah byly LeCun, He, Glorot, které se dále dělí podle toho, z jakého pravděpodobnostního rozložení jsou váhy vzorkovány na rovnoměrné a normální. Dále byla použita s metodou inicializace vah pomocí rozkladu na ortonormální bázi. Průběhy jsou zobrazeny do epochy optimálního naučení.

Tab. A.31: Výsledky ohodnocující učení hluboké konvoluční neuronové sítě – ResNet18. Zvolené inicializace vah byly LeCun, He, Glorot, které se dále dělí podle toho, z jakého pravděpodobnostního rozložení jsou váhy vzorkovány na rovnoměrné a normální. Dále byla použita metoda inicializaci vah pomocí rozkladu na ortonormální bázi. Síť byla učena na trénovací sadě databáze MNIST. Hodnoty jsou získány pro testovací sadu databáze MNIST. Šestý sloupec vyjadřuje počet epoch k dosažení optimálního naučení sítě. Žluté podbarvení vyjadřuje nejvyšší dosaženou správnost a oranžové podbarvení nejnižší počet epoch.

Ostatní metody náhodné inicializace					
typ metody	správnost	průměrná vážená přesnost	průměrná vážená výtežnost	průměrné vážené F1-skóre	počet epoch
LeCun rovnoměrné	99,030 %	0,99	0,99	0,99	4
LeCun normální	99,150 %	0,99	0,99	0,99	9
He rovnoměrné	99,290 %	0,99	0,99	0,99	15
He normální	99,140 %	0,99	0,99	0,99	30
Glorot rovnoměrné	99,370 %	0,99	0,99	0,99	23
Glorot normální	99,350 %	0,99	0,99	0,99	10
Ortonormální	<b>99,400 %</b>	0,99	0,99	0,99	9

Tab. A.32: Popis počtu epoch k dosažení určité správnosti na validační sadě při učení hluboké konvoluční neuronové sítě – ResNet18, které byla inicializována vybranými metodami náhodné inicializace. Zvolené inicializace vah byly LeCun, He, Glorot, které se dále dělí podle toho, z jakého pravděpodobnostního rozložení jsou váhy vzorkovány na rovnoměrné a normální. Dále byla použita metoda inicializaci vah pomocí rozkladu na ortonormální bázi. Hodnoty počtu epoch jsou vyobrazeny pro 4 metody a jejich případné konfigurace. Znak "-" značí nedosažení dané správnosti. Nejlepší výsledek je podbarven modrou.

typ konfigurace	Počet epoch k dosažení určité správnosti na validační sadě						
	85 %	90 %	95 %	96 %	97 %	98 %	99 %
LeCun rovnoměrné	1	1	2	2	2	2	4
LeCun normální	1	1	2	2	2	2	7
He rovnoměrné	1	1	1	1	1	4	5
He normální	1	1	1	2	1	4	5
Glorot rovnoměrné	1	1	1	2	2	2	10
Glorot normální	1	1	1	1	1	1	6
Ortonormální	1	1	1	1	1	1	9

## B Výsledky pro databázi CIFAR-10

### B.1 Mělká vícevrstvá perceptronová síť

#### B.1.1 LSUV

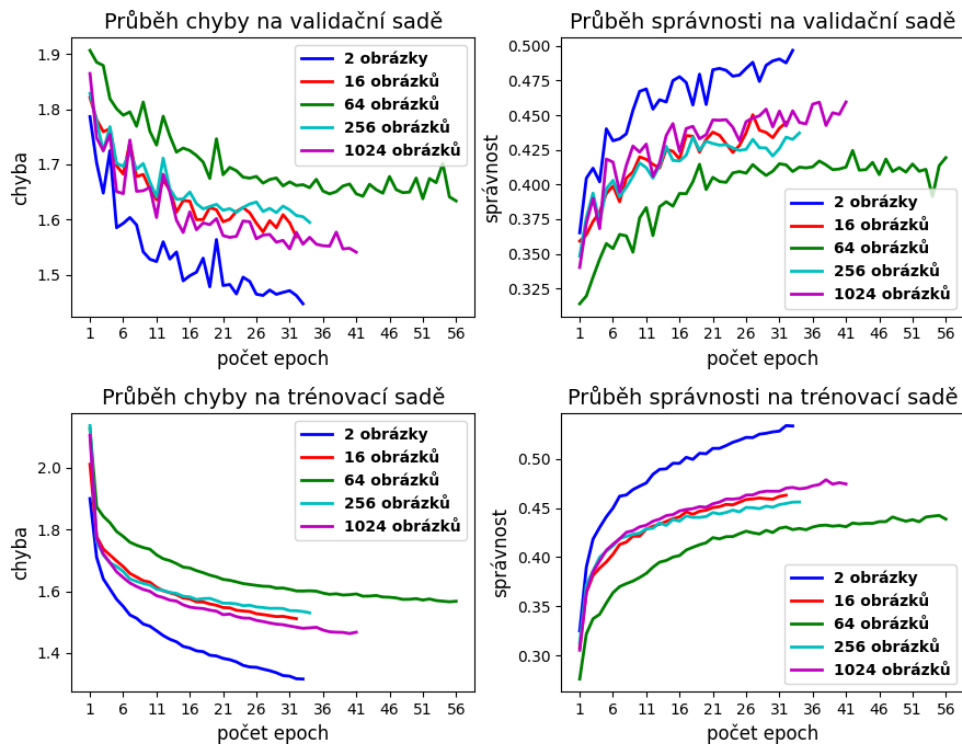
Tab. B.1: Výsledky ohodnocující učení mělké vícevrstvé perceptronové sítě, která byla inicializována metodou LSUV v různých konfiguracích a poté učena na trénovací sadě databáze CIFAR-10. Hodnoty jsou získány pro testovací sadu databáze CIFAR-10. Šestý sloupec vyjadřuje počet epoch k dosažení optimálního naučení sítě. Žluté podbarvení vyjadřuje nejvyšší dosaženou správnost a oranžové podbarvení nejnižší počet epoch.

Metoda LSUV					
typ konfigurace	správnost	průměrná vážená přesnost	průměrná vážená výtěžnost	průměrné vážené F1-skóre	počet epoch
2 obrázky	<b>48,980 %</b>	0,49	0,49	0,49	33
16 obrázků	44,150 %	0,44	0,44	0,44	<b>32</b>
64 obrázků	44,860 %	0,45	0,45	0,45	53
256 obrázků	43,920 %	0,44	0,44	0,43	39
1024 obrázků	43,890 %	0,44	0,44	0,43	39

Tab. B.2: Popis počtu epoch k dosažení 20 %, 25 %, 30 %, 35 %, 40 %, 45 % a 50 % správnosti na validační sadě při učení mělké vícevrstvé perceptronové sítě, která byla inicializována metodou LSUV. Hodnoty počtu epoch jsou vyobrazeny pro 5 různých konfigurací metody. Znak "-" značí nedosažení dané správnosti. Nejlepší výsledek je podbarven modrou.

typ konfigurace	Počet epoch k dosažení určité správnosti na validační sadě						
	20 %	25 %	30 %	35 %	40 %	45 %	50 %
2 obrázky	1	1	1	1	2	9	33
16 obrázků	1	1	1	1	8	27	-
64 obrázků	1	1	1	2	6	44	-
256 obrázků	1	1	1	1	8	47	-
1024 obrázků	1	1	1	2	5	-	-





Obr. B.1: Průběhy chyby a správnosti na validační a trénovací sadě databáze CIFAR-10 při učení mělké vícevrstvé perceptronové sítě pro 5 různých konfigurací metody inicializace vah LSUV. Průběhy jsou zobrazeny do epochy optimálního naučení.

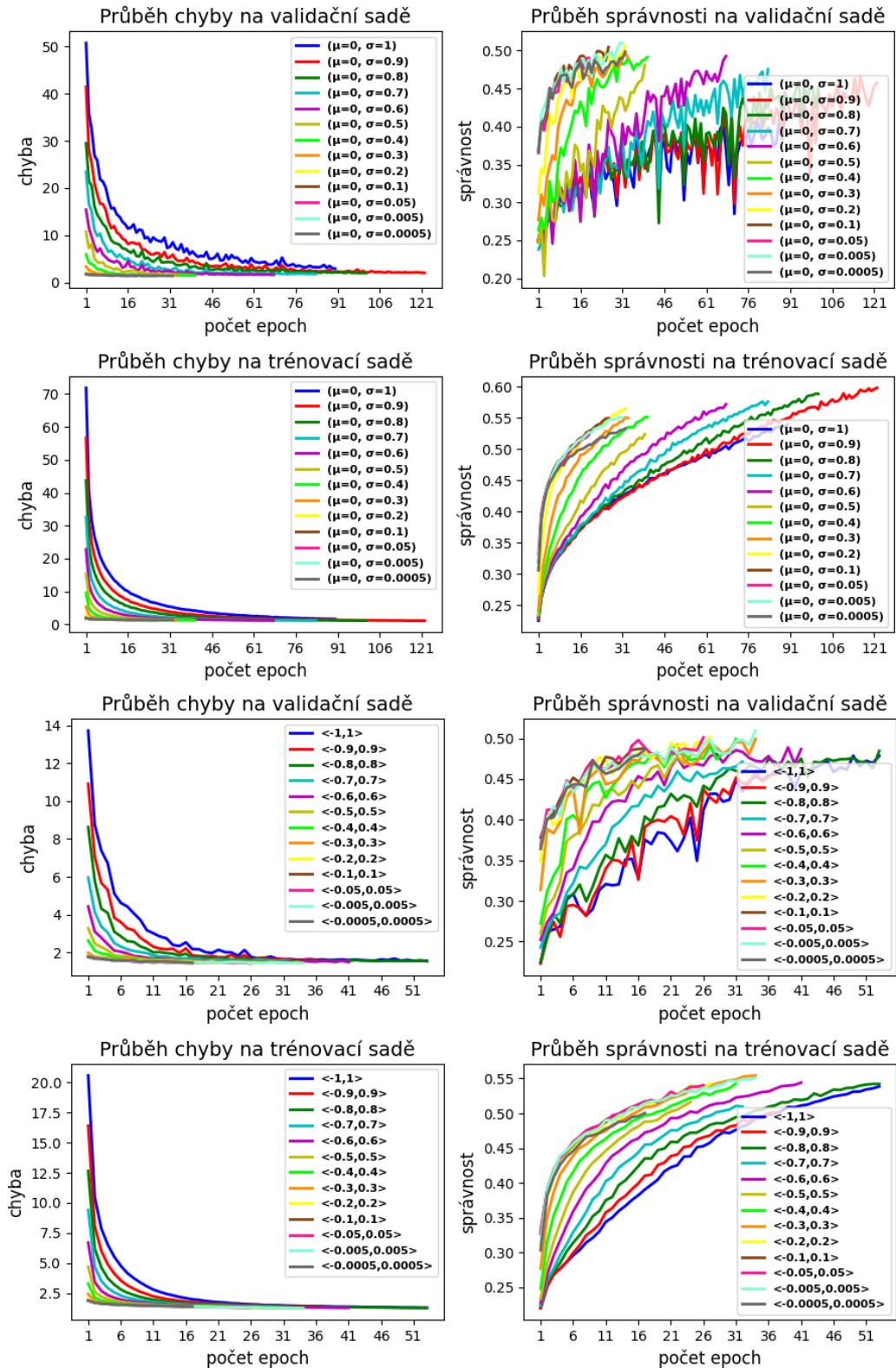
## B.1.2 Náhodná inicializace vah z normálního a rovnoměrného pravděpodobnostního rozdělení

Tab. B.3: Výsledky ohodnocující učení mělké vícevrstvé perceptronové sítě. Zvolená inicializace vah byla inicializace vah z normálního pravděpodobnostního rozložení v různých konfiguracích. Síť byla učena na trénovací sadě databáze CIFAR-10. Hodnoty jsou získány pro testovací sadu databáze CIFAR-10. Šestý sloupec vyjadřuje počet epoch k dosažení optimálního naučení sítě. Žluté podbarvení vyjadřuje nejvyšší dosaženou správnost a oranžové podbarvení nejnižší počet epoch.

