

I-SUNS: Zadanie č.2

Viac vrstvový perceptrón II.

Vypracoval: Jakub Šíp

AIS ID: 91419

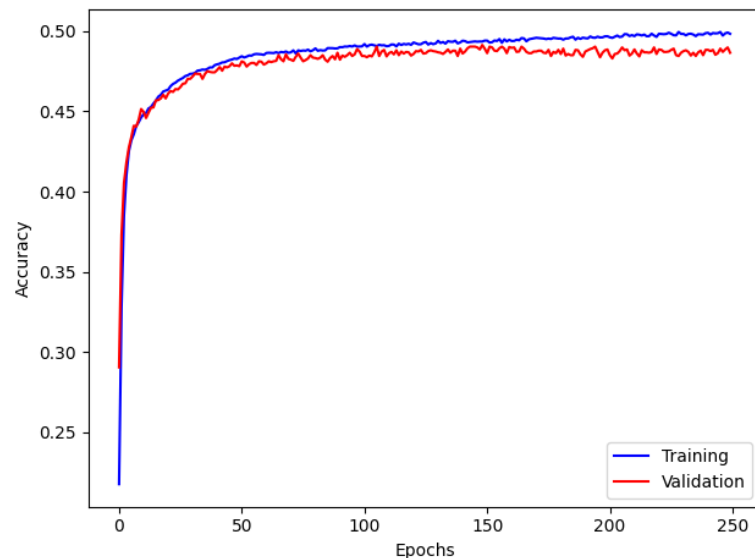
Na implementáciu tohto zadania som si zvolil programovací jazyk Python 3.8 a knižnicu sklearn a tensorflow. Pre predspracovanie dát som použil knižnicu pandas, ktorá uľahčuje prácu s dátovými štruktúrami.

Úloha 1:

- Z dát som odstránil pre nás nepodstatné stĺpce. Takže zostalo 11 stĺpcov a to ['danceability', 'energy', 'key', 'loudness', 'mode', 'speechiness', 'acousticness', 'instrumentalness', 'liveness', 'valence', 'tempo']
- Pomocou get_dummies() som pre potreby trénovania neurónovej siete dostal 6 dáta vo forme vhodnej na trénovanie a testovanie. 6 stĺpcov (pre každý žáner 1 stĺpec a hodnoty 0 a 1 pričom, 1 znamená, že je to práve tento žáner)
- Dáta normalizujem aj pomocou StandardScaler(), ktorý zachová pomery ale upraví hodnoty na čo najmenšie čísla. Ideálne by boli všetky v intervale <-1,1>
- Na testovanie používam dáta z „test.csv“ a na trénovanie z „train.csv“, z ktorých ešte vyberám validačné

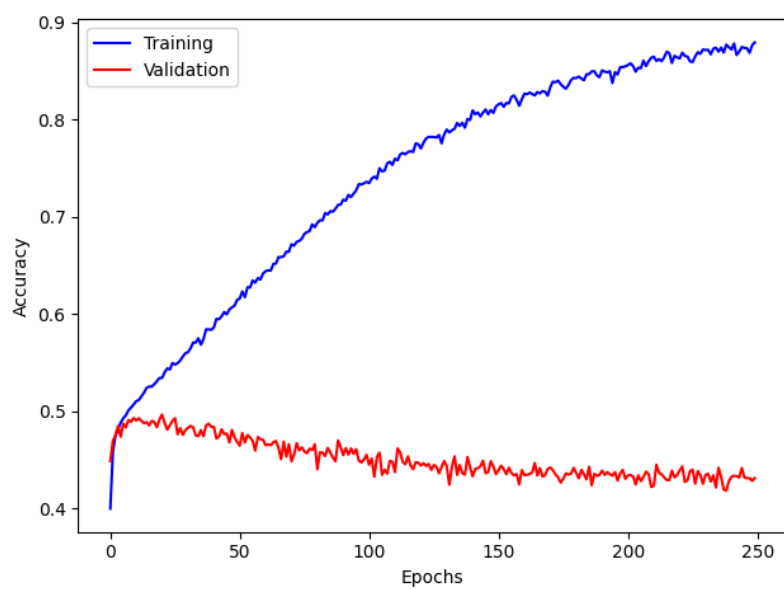
Úloha 2:

