

Neurónové siete

Zadanie 1 - MNIST

MNIST dataset - EDA

Na stránke <http://yann.lecun.com/exdb/mnist/> sú dostupné dáta:

train-images-idx3-ubyte.gz: training 50k
train-labels-idx1-ubyte.gz: training 50k
t10k-images-idx3-ubyte.gz: test 10k
t10k-labels-idx1-ubyte.gz: test 10k

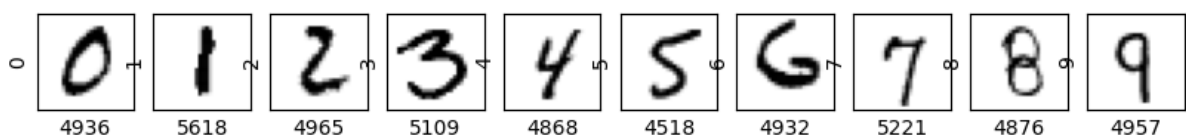
Rovnaké dáta sa nachádzajú v knižnici `keras.datasets.mnist`, ktorú sme po dohode použili. Je tak kvôli jednoduchšiemu načítaniu dát na cloudových službách.

“Train images” obsahuje 50 000 obrázkov, ktoré sme si následne rozdelili na 40 000 pre tréning a 10 000 pre validačnú množinu. Rozdelenie sme vykonali pomocou funkcie `sklearn.model_selection.train_test_split`, ktorá s parametrom “state” zaručuje rovnaké rozdelenie dát v rozdielnych zbehnutiach.

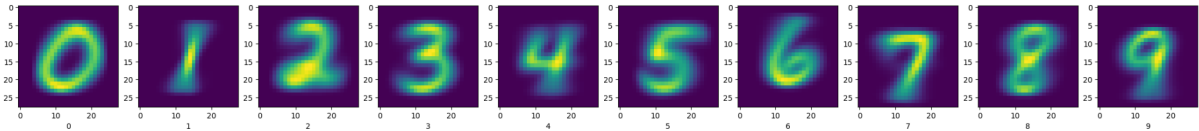
Načítané obrázky sú numpy polia o veľkosti 28x28. Výsledné dáta majú rozmery:

X_train: (50000, 28, 28)
Y_train: (50000,)
X_val: (10000, 28, 28)
Y_val: (10000,)
X_test: (10000, 28, 28)
Y_test: (10000,)

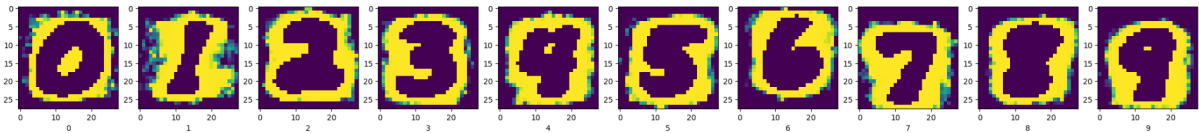
Vyobrazili sme si početnosť a vizuálnu reprezentáciu každej číslice.



Tiež sme si vyobrazili priemernú početnosť pixelov pre jednotlivé čísla:



Avšak zistili sme, že v datasete sa nachádzajú aj určité outlier hodnoty (žltým), ktoré sme sa pokúsili vyobraziť nasledovne:



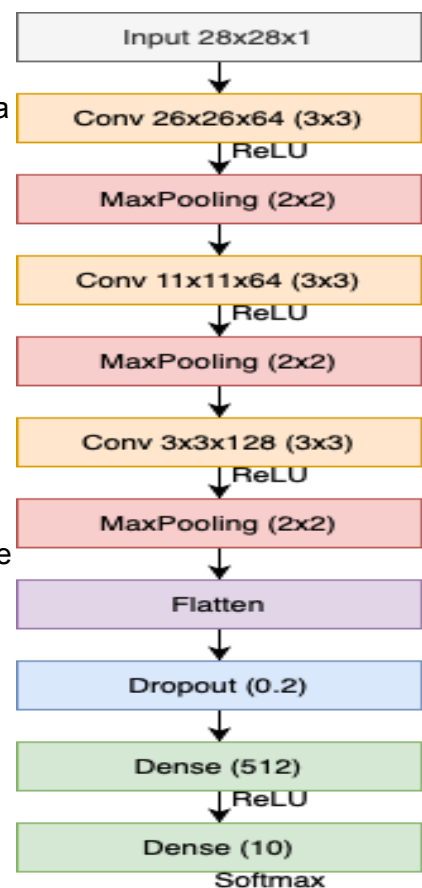
Architektúra siete

Pri tvorbe architektúry sme použili predošlé skúsenosti s neurónovými sieťami. Skúšali sme modifikovať architektúru a to tak, že sme skúšali meniť hodnotu dropout, meniť počty neurónov na Dense vrstve, pridať ďalšiu Dense vrstvu.

