

## Zadanie č.4.

Pozrime sa do vnútra neurónu. V predchádzajúcich úlohách ste si vytvorili súbory s tréningovými, testovacími a validačnými dátami. Použite tieto dáta a postupne sa zamerajte na jednotlivé stavebné časti perceptrónu. Odporúčame sa odraziť od príkladov na Tensorflow a DLib uvedených nižšie.

### Úloha č. 1

Natrénujte MLP s jednou skrytou vrstvou, v ktorej bude aspoň 1024 neurónov. Ako aktivačnú funkciu použite ReLU a trénujte pomocou **stochastického** solvera. Využite pri tréningu vaše validačné dáta.

- Výstup: úspešnosť systému na testovacích dátach + priebeh tréningovania zobrazený úspešnosťou dosiahnutou validačných dát.

### Úloha č. 2

Využite vo Vašom systéme L2 regularizáciu. Skúste nájsť hodnotu, ktorá vhodne ovplyvní tréningovanie.

- Výstup: graf úspešnosti vašej siete na testovacích dátach v závislosti od meniacej sa lambdy.

### Úloha č. 3

Skúmajte vplyv počtu vstupných batchov na priebeh tréningovania - dá sa na sieti pozorovať pretrénovanie?

- Výstup: graf úspešnosti vašej siete na testovacích/validačných dátach a tréningových dátach v závislosti od počtu vstupných batchov.

### Úloha č. 4

Pridajte do siete z úlohy 3. dropout aby ste sa vyhli pretrénovaniu. Pozor: dropout **nesmie** byť použitý pri testovaní.

- Výstup: Graf úspešnosti siete v závislosti od veľkosti dropoutu.

Linky:

[https://github.com/tensorflow/tensorflow/blob/master/tensorflow/examples/udacity/2\\_fullyconnected.ipynb](https://github.com/tensorflow/tensorflow/blob/master/tensorflow/examples/udacity/2_fullyconnected.ipynb)

[http://dlib.net/dnn\\_metric\\_learning\\_ex.cpp.html](http://dlib.net/dnn_metric_learning_ex.cpp.html)