Metoda inicializace z normálního pravděpodobnostního rozložení					
typ konfigurace	správnost	průměrná vážená přesnost	průměrná vážená výtěžnost	průměrné vážené F1-skóre	počet epoch
$(\mu = 0, \sigma = 1)$	42,240 %	0,43	0,42	0,41	90
$(\mu = 0, \sigma = 0,9)$	45,880 %	0,46	0,46	0,45	122
$(\mu = 0, \sigma = 0,8)$	45,580 %	0,46	0,46	0,44	101
$(\mu = 0, \sigma = 0,7)$	46,350 %	0,47	0,46	0,46	83
$(\mu = 0, \sigma = 0,6)$	47,040 %	0,48	0,47	0,47	68
$(\mu = 0, \sigma = 0,5)$	46,900 %	0,47	0,47	0,47	39
$(\mu = 0, \sigma = 0,4)$	47,960 %	0,49	0,48	0,47	40
$(\mu = 0, \sigma = 0,3)$	48,380 %	0,50	0,48	0,49	33
$(\mu = 0, \sigma = 0,2)$	50,060 %	0,50	0,50	0,50	32
$(\mu = 0, \sigma = 0,1)$	49,670 %	0,50	0,50	0,50	26
$(\mu = 0, \sigma = 0,05)$	48,480 %	0,50	0,50	0,50	19
$(\mu=0, \sigma=0,005)$	<b>50,120 %</b>	0,50	0,50	0,50	32
$(\mu = 0, \sigma = 0,0005)$	49,930 %	0,50	0,50	0,50	32

Tab. B.4: Popis počtu epoch k dosažení určité správnosti na validační sadě při učení mělké vícevrstvé perceptronové sítě, inicializované metodou náhodného inicializace vah z normálního pravděpodobnostního rozložení v různých konfiguracích. Hodnoty počtu epoch jsou vyobrazeny pro 13 různých konfigurací metody. Znak "-" značí nedosažení dané správnosti. Nejlepší výsledek je podbarven modrou.

typ konfigurace	Počet epoch k dosažení určité správnosti na validační sadě						
	20 %	25 %	30 %	35 %	40 %	45 %	50 %
$(\mu = 0, \sigma = 1)$	1	2	8	24	67	-	-
$(\mu = 0, \sigma = 0,9)$	1	1	8	24	67	108	-
$(\mu = 0, \sigma = 0,8)$	1	2	11	19	45	84	-
$(\mu = 0, \sigma = 0,7)$	1	4	8	25	36	66	-
$(\mu = 0, \sigma = 0,6)$	1	1	5	14	26	47	-
$(\mu = 0, \sigma = 0,5)$	1	2	5	14	22	36	-
$(\mu = 0, \sigma = 0,4)$	1	1	5	8	13	19	-
$(\mu = 0, \sigma = 0,3)$	1	1	2	6	9	14	-
$(\mu = 0, \sigma = 0,2)$	1	1	1	4	5	10	32
$(\mu = 0, \sigma = 0,1)$	1	1	1	1	2	7	26
$(\mu = 0, \sigma = 0,05)$	1	1	1	1	2	9	-
$(\mu = 0, \sigma = 0,005)$	1	1	1	1	2	5	29
$(\mu = 0, \sigma = 0,0005)$	1	1	1	1	2	5	-



Obr. B.2: Průběhy chyby a správnosti na validační a trénovací sadě databáze CIFAR-10 při učení mělké vícevrstvé perceptronovské sítě pro konfigurace metod inicializace vah z normálního a rovnoměrného pravděpodobnostního rozložení. Horní čtyři grafy vyjadřují průběhy pro inicializace vah z normálního pravděpodobnostního rozložení a čtyři spodní grafy vyjadřují průběhy pro inicializace vah z rovnoměrného pravděpodobnostního rozložení. Průběhy jsou zobrazeny do epochy optimálního naučení.

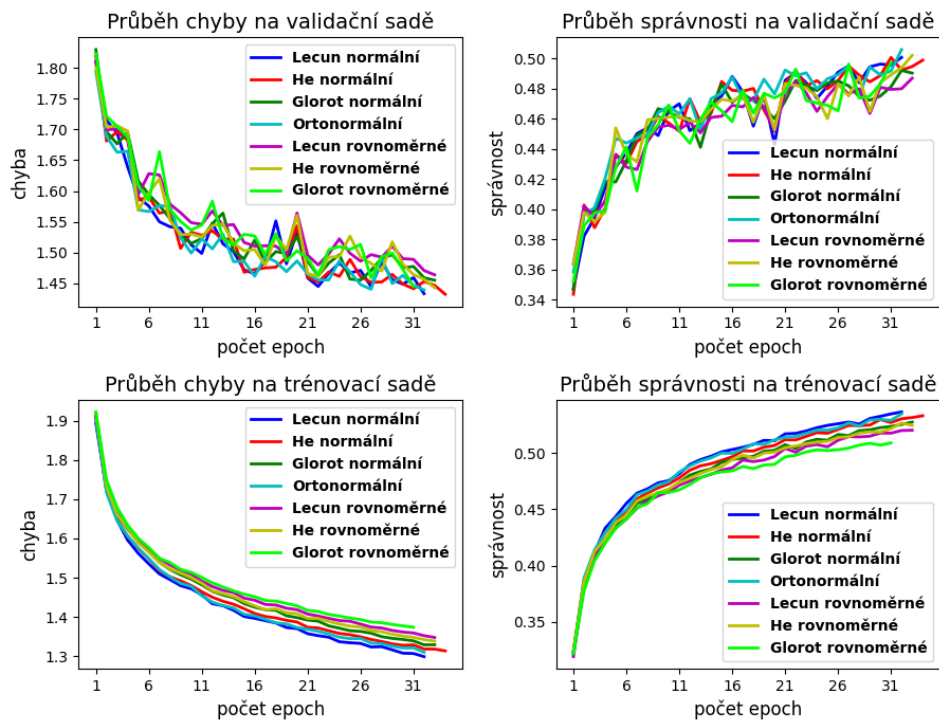
Tab. B.5: Výsledky ohodnocující učení mělké vícevrstvé perceptronové sítě. Zvolená inicializace vah byla inicializace vah z rovnoměrného pravděpodobnostního rozložení v různých konfiguracích. Síť byla učena na trénovací sadě databáze CIFAR-10. Hodnoty jsou získány pro testovací sadu databáze CIFAR-10. Šestý sloupec vyjadřuje počet epoch k dosažení optimálního naučení sítě. Žluté podbarvení vyjadřuje nejvyšší dosaženou správnost a oranžové podbarvení nejnižší počet epoch.

Metoda inicializace z rovnoměrného pravděpodobnostního rozložení					
typ konfigurace	správnost	průměrná vážená přesnost	průměrná vážená výtěžnost	průměrné vážené F1-skóre	počet epoch
<-1,1>	46,850 %	0,48	0,47	0,47	53
<-0,9;0,9>	45,390 %	0,45	0,45	0,45	37
<-0,8;0,8>	47,410 %	0,47	0,47	0,47	53
<-0,7;0,7>	46,460 %	0,47	0,47	0,46	32
<-0,6;0,6>	47,750 %	0,48	0,48	0,47	41
<-0,5;0,5>	47,390 %	0,48	0,47	0,47	24
<-0,4;0,4>	48,690 %	0,48	0,49	0,48	31
<-0,3;0,3>	49,140 %	0,50	0,49	0,49	34
<-0,2;0,2>	49,270 %	0,49	0,49	0,49	27
<-0,1;0,1>	49,370 %	0,48	0,49	0,48	17
<-0,05;0,05>	49,420 %	0,50	0,49	0,49	26
<-0,005;0,005>	49,330 %	0,50	0,49	0,49	34
<-0,0005;0,0005>	47,990 %	0,47	0,48	0,47	17

Tab. B.6: Popis počtu epoch k dosažení určité správnosti na validační sadě při učení mělké vícevrstvé perceptronové sítě, inicializované metodou náhodného inicializace vah z rovnoměrného pravděpodobnostního rozložení v různých konfiguracích. Hodnoty počtu epoch jsou vyobrazeny pro 13 různých konfigurací metody. Znak "-" značí nedosažení dané správnosti. Nejlepší výsledek je podbarven modrou.

typ konfigurace	Počet epoch k dosažení určité správnosti na validační sadě						
	20 %	25 %	30 %	35 %	40 %	45 %	50 %
<-1,1>	1	2	5	14	24	34	-
<-0,9;0,9>	1	2	9	12	21	31	-
<-0,8;0,8>	1	2	5	11	17	28	-
<-0,7;0,7>	1	2	5	10	14	22	-
<-0,6;0,6>	1	1	4	7	11	16	-
<-0,5;0,5>	1	1	3	5	9	17	-
<-0,4;0,4>	1	1	2	5	5	13	-
<-0,3;0,3>	1	1	1	2	5	9	-
<-0,2;0,2>	1	1	1	2	2	9	27
<-0,1;0,1>	1	1	1	1	2	6	27
<-0,05;0,05>	1	1	1	1	2	8	26
<-0,005;0,005>	1	1	1	1	2	8	34
<-0,0005;0,0005>	1	1	1	1	2	9	-

### B.1.3 Ostatní metody náhodné inicializace



Obr. B.3: Průběhy chyby a správnosti na validační a trénovací sadě databáze CIFAR-10 při učení mělké vícevrstvé perceptronové sítě, která byla inicializována vybranými metodami náhodné inicializace. Zvolené inicializace vah byly LeCun, He, Glorot, které se dále dělí podle toho, z jakého pravděpodobnostního rozložení jsou váhy vzorkovány na rovnoměrné a normální. Dále byla použita metoda inicializaci vah pomocí rozkladu na ortonormální bázi. Průběhy jsou zobrazeny do epochy optimálního naučení.

Tab. B.7: Výsledky ohodnocující učení mělké vícevrstvé perceptronovské sítě. Zvolené inicializace vah byly LeCun, He, Glorot, které se dále dělí podle toho, z jakého pravděpodobnostního rozložení jsou váhy vzorkovány na rovnoměrné a normální. Dále byla použita s metodou inicializaci vah pomocí rozkladu na ortonormální bázi. Síť byla učena na trénovací sadě databáze CIFAR-10. Hodnoty jsou získány pro testovací sadu databáze CIFAR-10. Šestý sloupec vyjadřuje počet epoch k dosažení optimálního naučení sítě. Žluté podbarvení vyjadřuje nejvyšší dosaženou správnost a oranžové podbarvení nejnižší počet epoch.

