



FAKULTA APLIKOVANÝCH VĚD
ZÁPADOČESKÉ UNIVERZITY
V PLZNI

KATEDRA
KYBERNETIKY



Základy klasifikace neuronovými sítěmi

Západočeská Univerzita V Plzni
Katedra Kybernetiky
Projekt 4

Vladimíra Kimlová
2. ročník
21. března 2022

Obsah

1	Zadání	1
2	Vypracování	1
2.1	Klasifikace ručně psaných číslic (MNIST dataset)	2
2.2	Klasifikace diagnóz pacientů s onemocněním srdce (Heart dataset)	4
3	Závěr	6

Seznam obrázků

1	Trénovací vzorek – číslo 5	2
2	Průměrné hodnoty, maximální hodnoty a unikátní pixely pro třídu 5	3
3	Počet aktivních tříd pro 1 000 hodnot	3
4	Hodnoty accuracy pro různě velké sítě s výchozími parametry trénování	4
5	ROC křivka pro testovací data	5
6	Matice záměn pro testovací data	6
7	ROC křivka pro celý dataset	6

1 Zadání

Cílem projektu je naučit se pracovat s vybranými frameworky, které využívají principy neuronových sítí k řešení klasifikačních problémů. Získané znalosti umožní komplexní řešení obecné klasifikační úlohy od předzpracování dat po vyhodnocení výsledků a budou základem pro navazující bakalářskou práci.

Postup práce / podúlohy

- seznámení se s teorií a principy neuronových sítí
- poskládání dat pro klasifikační úlohu
- sestavení modelu neuronové sítě
- trénování sítě
- vyhodnocení výsledků klasifikace
- sepsání dokumentace (2-4 strany)

Další dovednosti získané v rámci práce na projektu

- verzovací systém Git (sdílení kódu přes repozitář na GitHubu)
- práce na vzdáleném stroji přes SSH připojení
- práce v programovacím jazyce Python ve virtuálním prostředí nástroje Anaconda
- psaní dokumentace v LaTeXu

Technologie

Python, Keras, Scikit-Learn, PyTorch, Git, SSH, Anaconda, LaTeX

Repo projektu

<https://github.com/kitt10/jiv1>

2 Vypracování

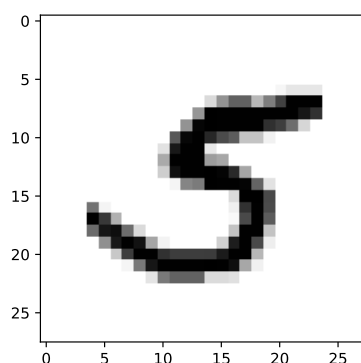
Projekt byl vypracován v Pythonu – konkrétně v JupyterLabu. V několika prvních úlohách byl využíván vzorový kód Simple MNIST convnet ¹ a bylo pracováno i se samotným MNIST

¹https://keras.io/examples/vision/mnist_convnet/

datasetem, tj. ručně psanými číslicemi. Ve druhé části projektu byl použit dataset Heart Disease UCI².

2.1 Klasifikace ručně psaných číslic (MNIST dataset)

V rámci tohoto úkolu byly analyzovány trénovací vzorky, tj. matice 28×28 s hodnotami 0 – 255 představující šedotónový obrázek ručně psaného čísla, a pro seznámení se s úlohou byly vypsaný "targety" k prvním 10 vzorkům z testovací části datasetu, tedy čísla od 0 do 9.