Z výsledkov tréningu vyšla najlepšie táto sieť, ktorú sme použili aj v prípade Tensorflow aj v prípade PyTorch.

V rámci našej architektúry sme použili viacero blokov, konkrétne Konvolučnú vrstvu, MaxPooling vrstvu, Flatten, Dropout, Dense.

Aktivačná funkcia, ktorá sa najčastejšie používa v rámci konvolučných neurónových sietí je ReLU, z toho dôvodu sme sa ju rozhodli použiť na všetky konvolučné vrstvy ako aj na Dense vrstvu. Na výstupnej vrstve máme Softmax z dôvodu toho, že jej výstupom sú v podstate percentá.



Priebehy trénovaní

Postupný vývoj sme zaznamenali nástrojom Wandb.

Odkaz pre report z Tensorflow:

https://wandb.ai/nsiete23-gaza-kuran/Nsiete_zadanie1_tensorflow/reports/Zadanie1_tensorflow--VmlldzozOTE1NDIx

Odkaz pre report z Pytorch:

https://wandb.ai/nsiete23-gaza-kuran/Nsiete_zadanie1_pytorch/reports/Zadanie1_pytorch--VmlldzozOTE1NTUx

Tensorflow

Prvý tréning obsahoval nasledovné nastavenia:

batch: 50

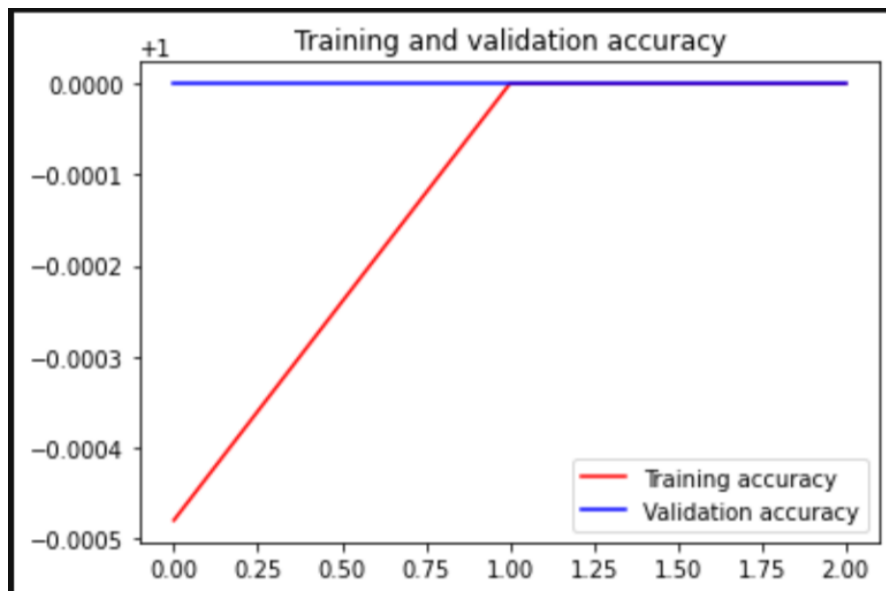
epoch: 3

learning rate: 0.001

dropout: 0.5

optimi: RMSProp

augmentácie: žiadne



V tomto bode sme ešte nemali pripojené Wandb preto je obrázok iba snímka z Jupyteru.

Model potreboval viacero úprav, pridanie zmeny hyperparametrov, odstránenie preučenia a zlepšenie generalizácie.