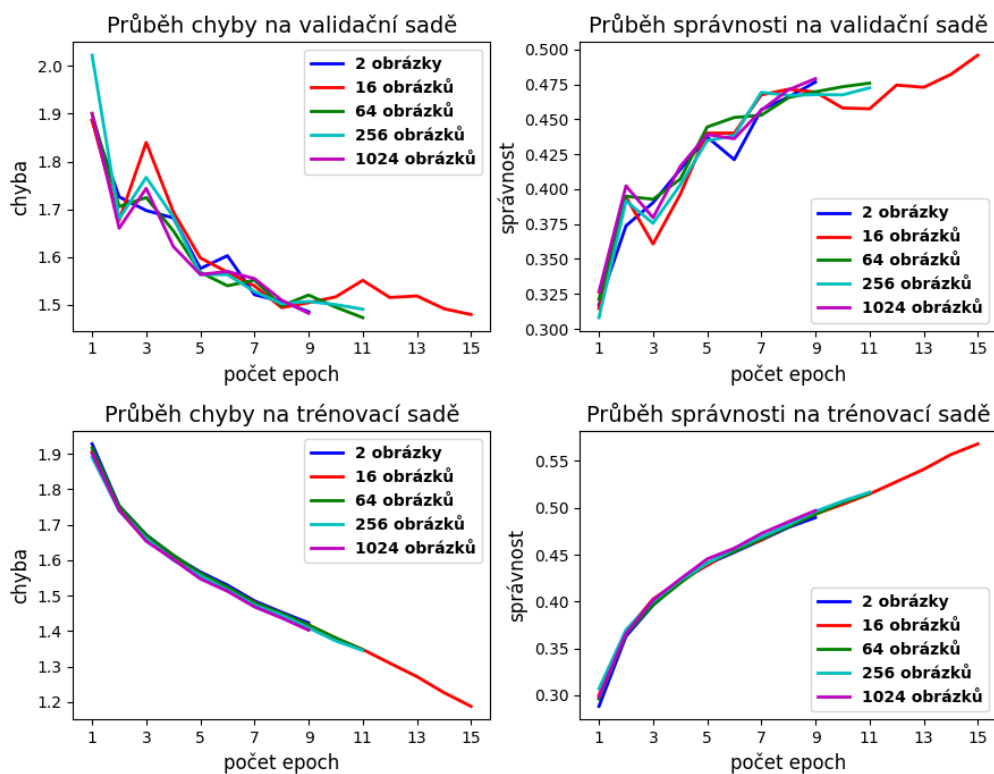
Ostatní metody náhodné inicializace					
typ metody	správnost	průměrná vážená přesnost	průměrná vážená výtěžnost	průměrné vážené F1-skóre	počet epoch
LeCun rovnoměrné	48,240 %	0,48	0,48	0,48	32
LeCun normální	49,070 %	0,50	0,49	0,49	32
He rovnoměrné	48,550 %	0,49	0,49	0,48	33
He normální	<b>49,230 %</b>	0,49	0,49	0,49	34
Glorot rovnoměrné	48,810 %	0,48	0,49	0,48	<b>31</b>
Glorot normální	48,210 %	0,49	0,48	0,48	33
Ortonormální	48,930 %	0,50	0,49	0,49	32

Tab. B.8: Popis počtu epoch k dosažení určité správnosti na validační sadě při učení mělké vícevrstvé perceptronovské sítě, která byla inicializována vybranými metodami náhodné inicializace. Zvolené inicializace vah byly LeCun, He, Glorot, které se dále dělí podle toho, z jakého pravděpodobnostního rozložení jsou váhy vzorkovány na rovnoměrné a normální. Dále byla použita metoda inicializaci vah pomocí rozkladu na ortonormální bázi. Hodnoty počtu epoch jsou vyobrazeny pro 4 metody a jejich případné konfigurace. Znak "-" značí nedosažení dané správnosti. Nejlepší výsledek je podbarven modrou.

typ konfigurace	Počet epoch k dosažení určité správnosti na validační sadě						
	20 %	25 %	30 %	35 %	40 %	45 %	50 %
LeCun rovnoměrné	1	1	1	1	2	9	-
LeCun normální	1	1	1	2	4	8	32
He rovnoměrné	1	1	1	1	5	5	33
He normální	1	1	1	2	2	9	31
Glorot rovnoměrné	1	1	1	1	5	9	-
Glorot normální	1	1	1	2	4	8	-
Ortonormální	1	1	1	1	3	9	32

## B.2 Hluboká vícevrstvá perceptronová síť

### B.2.1 LSUV



Obr. B.4: Průběhy chyby a správnosti na validační a trénovací sadě databáze CIFAR-10 při učení hluboké vícevrstvé perceptronové sítě pro 5 různých konfigurací metody inicializace vah LSUV. Průběhy jsou zobrazeny do epochy optimálního naučení.

Tab. B.9: Výsledky ohodnocující učení hluboké vícevrstvé perceptronové sítě, která byla inicializovaná metodou LSUV v různých konfiguracích a poté učena na trénovací sadě databáze CIFAR-10. Hodnoty jsou získány pro testovací sadu databáze CIFAR-10. Šestý sloupec vyjadřuje počet epoch k dosažení optimálního naučení sítě. Žluté podbarvení vyjadřuje nejvyšší dosaženou správnost a oranžové podbarvení nejnižší počet epoch.

Metoda LSUV					
typ konfigurace	správnost	průměrná vážená přesnost	průměrná vážená výtěžnost	průměrné vážené F1-skóre	počet epoch
2 obrázky	47,560 %	0,48	0,49	0,48	<b>9</b>
16 obrázků	<b>49,670 %</b>	0,50	0,50	0,50	15
64 obrázků	47,830 %	0,48	0,48	0,46	11
256 obrázků	47,960 %	0,49	0,48	0,47	11
1024 obrázků	47,970 %	0,49	0,48	0,47	<b>9</b>

Tab. B.10: Popis počtu epoch k dosažení 20 %, 25 %, 30 %, 35 %, 40 %, 45 % a 50 % správnosti na validační sadě při učení hluboké vícevrstvé perceptronové sítě, které byla inicializovaná metodou LSUV. Hodnoty počtu epoch jsou vyobrazeny pro 5 různých konfigurací metody. Znak "-" značí nedosažení dané správnosti. Nejlepší výsledek je podbarven modrou.

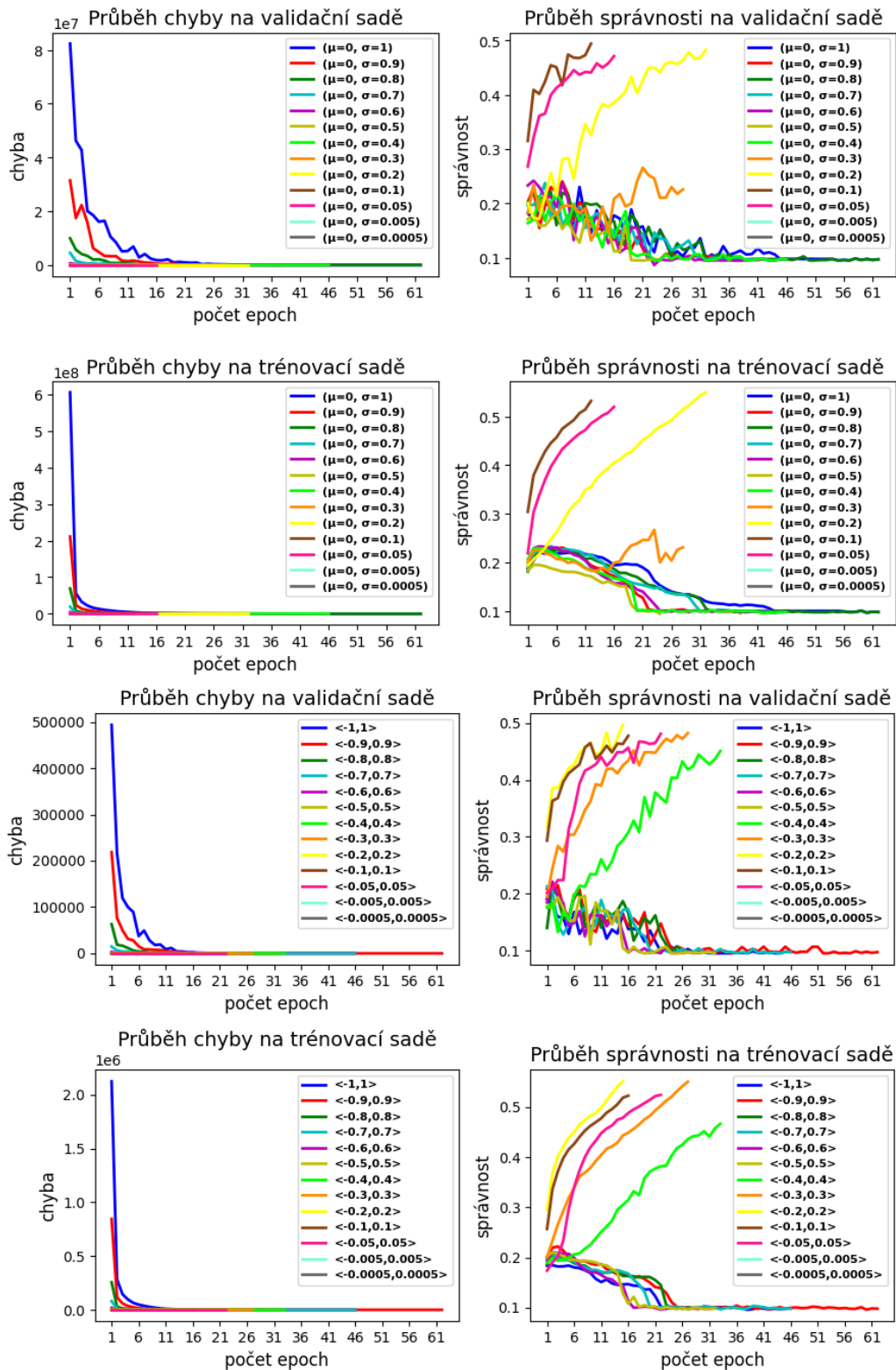
typ konfigurace	Počet epoch k dosažení určité správnosti na validační sadě						
	20 %	25 %	30 %	35 %	40 %	45 %	50 %
2 obrázky	1	1	1	2	4	7	-
16 obrázků	1	1	1	2	5	7	-
64 obrázků	1	1	1	2	4	6	-
256 obrázků	1	1	1	2	4	7	-
1024 obrázků	1	1	1	2	2	7	-



## B.2.2 Náhodná inicializace vah z normálního a rovnoměrného pravděpodobnostního rozdělení

Tab. B.11: Výsledky ohodnocující učení hluboké vícevrstvé perceptronové sítě. Zvolená inicializace vah byla inicializace vah z normálního pravděpodobnostního rozložení v různých konfiguracích. Síť byla učena na trénovací sadě databáze CIFAR-10. Hodnoty jsou získány pro testovací sadu databáze CIFAR-10. Šestý sloupec vyjadřuje počet epoch k dosažení optimálního naučení sítě. Žluté podbarvení vyjadřuje nejvyšší dosaženou správnost a oranžové podbarvení nejnižší počet epoch.