- Najvyššia dosiahnutá presnosť na validačných a testovacích dátach bola pri použití len 3 vrstvovej siete, kedy sa nedostavilo pretrénovanie. Počet neurónov v skrytých sieťach bol [20,20,6]. Testované to bolo na 250 epochách, kde výsledok sa dostavil už pri 70tej a zvyšok bola úroveň presnosti približne rovnaká s odchýlkou o +-1%.



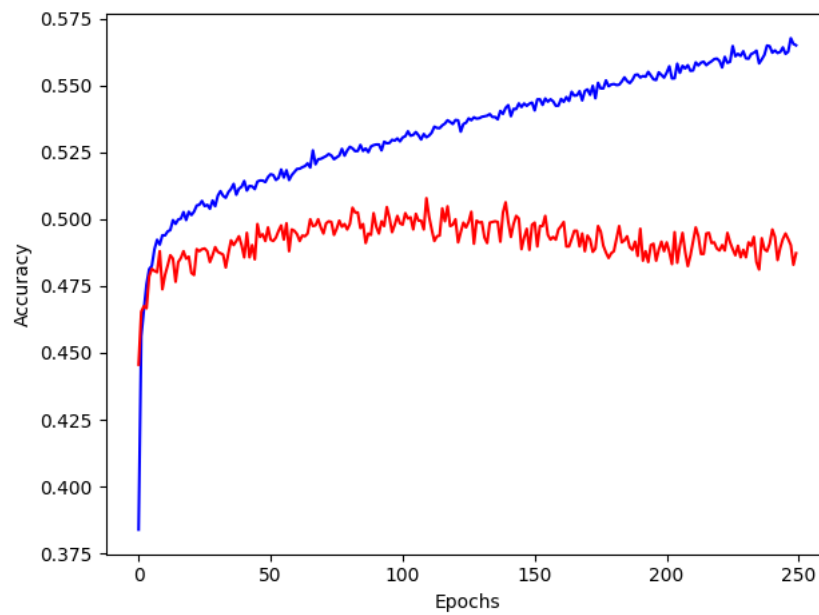
Úloha 3:

- Na preténovanie siete som do nej pridal viac vrstiev a do každej vrstvy aj viac neurónov.



- L1/L2 regularizácia:

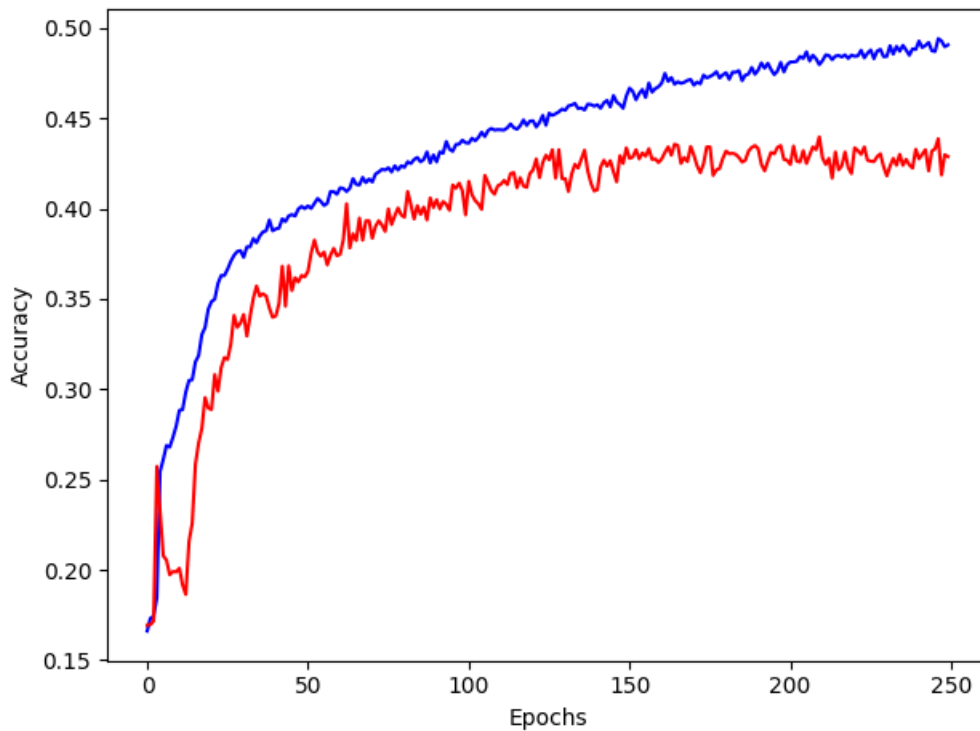
- Použil som zabudovanú funkcionálnosť v knižnici tensorflow. S parametrami na regularizáciu $L1 = L2 = 0.0001$. Po jej aplikovaní na každú vrstvu som dostal takéto výsledky.