Sieť neobsahovala finálnu architektúru. Skúšali sme meniť počet filtrov v konvolučnej vrstve a táto zmena sa nám overila. Tiež sme modifikovali počet neurónov a počet Dense vrstiev avšak tieto zmeny sa nám neosvedčili. Tiež sme pridali L2 regularizáciu a augmentáciu trénovacej množiny. Po úpravách mal model nasledovné hyper parametre:

batch: 50

epoch: 25

learning rate: 0.0001

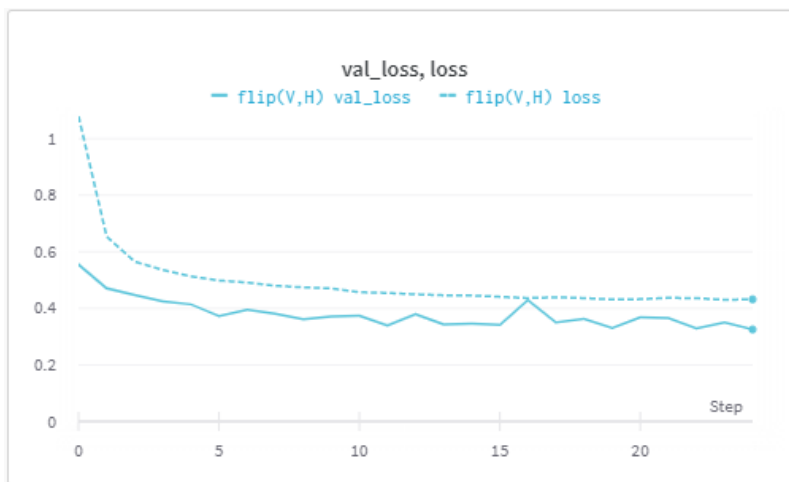
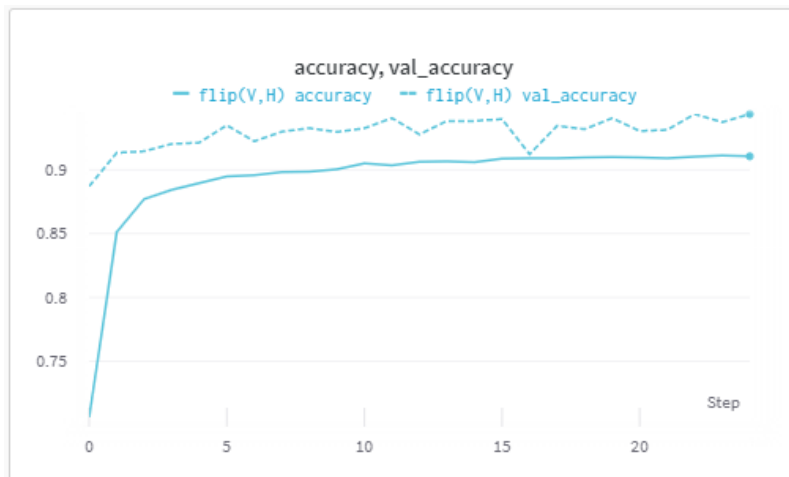
dropout: 0.5

optimi: RMSprop

augmentácie: horizontálny a vertikálny flip

pridanie regularizácie L2 na všetky Conv vrstvy 0.001

pridanie regularizácie L2 na Dense vrstvu 0.001



Priebeh učenia avšak stále nebol plynulý a preto sme riešili ďalšiu úpravu hyper parametrov. Uvedomili sme si, že horizontálne a vertikálne otáčanie obrázkov môže mať nežiadúce účinky, a preto sme upravili augmentáciu ako otáčania o určitý maximálny počet uhlov a o posun obrázka v osi X a Y. Tento posun a otáčanie sme ladili, zároveň sme ladili dropout a optimalizáciu. Výsledné hyper parametre pre model:

batch: 50

epoch: 25

learning rate: 0.0001

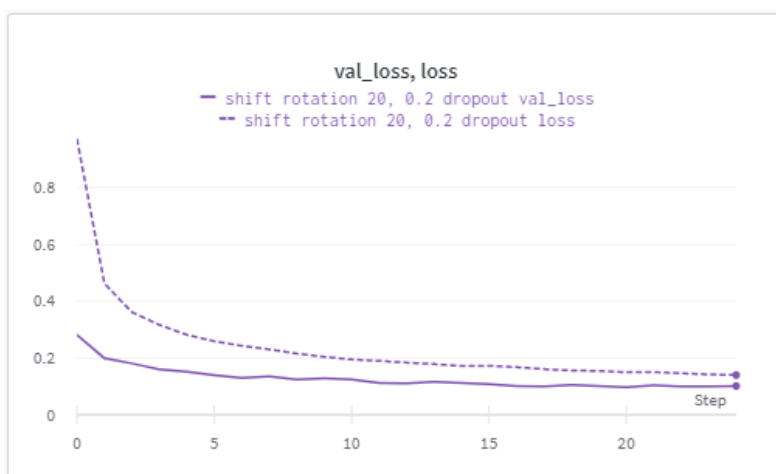
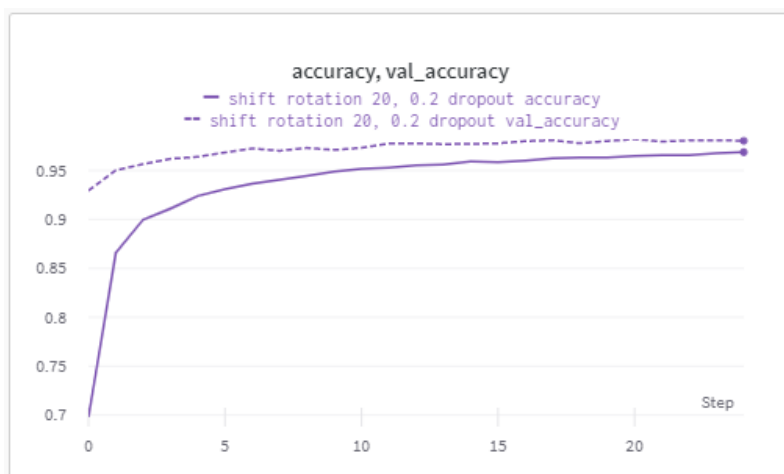
dropout: 0.2

optimi: ADAM

augmentácie: posunutie smer X a Y (10), rotacia(-20,+20)

pridanie regularizácie L2 na všetky Conv vrstvy 0.0001

pridanie regularizácie L2 na Dense vrstvu 0.0001



Výsledné riešenie má presnosť 96,8% na trénovacej množine, 98,04% na validačnej množine a 98,7% na testovacej množine. Zvyšné priebežné behy okrem uvedených sú dostupné vo Wandb report:

https://wandb.ai/nsiete23-gaza-kuran/Nsiete_zadanie1_tensorflow/reports/Zadanie1_tensorflow--VmlldzozOTE1NDIx

PyTorch

V rámci nášho prvého tréningu v PyTorch sme mali nastavené hyper parametre takto:

batch: 128

epoch: 25

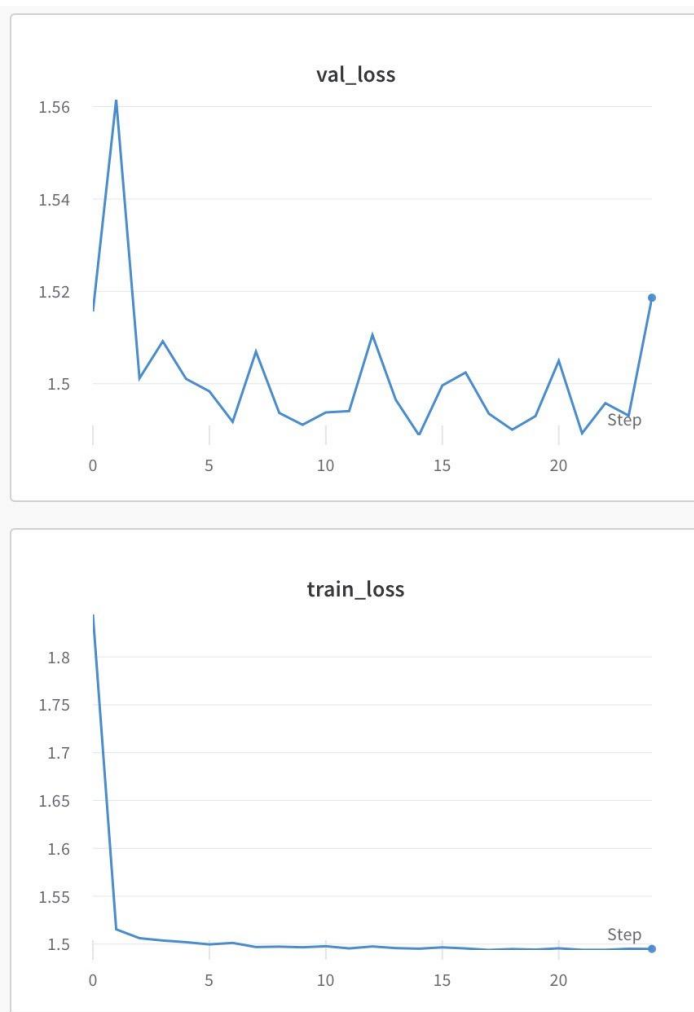
learning rate: 0.001

dropout: 0.1

optimi: Stochastic Gradient Descent

augmentácie: žiadne

Tu naša sieť dosahovala vysokú presnosť ale sledovali sme, že je tomu tak kvôli preučeniu.