Metoda inicializace z normálního pravděpodobnostního rozložení					
typ konfigurace	správnost	průměrná vážená přesnost	průměrná vážená výtěžnost	průměrné vážené F1-skóre	počet epoch
$(\mu = 0, \sigma = 1)$	10,000 %	0,06	0,10	0,02	62
$(\mu = 0, \sigma = 0,9)$	10,000 %	0,01	0,10	0,02	32
$(\mu = 0, \sigma = 0,8)$	10,000 %	0,01	0,10	0,02	62
$(\mu = 0, \sigma = 0,7)$	10,000 %	0,01	0,10	0,02	46
$(\mu = 0, \sigma = 0,6)$	10,010 %	0,11	0,10	0,02	46
$(\mu = 0, \sigma = 0,5)$	10,000 %	0,01	0,10	0,02	24
$(\mu = 0, \sigma = 0,4)$	10,000 %	0,01	0,10	0,02	46
$(\mu = 0, \sigma = 0,3)$	23,220 %	0,21	0,23	0,17	28
$(\mu = 0, \sigma = 0,2)$	47,990 %	0,49	0,48	0,48	32
$(\mu = 0, \sigma = 0,1)$	<b>48,600 %</b>	0,49	0,49	0,49	<b>12</b>
$(\mu = 0, \sigma = 0,05)$	47,940 %	0,48	0,48	0,48	16
$(\mu=0, \sigma=0,005)$	10,000 %	0,01	0,10	0,02	1
$(\mu = 0, \sigma = 0,0005)$	10,000 %	0,01	0,10	0,02	1



Obr. B.5: Průběhy chyby a správnosti na validační a trénovací sadě databáze CIFAR-10 při učení hluboké vícevrstvé perceptronové sítě pro konfigurace metod inicializace vah z normálního a rovnoměrného pravděpodobnostního rozložení. Horní čtyři grafy vyjadřují průběhy pro inicializace vah z normálního pravděpodobnostního rozložení a čtyři spodní grafy vyjadřují průběhy pro inicializace vah z rovnoměrného pravděpodobnostního rozložení. Průběhy jsou zobrazeny do epochy optimálního naučení.

Tab. B.12: Popis počtu epoch k dosažení 20 %, 25 %, 30 %, 35 %, 40 %, 45 % a 50 % správnosti na validační sadě při učení hluboké vícevrstvé perceptronové sítě, inicializované metodou náhodného inicializace vah z normálního pravděpodobnostního rozložení v různých konfiguracích. Hodnoty počtu epoch jsou vyobrazeny pro 13 různých konfigurací metody. Znak "-" značí nedosažení dané správnosti. Nejlepší výsledek je podbarven modrou.

typ konfigurace	Počet epoch k dosažení určité správnosti na validační sadě						
	20 %	25 %	30 %	35 %	40 %	45 %	50 %
$(\mu = 0, \sigma = 1)$	4	-	-	-	-	-	-
$(\mu = 0, \sigma = 0,9)$	1	-	-	-	-	-	-
$(\mu = 0, \sigma = 0,8)$	1	-	-	-	-	-	-
$(\mu = 0, \sigma = 0,7)$	3	-	-	-	-	-	-
$(\mu = 0, \sigma = 0,6)$	1	-	-	-	-	-	-
$(\mu = 0, \sigma = 0,5)$	-	-	-	-	-	-	-
$(\mu = 0, \sigma = 0,4)$	3	-	-	-	-	-	-
$(\mu = 0, \sigma = 0,3)$	2	21	-	-	-	-	-
$(\mu = 0, \sigma = 0,2)$	1	5	10	13	18	23	-
$(\mu = 0, \sigma = 0,1)$	1	1	1	2	2	5	20
$(\mu = 0, \sigma = 0,05)$	1	1	2	3	6	13	-
$(\mu = 0, \sigma = 0,005)$	-	-	-	-	-	-	-
$(\mu = 0, \sigma = 0,0005)$	-	-	-	-	-	-	-

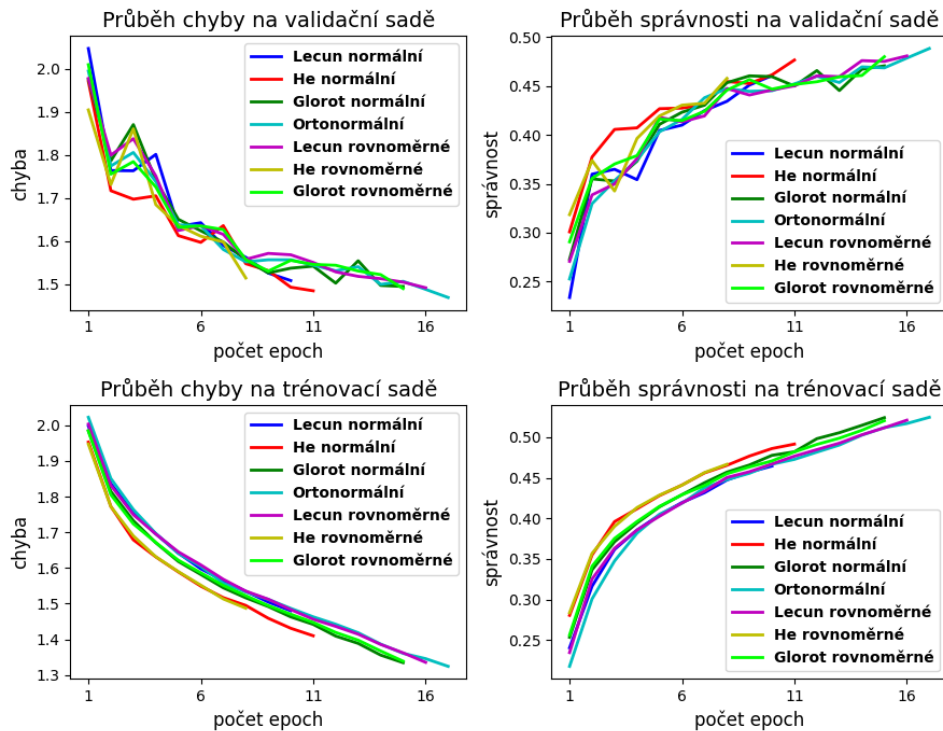
Tab. B.13: Výsledky ohodnocující učení hluboké vícevrstvé perceptronových sítí. Zvolená inicializace vah byla inicializace vah z rovnoměrného pravděpodobnostního rozložení v různých konfiguracích. Síť byla učena na trénovací sadě databáze CIFAR-10. Hodnoty jsou získány pro testovací sadu databáze CIFAR-10. Šestý sloupec vyjadřuje počet epoch k dosažení optimálního naučení sítě. Žluté podbarvení vyjadřuje nejvyšší dosaženou správnost a oranžové podbarvení nejnižší počet epoch.

Metoda inicializace z rovnoměrného pravděpodobnostního rozložení					
typ konfigurace	správnost	průměrná vážená přesnost	průměrná vážená výtěžnost	průměrné vážené F1-skóre	počet epoch
<-1,1>	10,000 %	0,11	0,10	0,02	46
<-0,9;0,9>	10,010 %	0,11	0,10	0,02	62
<-0,8;0,8>	10,000 %	0,01	0,10	0,02	29
<-0,7;0,7>	10,000 %	0,01	0,10	0,02	46
<-0,6;0,6>	10,000 %	0,01	0,10	0,02	32
<-0,5;0,5>	10,000 %	0,01	0,10	0,02	32
<-0,4;0,4>	45,570 %	0,46	0,46	0,46	33
<-0,3;0,3>	48,400 %	0,49	0,48	0,48	27
<-0,2;0,2>	<b>48,820 %</b>	0,50	0,49	0,49	<b>15</b>
<-0,1;0,1>	47,720 %	0,48	0,48	0,48	16
<-0,05;0,05>	47,650 %	0,48	0,48	0,47	22
<-0,005;0,005>	11,350 %	0,01	0,11	0,02	6
<-0,0005;0,0005>	11,350 %	0,01	0,11	0,02	6

Tab. B.14: Popis počtu epoch k dosažení určité správnosti na validační sadě při učení hluboké vícevrstvé perceptronové sítě, inicializované metodou náhodného inicializace vah z rovnoměrného pravděpodobnostního rozložení v různých konfiguracích. Hodnoty počtu epoch jsou vyobrazeny pro 13 různých konfigurací metody. Znak "-" značí nedosažení dané správnosti. Nejlepší výsledek je podbarven modrou.

typ konfigurace	Počet epoch k dosažení určité správnosti na validační sadě						
	20 %	25 %	30 %	35 %	40 %	45 %	50 %
<-1,1>	-	-	-	-	-	-	-
<-0,9;0,9>	1	-	-	-	-	-	-
<-0,8;0,8>	2	-	-	-	-	-	-
<-0,7;0,7>	1	-	-	-	-	-	-
<-0,6;0,6>	2	-	-	-	-	-	-
<-0,5;0,5>	-	-	-	-	-	-	-
<-0,4;0,4>	7	11	16	19	24	33	-
<-0,3;0,3>	2	3	5	9	12	17	-
<-0,2;0,2>	1	1	1	2	4	8	-
<-0,1;0,1>	1	1	2	2	5	8	-
<-0,05;0,05>	1	5	5	7	8	16	-
<-0,005;0,005>	1	5	-	-	-	-	-
<-0,0005;0,0005>	-	-	-	-	-	-	-

### B.2.3 Ostatní metody náhodné inicializace



Obr. B.6: Průběhy chyby a správnosti na validační a trénovací sadě databáze CIFAR-10 při učení hluboké vícevrstvé perceptronové sítě, která byla inicializována vybranými metodami náhodné inicializace. Zvolené inicializace vah byly LeCun, He, Glorot, které se dále dělí podle toho, z jakého pravděpodobnostního rozložení jsou váhy vzorkovány na rovnoměrné a normální. Dále byla použita metoda inicializaci vah pomocí rozkladu na ortonormální bázi. Průběhy jsou zobrazeny do epochy optimálního naučení.

Tab. B.15: Výsledky ohodnocující učení hluboké vícevrstvé perceptronové sítě. Zvolené inicializace vah byly LeCun, He, Glorot, které se dále dělí podle toho, z jakého pravděpodobnostního rozložení jsou váhy vzorkovány na rovnoměrné a normální. Dále byla použita metoda inicializaci vah pomocí rozkladu na ortonormální bázi. Síť byla učena na trénovací sadě databáze CIFAR-10. Hodnoty jsou získány pro testovací sadu databáze CIFAR-10. Šestý sloupec vyjadřuje počet epoch k dosažení optimálního naučení sítě. Žluté podbarvení vyjadřuje nejvyšší dosaženou správnost a oranžové podbarvení nejnižší počet epoch.

Ostatní metody náhodné inicializace					
typ metody	správnost	průměrná vážená přesnost	průměrná vážená výtežnost	průměrné vážené F1-skóre	počet epoch
LeCun rovnoměrné	48,010 %	0,48	0,48	0,48	16
LeCun normální	46,650 %	0,46	0,47	0,46	10
He rovnoměrné	46,900 %	0,47	0,47	0,46	8
He normální	47,660 %	0,48	0,48	0,47	11
Glorot rovnoměrné	<b>48,120 %</b>	0,49	0,48	0,48	15
Glorot normální	47,580 %	0,49	0,48	0,48	15
Ortonormální	47,860 %	0,48	0,48	0,47	17

Tab. B.16: Popis počtu epoch k dosažení určité správnosti na validační sadě při učení hluboké vícevrstvé perceptronové sítě, která byla inicializována vybranými metodami náhodné inicializace. Zvolené inicializace vah byly LeCun, He, Glorot, které se dále dělí podle toho, z jakého pravděpodobnostního rozložení jsou váhy vzorkovány na rovnoměrné a normální. Dále byla použita metoda inicializaci vah pomocí rozkladu na ortonormální bázi. Hodnoty počtu epoch jsou vyobrazeny pro 4 metody a jejich případné konfigurace. Znak "-" značí nedosažení dané správnosti. Nejlepší výsledek je podbarven modrou.