Obrázek 1: Trénovací vzorek – číslo 5

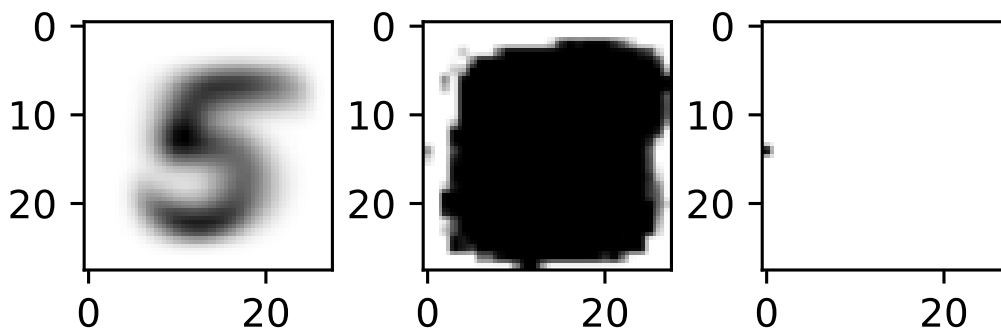
Dále bylo trénování neuronové sítě spuštěno pouze na prvních 10 000 vzorcích. Tímto došlo k nepatrnému poklesu úspěšnosti klasifikace z původních 99.22 % (pro výchozích 60 000 trénovacích vzorků) na 98.0 %. Protože vzorky v trénovací množině nejsou organizovaně řazeny, je zastoupení všech deseti tříd mezi prvními 10 000 vzorky statisticky rovnoměrné. Také díky tomuto faktu je pokles úspěšnosti klasifikace na testovací sadě minimální.

Následně byl klasifikační problém zúžen pouze na rozpoznávání čísel 8 a 6. Došlo tím k redukování počtu trénovacích dat na přibližně 12 000 a testovacích na přibližně 2 000, zastoupení obou tříd bylo rovnoměrné. Přesnost neuronové sítě po předložení takového klasifikačního problému byla 99.8964 %.

Jednotlivé vzorky byly dále analyzovány pro každou třídu separátně, tedy pro jednotlivé číslice 0 až 9. Byly zjištěny průměrné hodnoty, maximální hodnoty pro každý pixel přes všechny vzorky pro jednotlivé třídy a unikátní pixely, tj. pixely, které jsou nenulové pro vzorky jen jediné třídy (viz obrázek 2).

Nakonec byl pro každý pixel v obrázku 28×28 znázorněn počet aktivních tříd. Třída byla označena za aktivní, pokud pro daný pixel měl určitý počet vzorků (= ručně psaných číslic z dané třídy) nenulovou hodnotu. Samotná matice aktivních pixelů, pak byla naplněna čísly od 0

²<https://www.kaggle.com/datasets/cherngs/heart-disease-cleveland-uci>

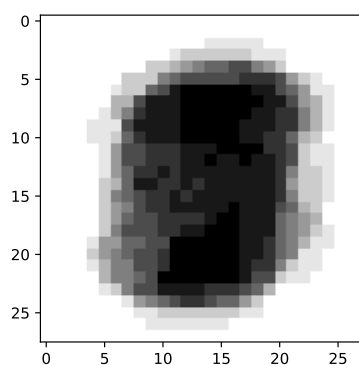


Obrázek 2: Průměrné hodnoty, maximální hodnoty a unikátní pixely pro třídu 5

do počtu tříd (10) – pro každý pixel bylo tedy zřejmé kolik tříd má na něm dostatečný počet zástupců s nenulovou hodnotou. Aktivnost třídy byla dána deseti, stem, tisícem a pěti tisíci nenulovými hodnotami pro daný pixel přes všechny třídy (viz obrázek 3).

```
for col in range(28): # iterace pres sloupce matice
    for row in range(28): # iterace pres radky matice
        for number in classes: # iterace pres jednotlivé tridy - cislice
            samples = 0
            for val in number: # iterace pres jednotlivé zastupce tridy
                if val[row][col] != 0: # nenulova hodnota na danem pixelu
                    samples += 1
                if samples >= count: # aktivni pixely splnuji danou mez
                    active_pixs[row][col] += 1
                    break
```

Na základě takového rozboru lze pak určit důležitost jednotlivých pixelů, a navíc i jejich klíčovou roli v rámci jediné třídy.



