

Implementace: Python 3.12

Použité knihovny/balíčky budou specifikovány v souboru `requirements.txt` (starší jednoduchý způsob jak specifikovat závislosti k instalaci přes `pip`).

Povinná je existence dvojice skriptů:

**train.py**

- natrénuje a uloží model **umělé neuronové sítě** pro binární klasifikátor v aktuálním pracovním adresáři (jako soubor nebo celý adresář)
- cesta ke vstupnímu CSV souboru je předána na prvním parametru příkazové řádky
- formát vstupního CSV souboru odpovídá souboru `p1_train.csv` v e-learningu

**inference.py**

- použije model uložený skriptem `train.py` k inferenci dat ve vstupním CSV souboru (cesta předána na prvním parametru příkazového řádku) a výsledek klasifikace vypíše na standardní výstup (*stdio*) pro každý řádek.
  - Pokud soubor neobsahuje sloupec *class*:  
`class`  
`QqQqQ`  
`QqQqQ`  
`MmMmM`  
`(...)`
  - Pokud soubor obsahuje sloupec *class*:  
`class, class_expected`  
`QqQqQ, QqQqQ`  
`MmMmM, QqQqQ`  
`QqQqQ, MmMmM`  
`(...)`
- Na základě výstupů bude zhodnocena úspěšnost Vašeho modelu.

Existence jiných zdrojových a konfiguračních souborů je na Vás.

Pro uložení natrénovaného modelu použijte dobře definované formáty jako `numpy.savez`, `HDF5`, `JSON`, `XML`, `SQLite`, `protobuf`, `tflite` (nepoužívejte: ~~`pickle`~~, ~~`joblib.dump`~~, ~~`cloudpickle`~~).

Frameworky (Keras, JAX, TensorFlow, PyTorch, ...) mají metody pro uložení modelů. Případně můžete použít standard `ONNX`.

Nedoporučuji Scikit-learn (a s ním narazíte tutoriál se zakázanými ~~`pickle`~~/~~`joblib`~~)

Příklad spuštění ve složce Vašeho projektu, které se provede automaticky (cesty a názvy souborů se mohou lišit):

```
python3.12 -m venv venv
venv/bin/activate                # na Windows: .\venv\Scripts\activate
pip install -r requirements.txt
python3 train.py /home/ucitel/VSC_2025/projekt1/trainset.csv
python3 inference.py /home/ucitel/VSC_2025/projekt1/p1_test_student.csv
python3 inference.py /home/ucitel/VSC_2025/projekt1/p1_test_teacher.csv
```