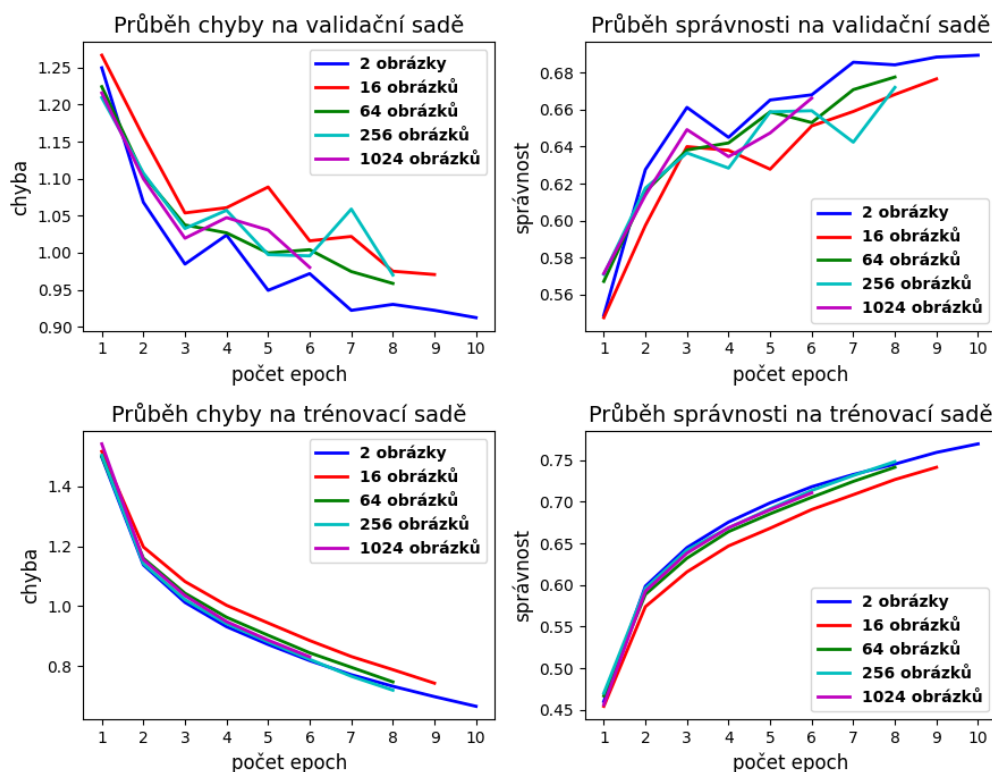
typ konfigurace	Počet epoch k dosažení určité správnosti na validační sadě						
	20 %	25 %	30 %	35 %	40 %	55 %	50 %
LeCun rovnoměrné	1	1	2	4	5	11	-
LeCun normální	1	2	2	2	5	9	-
He rovnoměrné	1	1	1	1	2	7	-
He normální	1	1	1	2	3	8	-
Glorot rovnoměrné	1	1	2	2	5	9	-
Glorot normální	1	1	2	2	5	8	-
Ortonormální	1	1	2	3	5	11	-

## B.2.4 Jednoduchá konvoluční neuronová síť

### LSUV

Tab. B.17: Výsledky ohodnocující učení jednoduché konvoluční neuronové sítě, která byla inicializovaná metodou LSUV v různých konfiguracích a poté učena na trénovací sadě databáze CIFAR-10. Hodnoty jsou získány pro testovací sadu databáze CIFAR-10. Šestý sloupec vyjadřuje počet epoch k dosažení optimálního naučení sítě. Žluté podbarvení vyjadřuje nejvyšší dosaženou správnost a oranžové podbarvení nejnižší počet epoch.

Metoda LSUV					
typ konfigurace	správnost	průměrná vážená přesnost	průměrná vážená výtěžnost	průměrné vážené F1-skóre	počet epoch
2 obrázky	<b>68,480 %</b>	0,69	0,68	0,68	10
16 obrázků	66,360 %	0,66	0,66	0,66	9
64 obrázků	65,920 %	0,67	0,66	0,66	8
256 obrázků	65,350 %	0,67	0,65	0,65	8
1024 obrázků	66,330 %	0,67	0,66	0,66	<b>7</b>



Obr. B.7: Průběhy chyby a správnosti na validační a trénovací sadě databáze CIFAR-10 při učení jednoduché konvoluční neuronové sítě pro 5 různých konfigurací metody inicializace vah LSUV. Průběhy jsou zobrazeny do epochy optimálního naučení.

Tab. B.18: Popis počtu epoch k dosažení 40 %, 45 %, 50 %, 55 %, 60 %, 65 % a 70 % správnosti na validační sadě při učení jednoduché konvoluční neuronové sítě, které byla inicializovaná metodou LSUV. Hodnoty počtu epoch jsou vyobrazeny pro 5 různých konfigurací metody. Znak "-" značí nedosažení dané správnosti. Nejlepší výsledek je podbarven modrou.

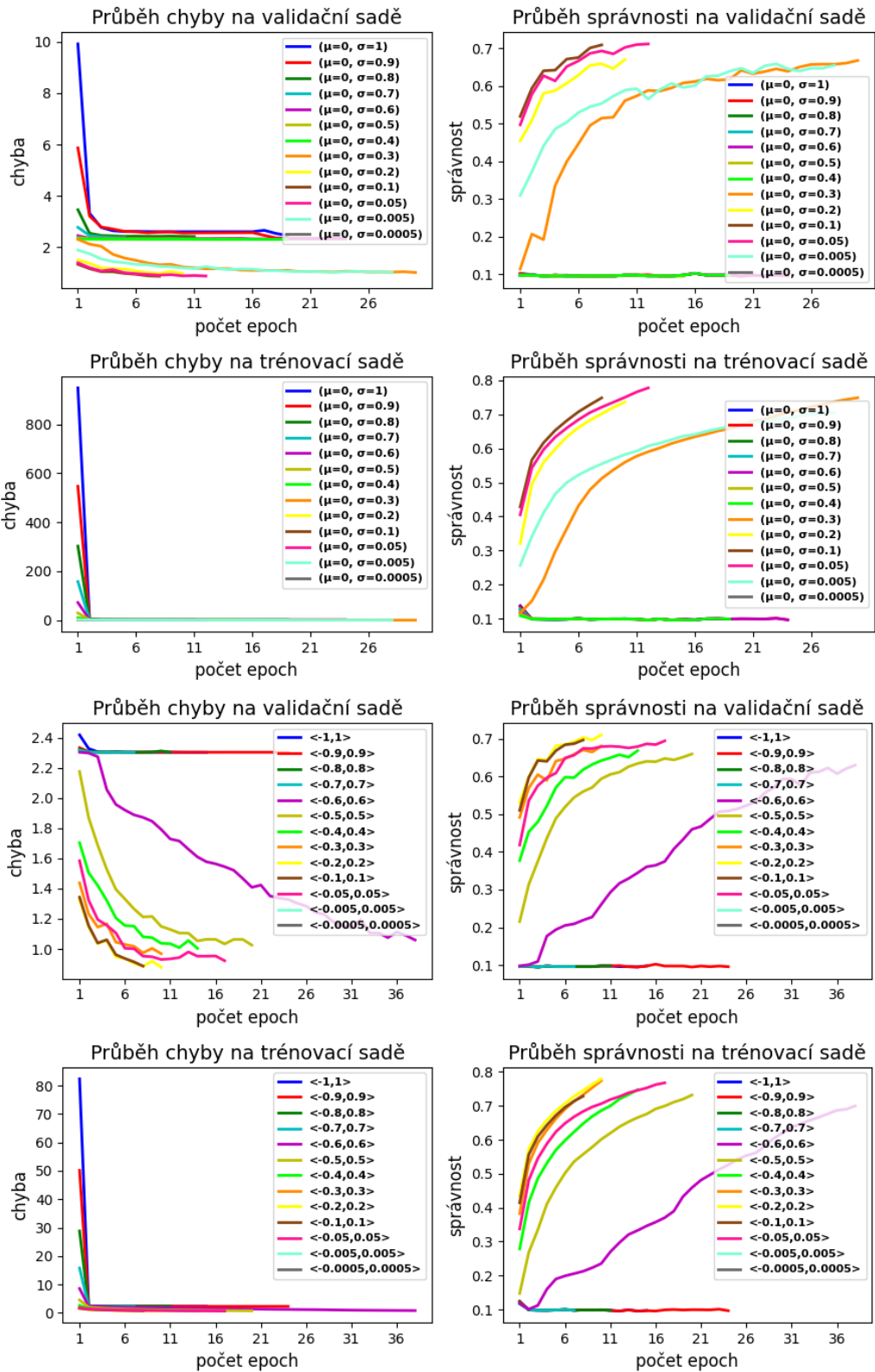
typ konfigurace	Počet epoch k dosažení určité správnosti na validační sadě						
	40 %	45 %	50 %	55 %	60 %	65 %	70 %
2 obrázky	1	1	1	2	2	3	-
16 obrázků	1	1	1	2	3	6	-
64 obrázků	1	1	1	1	2	5	-
256 obrázků	1	1	1	1	2	6	-
1024 obrázků	1	1	1	1	2	5	-



## Náhodná inicializace vah z normálního a rovnoměrného pravděpodobnostního rozložení

Tab. B.19: Výsledky ohodnocující učení jednoduché konvoluční neuronové sítě. Zvolená inicializace vah byla inicializace vah z normálního pravděpodobnostního rozložení v různých konfiguracích. Síť byla učena na trénovací sadě databáze CIFAR-10. Hodnoty jsou získány pro testovací sadu databáze CIFAR-10. Šestý sloupec vyjadřuje počet epoch k dosažení optimálního naučení sítě. Žluté podbarvení vyjadřuje nejvyšší dosaženou správnost a oranžové podbarvení nejnižší počet epoch.

Metoda inicializace z normálního pravděpodobnostního rozložení					
typ konfigurace	správnost	průměrná vážená přesnost	průměrná vážená výťažnost	průměrné vážené F1-skóre	počet epoch
$(\mu = 0, \sigma = 1)$	10,000 %	0,01	0,10	0,02	19
$(\mu = 0, \sigma = 0,9)$	10,010 %	0,11	0,10	0,02	24
$(\mu = 0, \sigma = 0,8)$	9,990 %	0,01	0,10	0,02	11
$(\mu = 0, \sigma = 0,7)$	9,990 %	0,01	0,10	0,02	5
$(\mu = 0, \sigma = 0,6)$	9,990 %	0,01	0,10	0,02	24
$(\mu = 0, \sigma = 0,5)$	9,990 %	0,01	0,10	0,02	15
$(\mu = 0, \sigma = 0,4)$	9,990 %	0,01	0,10	0,02	19
$(\mu = 0, \sigma = 0,3)$	65,740 %	0,66	0,66	0,66	30
$(\mu = 0, \sigma = 0,2)$	66,720 %	0,67	0,67	0,67	10
$(\mu = 0, \sigma = 0,1)$	69,150 %	0,70	0,69	0,69	8
$(\mu = 0, \sigma = 0,05)$	<b>69,360 %</b>	0,70	0,69	0,69	12
$(\mu=0, \sigma=0,005)$	64,590 %	0,65	0,65	0,64	28
$(\mu = 0, \sigma = 0,0005)$	10,000 %	0,01	0,10	0,02	1



Obr. B.8: Průběhy chyby a správnosti na validační a trénovací sadě databáze CIFAR-10 při učení jednoduché konvoluční neuronové sítě pro konfigurace metod inicializace vah z normálního a rovnoměrného pravděpodobnostního rozložení. Horní čtyři grafy vyjadřují průběhy pro inicializace vah z normálního pravděpodobnostního rozložení a čtyři spodní grafy vyjadřují průběhy pro inicializace vah z rovnoměrného pravděpodobnostního rozložení. Průběhy jsou zobrazeny do epochy optimálního naučení.