Prvým cieľom bolo vyriešiť toto preučenie a to sa nám podarilo pri použití týchto hyper parametrov:

batch: 50

epoch: 25

learning rate: 0.00001

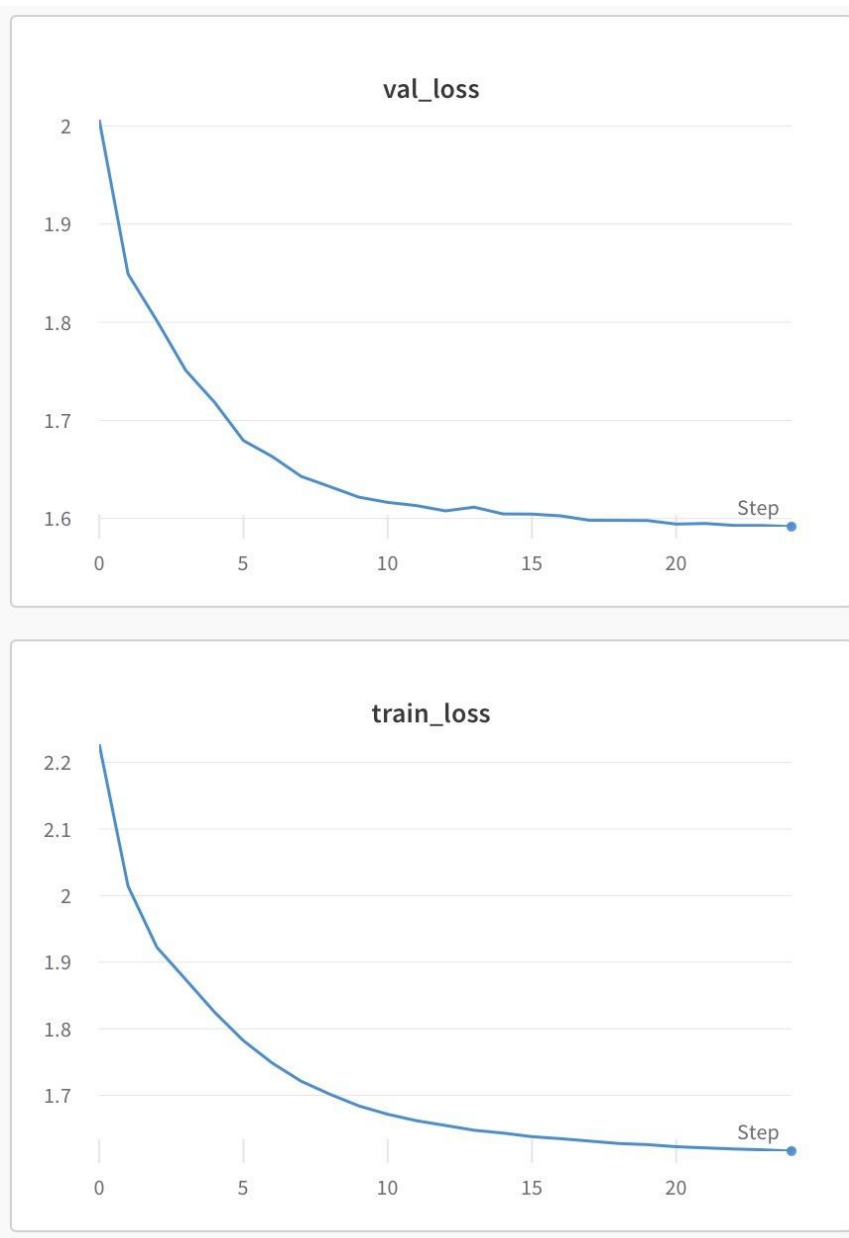
dropout: 0.2

optimi: RMSprop

augmentácie: horizontálny a vertikálny flip

pridanie regularizácie L2 na všetky Conv vrstvy 0.001

pridanie regularizácie L2 na Dense vrstvu 0.001



Akonáhle sme mali vyriešený problém preučenia hľadali sme nastavenie parametrov také, ktoré nám dali najvyššiu možnú presnosť. To sa nám podarilo pri použití týchto hyper parametrov:

batch: 50

epoch: 25

learning rate: 0.00001

dropout: 0.2

optimi: ADAM

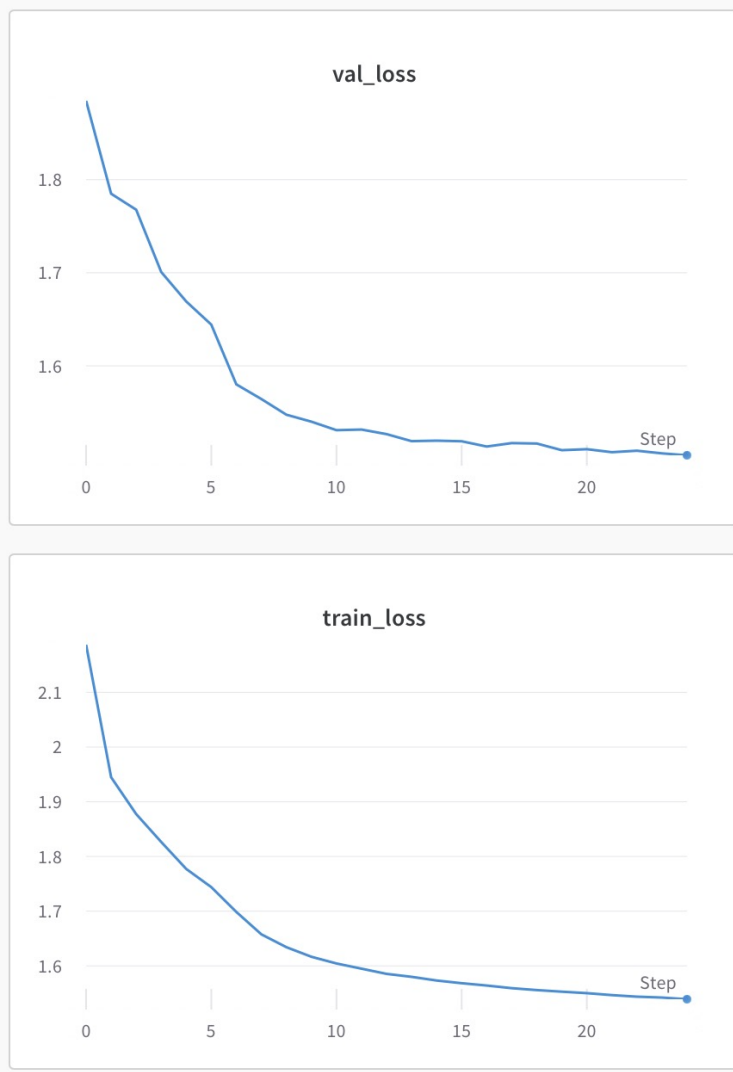
augmentácie: posunutie smer X a Y (10), rotacia(-20,+20)

pridanie regularizácie L2 na všetky Conv vrstvy 0.0001

pridanie regularizácie L2 na Dense vrstvu 0.0001

Najväčšia dosiahnutá presnosť nášho modelu v pytorch bola na trenovacej množine: 92.93% na validacnej množine: 95.72% na testovacej množine: 96.21%. Zvyšné behy je možné vidieť v rámci reportu z Wandb:

https://wandb.ai/nsiete23-gaza-kuran/Nsiete_zadanie1_pytorch/reports/Zadanie1_pytorch--VmlldzozOTE1NTUx



Changelog:

Počas priebežného prezentovania boli navrhnuté tieto požiadavky na zmeny, ktoré sme implementovali do finálneho prezentovania:

- **opraviť overfit** - vyššie uvedené pokusy, ktoré sú buď prvé alebo druhé v poradí pre Tensorflow alebo pre Pytorch obsahovali overfit alebo generalizácia bola nedostatočná. V nasledujúcom pokuse (treťom uvedenom pre Tensorflow/Pytorch) je overfit odstránený.
- **hyperparametre: batch size, epochy, learning rate manualne nastaviť a otestovať** - druhý/tretí pokus obsahujú manuálne nastavované hyper parametre, ktoré sú zároveň obsiahnuté vo Wandb logu pre jednotlivý beh.
- **augmentácia** - prvotná pridaná augmentácia bola vertikálne a horizontálne otáčanie obrázkov, avšak pri takomto otáčaní môžu vzniknúť nechcené obrázky (napr. 6 a 9), preto sme sa rozhodli pre augmentáciu pomocou posúvania a rotácie.