Obrázek 3: Počet aktivních tříd pro 1 000 hodnot

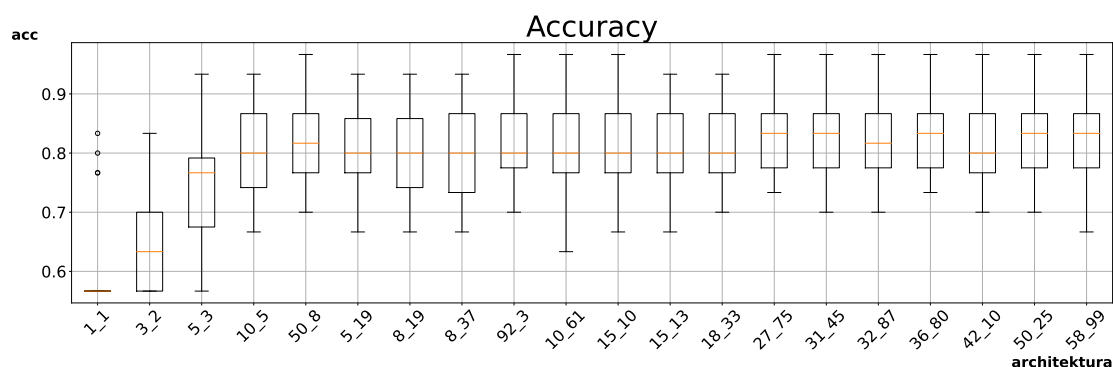
2.2 Klasifikace diagnóz pacientů s onemocněním srdce (Heart dataset)

Tato část projektu byla zaměřena na binární klasifikaci, konkrétně na určení, zda se u daného pacienta jedná o chorobu srdce či nikoliv. Dataset obsahoval 303 pacientů s třinácti diagnostickými údaji (věk, pohlaví, krevní tlak, ...). Tato data byla nejprve normalizována, tj. převedena na hodnoty od 0 do 1. Dále byla rozdělena na data trénovací, testovací a validační v poměru 8 : 1 : 1. Architektura neuronové sítě byla zvolena jako Multilayer perceptron s aktivací funkcí sigmoid. Parametry neuronové sítě a parametry trénovacího procesu byly postupně měněny za účelem jejího natrénování na co nejlepší výsledky.

Výchozí parametry trénovacího procesu:

learning rate	0.0005
batch size	16
epochy	100
optimizer	Adam

Nejprve byly ponechány výchozí parametry trénovacího procesu a měněny byly pouze počty neuronů ve skrytých vrstvách od 1 do 1000, tyto změny neměly diametrální vliv na accuracy a loss (kromě úplně nejmenších sítí - viz obrázek 4).



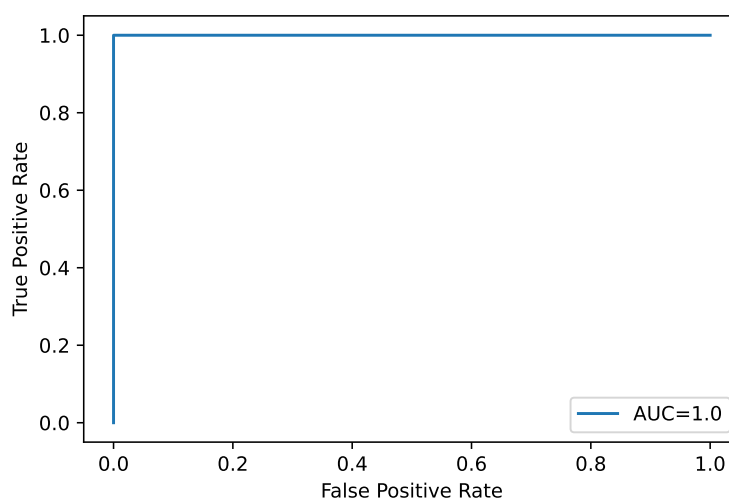
Obrázek 4: Hodnoty accuracy pro různé velké sítě s výchozími parametry trénování

Posléze bylo vybráno 10 kombinací poměrně malých neuronových sítí (počty neuronů ve skrytých vrstvách se pohybovaly do 50), u nichž byly měněny i další parametry jako learning rate, batch size, počet epoch a optimizery. Celkový počet těchto testovaných neuronových sítí dosáhl 3 600, pro každou z nich bylo ještě provedeno 5 realizací, aby statistika byla více vypovídající. Nejlepších výsledků dosáhla síť s 5 neurony v první skryté vrstvě a 3 ve druhé s následujícími parametry trénovacího procesu:

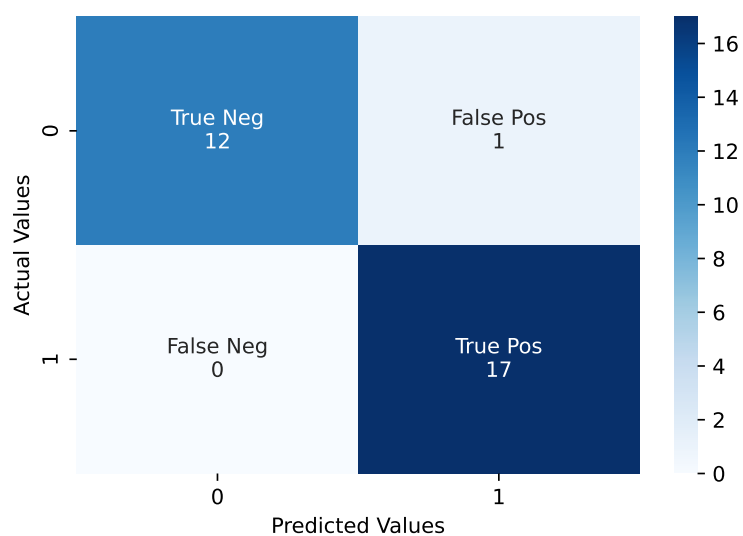
learning rate	0.1
batch size	128
epochy	200
optimizer	Adamax

V nejlepším případě (pro vhodný random seed = 19, který byl nastavený při zamíchání dat u jejich rozdělení v daných poměrech) dosáhla tato síť accuracy 96.6666 % a loss 14.3505 %.

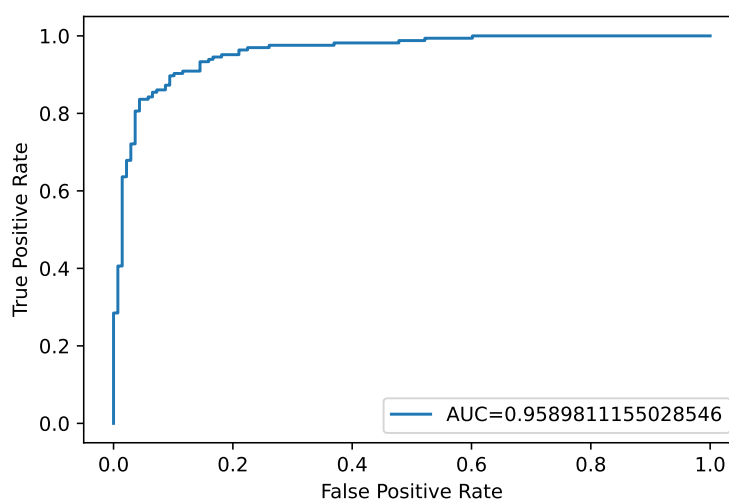
Na závěr byla provedena evaluace této sítě, a to na testovacích datech, a pak i na celém datasetu. Pro testovací data o velikosti 30 vzorků dosáhla síť učebnicových výsledků – ROC křivka měla ideální tvar (viz obrázek 5) a obsah pod touto křivkou byl roven jedné. Byla také napočtena matice záměn (viz obrázek 6) a další vyhodnocovací kritéria jako recall, percision, F1 score, specificity, atd. Výsledky vyhodnocení této sítě pro celkový dataset o velikosti 303 vzorků byly již realističtější, ale pořád příznivé (viz obrázek 7).



Obrázek 5: ROC křivka pro testovací data



Obrázek 6: Matice záměn pro testovací data



Obrázek 7: ROC křivka pro celý dataset

3 Závěr

V rámci projektu byla vyzkoušena práce se dvěma datasety, dále předzpracování dat, tvorba modelu neuronové sítě a také vyhodnocení jejích výsledků. Na MNIST datech bylo dosaženo 99.22 % accuracy pro celou trénovací množinu, 98.0 % accuracy pro množinu 10 000 dat a

99.8964 % pro binární klasifikaci tříd 8 a 6. V případě datasetu Heart byla po rozsáhlé analýze vybrána nejúspěšnější struktura sítě, která pro vybraný split dat (zamíchání vzorků s random seed = 19) dosáhla přesnosti klasifikace 96.6666 %.

Úloha splnila akademické účely, ale pro reálné nasazení sítě by bylo nutné připravené experimenty spustit a vyhodnotit na větším počtu dat.