Tab. B.20: Popis počtu epoch k dosažení 40 %, 45 %, 50 %, 55 %, 60 %, 65 % a 70 % správnosti na validační sadě při učení jednoduché konvoluční neuronové sítě, inicializované metodou náhodného inicializace vah z normálního pravděpodobnostního rozložení v různých konfiguracích. Hodnoty počtu epoch jsou vyobrazeny pro 13 různých konfigurací metody. Znak "-" značí nedosažení dané správnosti. Nejlepší výsledek je podbarven modrou.

typ konfigurace	Počet epoch k dosažení určité správnosti na validační sadě						
	40 %	45 %	50 %	55 %	60 %	65 %	70 %
$(\mu = 0, \sigma = 1)$	-	-	-	-	-	-	-
$(\mu = 0, \sigma = 0,9)$	-	-	-	-	-	-	-
$(\mu = 0, \sigma = 0,8)$	-	-	-	-	-	-	-
$(\mu = 0, \sigma = 0,7)$	-	-	-	-	-	-	-
$(\mu = 0, \sigma = 0,6)$	-	-	-	-	-	-	-
$(\mu = 0, \sigma = 0,5)$	-	-	-	-	-	-	-
$(\mu = 0, \sigma = 0,4)$	-	-	-	-	-	-	-
$(\mu = 0, \sigma = 0,3)$	6	7	8	10	15	25	-
$(\mu = 0, \sigma = 0,2)$	1	1	2	3	5	7	-
$(\mu = 0, \sigma = 0,1)$	1	1	1	2	3	5	7
$(\mu = 0, \sigma = 0,05)$	1	1	2	2	3	5	10
$(\mu = 0, \sigma = 0,005)$	3	4	5	8	14	22	-
$(\mu = 0, \sigma = 0,0005)$	-	-	-	-	-	-	-

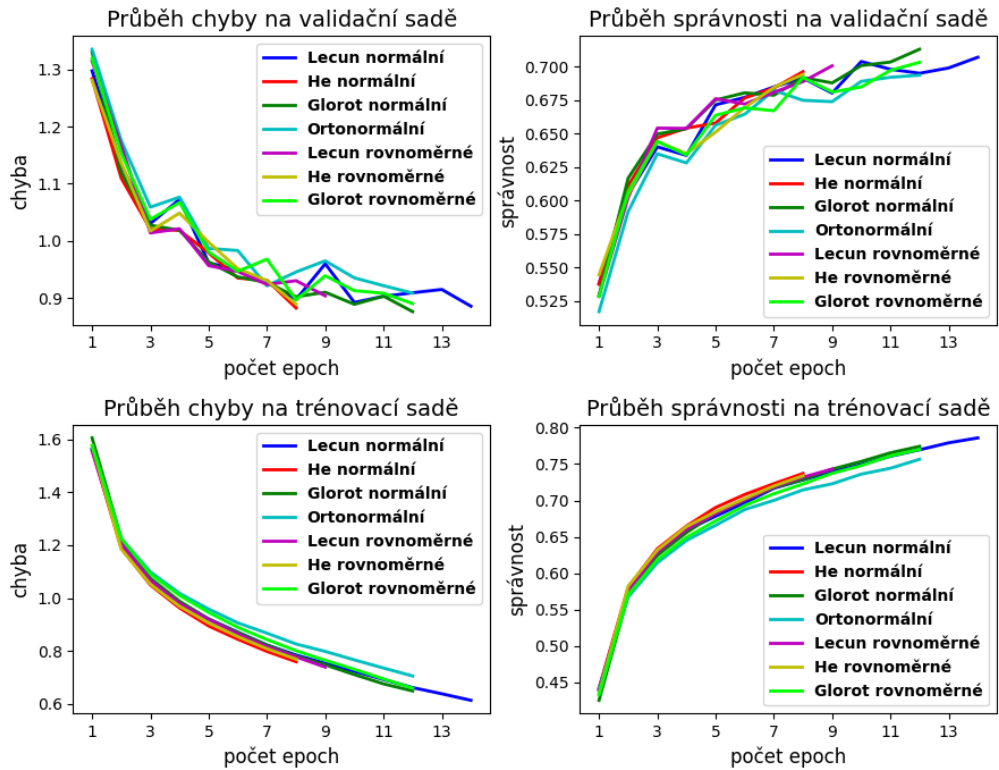
Tab. B.21: Výsledky ohodnocující učení jednoduché konvoluční neuronové sítě. Zvolená inicializace vah byla inicializace vah z rovnoměrného pravděpodobnostního rozložení v různých konfiguracích. Síť byla učena na trénovací sadě databáze CIFAR-10. Hodnoty jsou získány pro testovací sadu databáze CIFAR-10. Šestý sloupec vyjadřuje počet epoch k dosažení optimálního naučení sítě. Žluté podbarvení vyjadřuje nejvyšší dosaženou správnost a oranžové podbarvení nejnižší počet epoch.

Metoda inicializace z rovnoměrného pravděpodobnostního rozložení					
typ konfigurace	správnost	průměrná vážená přesnost	průměrná vážená výtěžnost	průměrné vážené F1-skóre	počet epoch
<-1,1>	9,990 %	0,01	0,10	0,02	15
<-0,9;0,9>	10,010 %	0,10	0,11	0,02	24
<-0,8;0,8>	10,000 %	0,01	0,11	0,02	11
<-0,7;0,7>	10,030 %	0,16	0,10	0,02	21
<-0,6;0,6>	62,470 %	0,62	0,62	0,62	38
<-0,5;0,5>	65,060 %	0,65	0,65	0,65	20
<-0,4;0,4>	66,010 %	0,66	0,66	0,66	14
<-0,3;0,3>	68,160 %	0,68	0,68	0,68	10
<-0,2;0,2>	<b>69,460 %</b>	0,69	0,69	0,69	10
<-0,1;0,1>	68,360 %	0,69	0,68	0,68	<b>8</b>
<-0,05;0,05>	68,240 %	0,68	0,68	0,68	17
<-0,005;0,005>	10,000 %	0,01	0,10	0,02	1
<-0,0005;0,0005>	10,000 %	0,01	0,10	0,02	1

Tab. B.22: Popis počtu epoch k dosažení určité správnosti na validační sadě při učení jednoduché konvoluční neuronové sítě, inicializované metodou náhodného inicializace vah z rovnoměrného pravděpodobnostního rozložení v různých konfiguracích. Hodnoty počtu epoch jsou vyobrazeny pro 13 různých konfigurací metody. Znak "-" značí nedosažení dané správnosti. Nejlepší výsledek je podbarven modrou.

typ konfigurace	Počet epoch k dosažení určité správnosti na validační sadě						
	40 %	45 %	50 %	55 %	60 %	65 %	70 %
<-1,1>	-	-	-	-	-	-	-
<-0,9;0,9>	-	-	-	-	-	-	-
<-0,8;0,8>	-	-	-	-	-	-	-
<-0,7;0,7>	-	-	-	-	-	-	-
<-0,6;0,6>	18	20	23	28	33	-	-
<-0,5;0,5>	4	5	6	8	11	19	-
<-0,4;0,4>	2	2	4	5	8	12	-
<-0,3;0,3>	1	1	2	2	3	7	-
<-0,2;0,2>	1	1	1	2	3	5	8
<-0,1;0,1>	1	1	1	2	3	5	10
<-0,05;0,05>	1	2	2	3	5	7	-
<-0,005;0,005>	-	-	-	-	-	-	-
<-0,0005;0,0005>	-	-	-	-	-	-	-

## Ostatní metody náhodné inicializace



Obr. B.9: Průběhy chyby a správnosti na validační a trénovací sadě databáze CIFAR-10 při učení jednoduché konvoluční neuronové sítě, která byla inicializována vybranými metodami náhodné inicializace. Zvolené inicializace vah byly LeCun, He, Glorot, které se dále dělí podle toho, z jakého pravděpodobnostního rozložení jsou váhy vzorkovány na rovnoměrné a normální. Dále byla použita metoda inicializaci vah pomocí rozkladu na ortonormální bázi. Průběhy jsou zobrazeny do epochy optimálního naučení.

Tab. B.23: Výsledky ohodnocující učení jednoduché konvoluční neuronové sítě. Zvolené inicializace vah byly LeCun, He, Glorot, které se dále dělí podle toho, z jakého pravděpodobnostního rozložení jsou váhy vzorkovány na rovnoměrné a normální. Dále byla použita metoda inicializaci vah pomocí rozkladu na ortonormální bázi. Síť byla učena na trénovací sadě databáze CIFAR-10. Hodnoty jsou získány pro testovací sadu databáze CIFAR-10. Šestý sloupec vyjadřuje počet epoch k dosažení optimálního naučení sítě. Žluté podbarvení vyjadřuje nejvyšší dosaženou správnost a oranžové podbarvení nejnižší počet epoch.

Ostatní metody náhodné inicializace					
typ metody	správnost	průměrná vážená přesnost	průměrná vážená výtežnost	průměrné vážené F1-skóre	počet epoch
LeCun rovnoměrné	68,940 %	0,69	0,69	0,68	9
LeCun normální	<b>70,370 %</b>	0,71	0,70	0,70	14
He rovnoměrné	68,240 %	0,69	0,68	0,68	<b>8</b>
He normální	69,350 %	0,70	0,69	0,69	<b>8</b>
Glorot rovnoměrné	69,210 %	0,70	0,69	0,69	12
Glorot normální	68,990 %	0,70	0,69	0,69	12
Ortonormální	67,940 %	0,69	0,68	0,68	12

Tab. B.24: Popis počtu epoch k dosažení určité správnosti na validační sadě při učení jednoduché konvoluční neuronové sítě, která byla inicializována vybranými metodami náhodné inicializace. Zvolené inicializace vah byly LeCun, He, Glorot, které se dále dělí podle toho, z jakého pravděpodobnostního rozložení jsou váhy vzorkovány na rovnoměrné a normální. Dále byla použita metoda inicializaci vah pomocí rozkladu na ortonormální bázi. Hodnoty počtu epoch jsou vyobrazeny pro 4 metody a jejich případné konfigurace. Znak "-" značí nedosažení dané správnosti. Nejlepší výsledek je podbarven modrou.