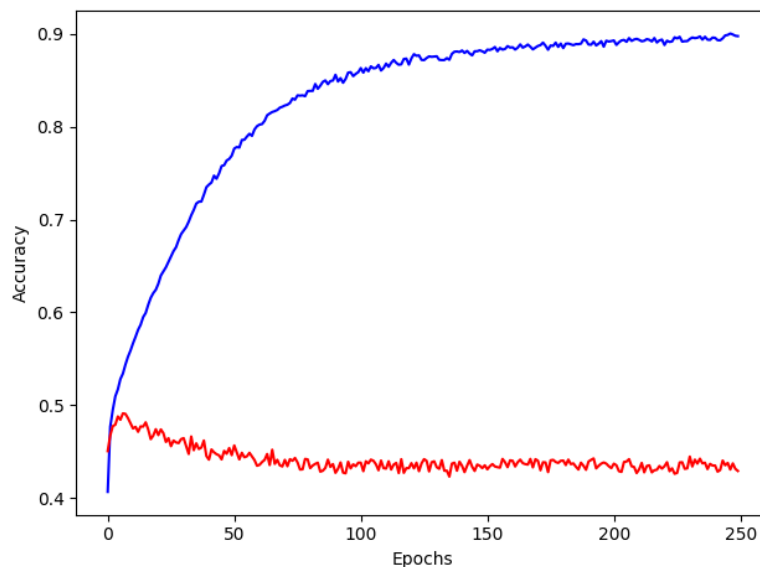
- Táto regularizácia sa preukázala zo všetkých ako najlepšia

- Dropout:

- Realizoval som ho medzi všetkými vrstvami s parametrom 0.6



- Ani zmenou parametra sa mi však nepodarilo dosiahnuť presnosť ako pri L1/L2 regularizácii.
- BatchNormalization:
 - Nastavoval som len parameter momentum a epsilon.



- Tento druh regularizácie sa však v mojom prípade vôbec neosvedčil

Úloha 4:

- Na riešenie som využil už pripravený SVM klasifikátor z knižnice sklearn.
- Jeho tréning je extrémne pomalé a v porovnaní s neurónovou sieťou nedosahuje ani takú presnosť.
- Najvyššie mnou dosiahnutá presnosť je 44%

-----SVM-----					
[[872 180 247 158 148 64]					
[134 640 189 296 146 144]					
[259 335 421 149 321 173]					
[164 255 90 874 65 172]					
[118 118 209 42 1039 159]					
[108 282 202 403 196 462]]					
		precision	recall	f1-score	support
	0	0.53	0.52	0.52	1669
	1	0.35	0.41	0.38	1549
	2	0.31	0.25	0.28	1658
	3	0.45	0.54	0.49	1620
	4	0.54	0.62	0.58	1685
	5	0.39	0.28	0.33	1653
accuracy				0.44	9834
macro avg		0.43	0.44	0.43	9834
weighted avg		0.43	0.44	0.43	9834

1	EDM
2	Latin
3	Pop
4	Rap
5	Rock
6	R&B