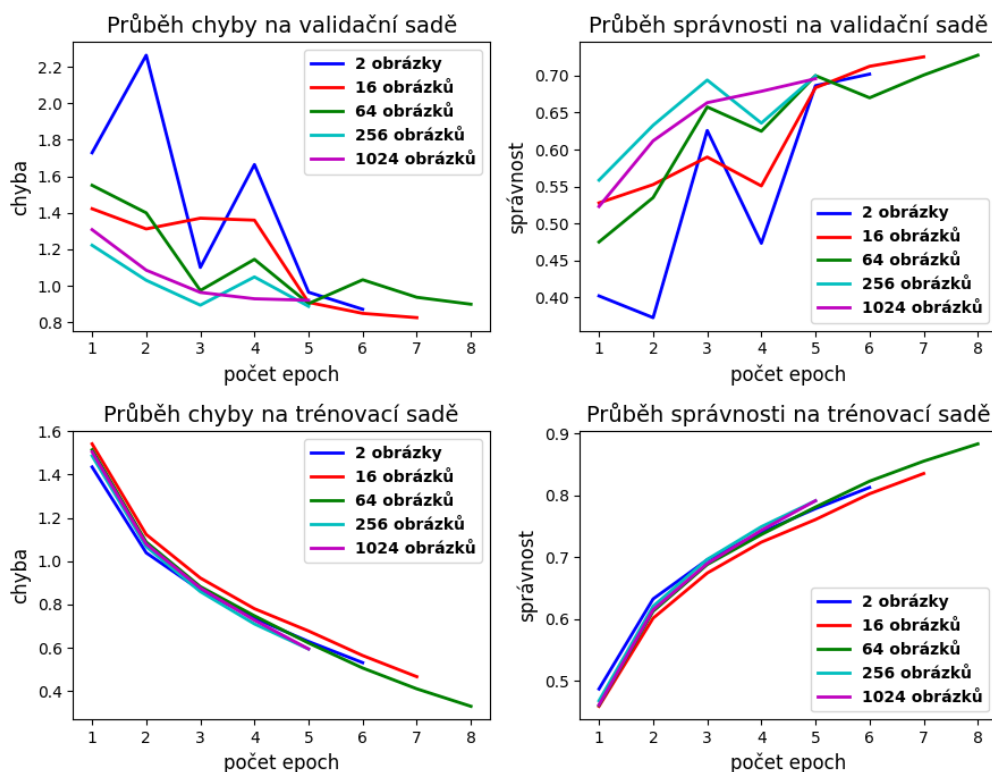
typ konfigurace	Počet epoch k dosažení určité správnosti na validační sadě						
	40 %	45 %	50 %	55 %	60 %	65 %	70 %
LeCun rovnoměrné	1	1	1	2	2	3	9
LeCun normální	1	1	1	2	2	5	10
He rovnoměrné	1	1	1	2	2	5	17
He normální	1	1	1	2	2	4	9
Glorot rovnoměrné	1	1	1	2	2	5	12
Glorot normální	1	1	1	1	2	4	10
Ortonormální	1	1	1	2	3	5	17

## B.2.5 Hluboká konvoluční neuronová síť – ResNet18

### LSUV

Tab. B.25: Výsledky ohodnocující učení hluboké konvoluční neuronové sítě – ResNet18, která byla inicializovaná metodou LSUV v různých konfiguracích a poté učena na trénovací sadě databáze CIFAR-10. Hodnoty jsou získány pro testovací sadu databáze CIFAR-10. Šestý sloupec vyjadřuje počet epoch k dosažení optimálního naučení sítě. Žluté podbarvení vyjadřuje nejvyšší dosaženou správnost a oranžové podbarvení nejnižší počet epoch.

Metoda LSUV					
typ konfigurace	správnost	průměrná vážená přesnost	průměrná vážená výtěžnost	průměrné vážené F1-skóre	počet epoch
2 obrázky	70,730 %	0,72	0,71	0,71	6
16 obrázků	71,220 %	0,72	0,71	0,71	7
64 obrázků	<b>72,170 %</b>	0,72	0,72	0,72	8
256 obrázků	68,840 %	0,69	0,69	0,68	<b>5</b>
1024 obrázků	68,310 %	0,69	0,68	0,68	<b>5</b>



Obr. B.10: Průběhy chyby a správnosti na validační a trénovací sadě databáze CIFAR-10 při učení hluboké konvoluční neuronové sítě – ResNet18, pro 5 různých konfigurací metody inicializace vah LSUV. Průběhy jsou zobrazeny do epochy optimálního naučení.

Tab. B.26: Popis počtu epoch k dosažení 45 %, 50 %, 55 %, 60 %, 65 %, 70 % a 75 % správnosti na validační sadě při učení hluboké konvoluční neuronové sítě – ResNet18, které byla inicializovaná metodou LSUV. Hodnoty počtu epoch jsou vyobrazeny pro 5 různých konfigurací metody. Znak "-" značí nedosažení dané správnosti. Nejlepší výsledek je podbarven modrou.

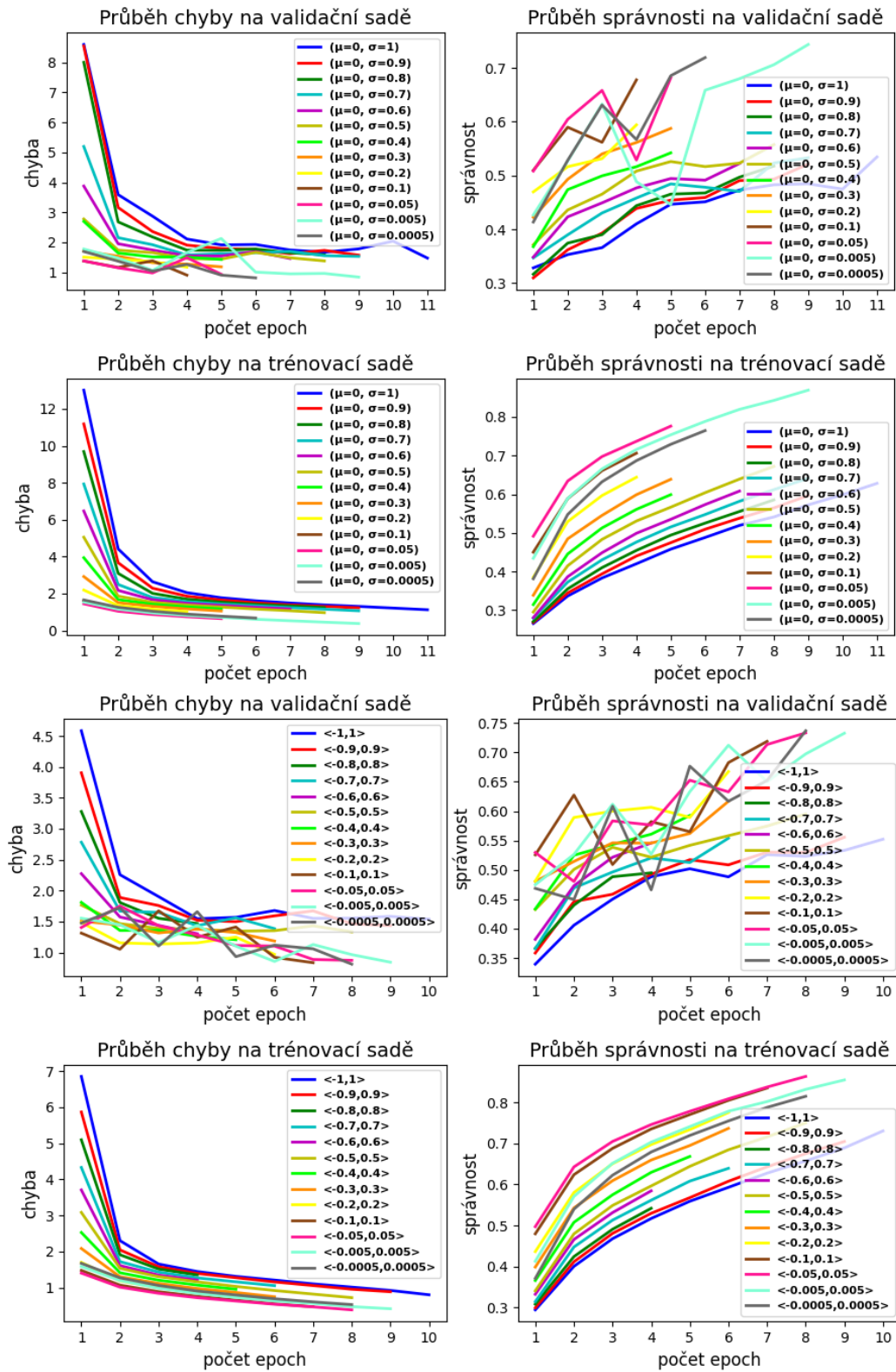
typ konfigurace	Počet epoch k dosažení určité správnosti na validační sadě						
	45 %	50 %	55 %	60 %	65 %	70 %	75 %
2 obrázky	3	3	3	3	5	6	-
16 obrázků	1	1	2	5	5	6	-
64 obrázků	1	2	3	3	3	7	13
256 obrázků	1	1	1	2	3	9	-
1024 obrázků	1	1	2	2	3	7	-



## Náhodná inicializace vah z normálního a rovnoměrného pravděpodobnostního rozložení

Tab. B.27: Výsledky ohodnocující učení hluboké konvoluční neuronové sítě – ResNet18. Zvolená inicializace vah byla inicializace vah z normálního pravděpodobnostního rozložení v různých konfiguracích. Síť byla učena na trénovací sadě databáze CIFAR-10. Hodnoty jsou získány pro testovací sadu databáze CIFAR-10. Šestý sloupec vyjadřuje počet epoch  $k$  dosažení optimálního naučení sítě. Žluté podbarvení vyjadřuje nejvyšší dosaženou správnost a oranžové podbarvení nejnižší počet epoch.

Metoda inicializace z normálního pravděpodobnostního rozložení					
typ konfigurace	správnost	průměrná vážená přesnost	průměrná vážená výťažnost	průměrné vážené F1-skóre	počet epoch
$(\mu = 0, \sigma = 1)$	52,090 %	0,54	0,52	0,52	11
$(\mu = 0, \sigma = 0,9)$	49,670 %	0,50	0,50	0,49	9
$(\mu = 0, \sigma = 0,8)$	50,260 %	0,52	0,50	0,51	8
$(\mu = 0, \sigma = 0,7)$	51,200 %	0,53	0,51	0,51	9
$(\mu = 0, \sigma = 0,6)$	51,470 %	0,54	0,51	0,52	7
$(\mu = 0, \sigma = 0,5)$	54,750 %	0,55	0,55	0,54	8
$(\mu = 0, \sigma = 0,4)$	52,670 %	0,53	0,53	0,52	5
$(\mu = 0, \sigma = 0,3)$	57,890 %	0,58	0,58	0,58	5
$(\mu = 0, \sigma = 0,2)$	58,370 %	0,61	0,58	0,57	4
$(\mu = 0, \sigma = 0,1)$	66,080 %	0,68	0,66	0,66	4
$(\mu = 0, \sigma = 0,05)$	67,020 %	0,69	0,67	0,66	5
$(\mu=0, \sigma=0,005)$	<b>73,380 %</b>	0,75	0,73	0,73	9
$(\mu = 0, \sigma = 0,0005)$	71,230 %	0,72	0,71	0,71	6



Obr. B.11: Průběhy chyby a správnosti na validační a trénovací sadě databáze CIFAR-10 při učení hluboké konvoluční neuronové sítě – ResNet18 pro konfigurace metod inicializace vah z normálního a rovnoměrného pravděpodobnostního rozložení. Horní čtyři grafy vyjadřují průběhy pro inicializace vah z normálního pravděpodobnostního rozložení a čtyři spodní grafy vyjadřují průběhy pro inicializace vah z rovnoměrného pravděpodobnostního rozložení. Průběhy jsou zobrazeny do epochy optimálního naučení.

Tab. B.28: Popis počtu epoch k dosažení 45 %, 50 %, 55 %, 60 %, 65 % a 70 %, 75 % správnosti na validační sadě při učení hluboké konvoluční neuronové sítě – ResNet18 inicializované metodou náhodného inicializace vah z normálního pravděpodobnostního rozložení v různých konfiguracích. Hodnoty počtu epoch jsou vyobrazeny pro 13 různých konfigurací metody. Znak "-" značí nedosažení dané správnosti. Nejlepší výsledek je podbarven modrou.

typ konfigurace	Počet epoch k dosažení určité správnosti na validační sadě						
	45 %	50 %	55 %	60 %	65 %	70 %	75 %
$(\mu = 0, \sigma = 1)$	6	11	19	-	-	-	-
$(\mu = 0, \sigma = 0,9)$	5	9	19	-	-	-	-
$(\mu = 0, \sigma = 0,8)$	5	8	17	-	-	-	-
$(\mu = 0, \sigma = 0,7)$	4	8	11	-	-	-	-
$(\mu = 0, \sigma = 0,6)$	4	7	10	-	-	-	-
$(\mu = 0, \sigma = 0,5)$	3	4	8	-	-	-	-
$(\mu = 0, \sigma = 0,4)$	2	4	8	-	-	-	-
$(\mu = 0, \sigma = 0,3)$	2	3	4	10	-	-	-
$(\mu = 0, \sigma = 0,2)$	1	2	4	8	11	-	-
$(\mu = 0, \sigma = 0,1)$	1	1	2	4	4	9	-
$(\mu = 0, \sigma = 0,05)$	2	1	2	2	3	9	-
$(\mu = 0, \sigma = 0,005)$	2	2	3	3	6	8	13
$(\mu = 0, \sigma = 0,0005)$	2	2	3	3	5	6	16

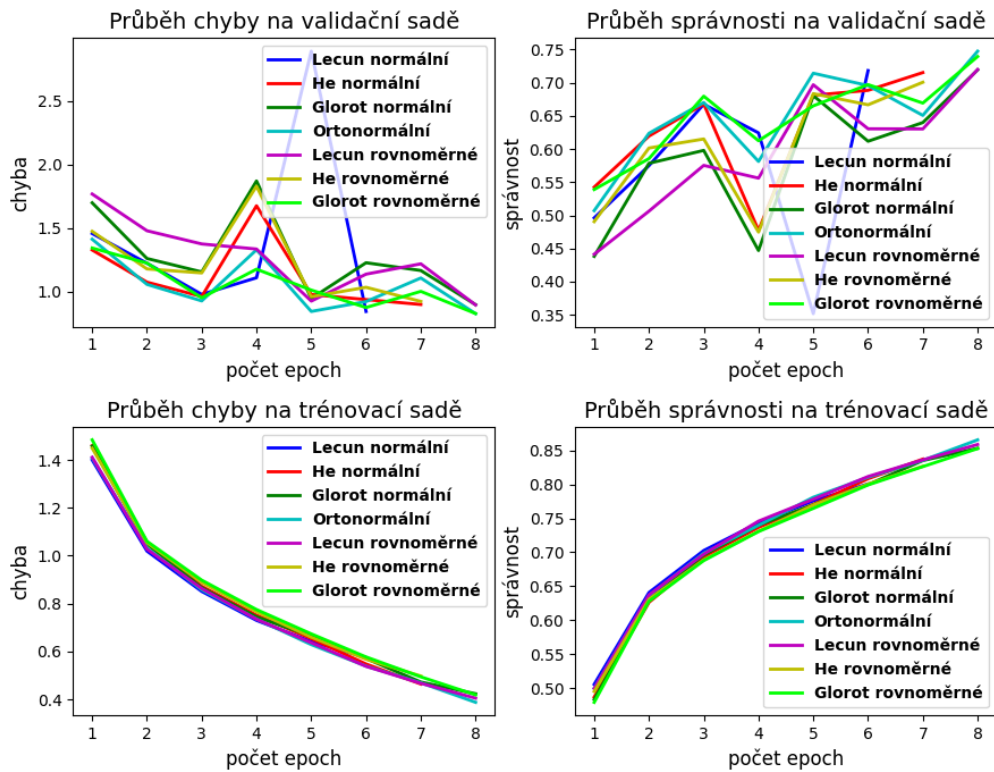
Tab. B.29: Výsledky ohodnocující učení hluboké konvoluční neuronové sítě – ResNet18. Zvolená inicializace vah byla inicializace vah z rovnoměrného pravděpodobnostního rozložení v různých konfiguracích. Síť byla učena na trénovací sadě databáze CIFAR-10. Hodnoty jsou získány pro testovací sadu databáze CIFAR-10. Šestý sloupec vyjadřuje počet epoch k dosažení optimálního naučení sítě. Žluté podbarvení vyjadřuje nejvyšší dosaženou správnost a oranžové podbarvení nejnižší počet epoch.

Metoda inicializace z rovnoměrného pravděpodobnostního rozložení					
typ konfigurace	správnost	průměrná vážená přesnost	průměrná vážená výtěžnost	průměrné vážené F1-skóre	počet epoch
<-1,1>	54,640 %	0,56	0,55	0,55	10
<-0,9;0,9>	53,910 %	0,56	0,54	0,54	9
<-0,8;0,8>	49,060 %	0,50	0,50	0,49	4
<-0,7;0,7>	54,090 %	0,56	0,54	0,54	6
<-0,6;0,6>	53,100 %	0,53	0,53	0,53	4
<-0,5;0,5>	59,290 %	0,60	0,59	0,58	8
<-0,4;0,4>	58,400 %	0,59	0,58	0,57	5
<-0,3;0,3>	61,160 %	0,63	0,63	0,61	6
<-0,2;0,2>	65,520 %	0,67	0,66	0,66	6
<-0,1;0,1>	71,420 %	0,74	0,71	0,72	7
<-0,05;0,05>	72,650 %	0,74	0,73	0,72	8
<-0,005;0,005>	72,750 %	0,74	0,73	0,73	9
<-0,0005;0,0005>	72,240 %	0,74	0,72	0,72	8

Tab. B.30: Popis počtu epoch k dosažení určité správnosti na validační sadě při učení hluboké konvoluční neuronové sítě – ResNet18, inicializované metodou náhodného inicializace vah z rovnoměrného pravděpodobnostního rozložení v různých konfiguracích. Hodnoty počtu epoch jsou vyobrazeny pro 13 různých konfigurací metody. Znak "-" značí nedosažení dané správnosti. Nejlepší výsledek je podbarven modrou.

typ konfigurace	Počet epoch k dosažení určité správnosti na validační sadě						
	45 %	50 %	55 %	60 %	65 %	70 %	75 %
<-1,1>	3	5	10	-	-	-	-
<-0,9;0,9>	3	5	9	-	-	-	-
<-0,8;0,8>	3	5	11	-	-	-	-
<-0,7;0,7>	2	4	6	-	-	-	-
<-0,6;0,6>	2	3	5	11	-	-	-
<-0,5;0,5>	2	2	6	10	-	-	-
<-0,4;0,4>	2	2	4	7	-	-	-
<-0,3;0,3>	1	2	5	6	11	-	-
<-0,2;0,2>	1	2	2	3	6	16	-
<-0,1;0,1>	1	1	2	2	6	7	-
<-0,05;0,05>	1	1	3	5	5	7	12
<-0,005;0,005>	1	2	3	3	6	6	12
<-0,0005;0,0005>	1	3	3	3	5	8	15

## Ostatní metody náhodné inicializace



Obr. B.12: Průběhy chyby a správnosti na validační a trénovací sadě databáze CIFAR-10 při učení hluboké konvoluční neuronové sítě – ResNet18, která byla inicializována vybranými metodami náhodné inicializace. Zvolené inicializace vah byly LeCun, He, Glorot, které se dále dělí podle toho, z jakého pravděpodobnostního rozložení jsou váhy vzorkovány na rovnoměrné a normální. Dále byla použita s metodou inicializace vah pomocí rozkladu na ortonormální bázi. Průběhy jsou zobrazeny do epochy optimálního naučení.

Tab. B.31: Výsledky ohodnocující učení hluboké konvoluční neuronové sítě – ResNet18. Zvolené inicializace vah byly LeCun, He, Glorot, které se dále dělí podle toho, z jakého pravděpodobnostního rozložení jsou váhy vzorkovány na rovnoměrné a normální. Dále byla použita metoda inicializaci vah pomocí rozkladu na ortonormální bázi. Síť byla učena na trénovací sadě databáze CIFAR–10. Hodnoty jsou získány pro testovací sadu databáze CIFAR–10. Šestý sloupec vyjadřuje počet epoch k dosažení optimálního naučení sítě. Žluté podbarvení vyjadřuje nejvyšší dosaženou správnost a oranžové podbarvení nejnižší počet epoch.

Ostatní metody náhodné inicializace					
typ metody	správnost	průměrná vážená přesnost	průměrná vážená výťažnost	průměrné vážené F1–skóre	počet epoch
LeCun rovnoměrné	71,640 %	0,74	0,72	0,71	8
LeCun normální	70,990 %	0,74	0,71	0,71	<b>6</b>
He rovnoměrné	69,330 %	0,74	0,69	0,70	7
He normální	69,990 %	0,73	0,70	0,70	7
Glorot rovnoměrné	72,720 %	0,75	0,73	0,73	8
Glorot normální	70,090 %	0,74	0,70	0,71	8
Ortonormální	<b>72,820 %</b>	0,73	0,73	0,73	8

Tab. B.32: Popis počtu epoch k dosažení určité správnosti na validační sadě při učení hluboké konvoluční neuronové sítě – ResNet18, které byla inicializována vybranými metodami náhodné inicializace. Zvolené inicializace vah byly LeCun, He, Glorot, které se dále dělí podle toho, z jakého pravděpodobnostního rozložení jsou váhy vzorkovány na rovnoměrné a normální. Dále byla použita metoda inicializaci vah pomocí rozkladu na ortonormální bázi. Hodnoty počtu epoch jsou vyobrazeny pro 4 metody a jejich případné konfigurace. Znak "-" značí nedosažení dané správnosti. Nejlepší výsledek je podbarven modrou.

typ konfigurace	Počet epoch k dosažení určité správnosti na validační sadě						
	45 %	50 %	55 %	60 %	65 %	70 %	75 %
LeCun rovnoměrné	2	2	3	5	5	8	-
LeCun normální	1	2	2	3	3	6	-
He rovnoměrné	1	2	2	2	5	7	-
He normální	1	1	2	2	3	7	-
Glorot rovnoměrné	1	1	2	3	3	8	17
Glorot normální	2	2	2	5	5	8	-
Ortonormální	1	1	2	2	3	5	18

## C Obsah elektronické přílohy

Elektronická příloha s názvem *Příloha.zip* obsahuje celkem 5 souborů. V souborech *BP\_CNN.ipynb* a *BP\_MLP.ipynb* je obsažen kód použitý pro trénování a testování neuronových sítí. Soubor *BP\_CNN.ipynb* obsahuje kód pro proběhlé experimenty pro jednoduchou konvoluční neuronové síti a ResNet18. *BP\_MLP.ipynb* obsahuje kód pro proběhlé experimenty pro mělkou vícevrstvou perceptronovskou síť a hlubokou vícevrstvou perceptronovskou síť.

Dále soubor *Příloha.zip* obsahuje soubory *BP\_CNN.txt*, *BP\_plotting\_graphs.ipynb* a *BP\_MLP.txt* a . Soubory *BP\_CNN.txt* a *BP\_MLP.txt* obsahují veškeré hodnoty pro průběhy chyby a správnosti pro validační a trénovací sadu. Díky těmto hodnotám byly pomocí kódu ve skriptu *BP\_plotting\_graphs.ipynb* generovány použité grafy v bakalářské práci.