

Sieci neuronowe 2020 – Lista 1 **Uzupełniona**

Zadanie 1. Wykonaj kod tworzenia, uczenia i testowania sieci gęstej dla zbioru MNIST [1,2].

Zadanie 2. Korzystając z biblioteki Keras (<https://keras.io>) lub innej, zaimplementuj sieć neuronową dla problemu klasyfikacji recenzji filmowych z bazy IMDB (<https://keras.io/api/datasets/imdb/>). Wykorzystaj parametry sieci z Rozdziału 3.5 [1] lub [2]. Czy sprawność sieci zmieni się jeżeli:

- będzie tylko jedna warstwa gęsta?
- będą trzy warstwy gęste?
- liczba jednostek w warstwie/warstwach zmniejszy się?
- liczba jednostek w warstwie/warstwach zwiększy się?
- zamiast entropii krzyżowej wykorzystamy inną funkcję straty, np. błąd średniokwadratowy?
- w ostatniej warstwie funkcją aktywacji będzie tangens hiperboliczny?

Na podstawie powyższych eksperymentów wybierz najlepsze wartości hiperparametrów sieci (liczba warstw, liczba jednostek w warstwie, funkcje aktywacji, rozmiar wsadu, liczba epok, itd.) Dla najlepszych wartości hiperparametrów wytrenuj sieć na całym zbiorze treningowym (25000 próbek). Sprawdź trafność klasyfikacji (accuracy) tak wytrenowanej sieci na zbiorze testowym.

Zadanie 3. Czy dla sieci neuronowej i zbioru „reuters” (<https://keras.io/api/datasets/reuters/>) sprawność zmieni się w stosunku do Rozdziału 3.6 [1] lub [2] jeżeli:

- będzie tylko jedna warstwa gęsta?
- będą trzy warstwy gęste?
- liczba jednostek w warstwie/warstwach zmniejszy się?
- liczba jednostek w warstwie/warstwach zwiększy się?
- rozmiar wsadu zmniejszy się?
- rozmiar wsadu zwiększy się?

Dla najlepszych wartości hiperparametrów znalezionych w trakcie powyższych eksperymentów, wytrenuj sieć na całym zbiorze treningowym. Sprawdź trafność klasyfikacji (accuracy) tak wytrenowanej sieci na zbiorze testowym.

Zadanie 4. Eksperymentalnie dobierz najlepsze wartości hiperparametrów sieci neuronowej dla zadania regresji dla danych ze zbioru „Boston Housing” (https://keras.io/api/datasets/boston_housing/). Jaki jest średni błąd bezwzględny dla zbioru testowego? Czy zastosowanie K -składowej walidacji krzyżowej poprawia proces doboru wartości hiperparametrów?

Zadanie 5. Pobierz zbiór *Iris* (<https://www.tensorflow.org/datasets/catalog/iris?hl=en>) z modułu *TensorFlow Datasets*. Dobierz najlepsze wartości hiperparametrów sieci na podstawie:

- [na ocenę 3.0]: K -składowej walidacji krzyżowej.
- [na wyższą ocenę]: K -składowej walidacji krzyżowej ze zbiorem walidacyjnym i testowym.

K -składowa walidacja krzyżowa. Najpierw ze zbioru danych wydzielany jest zbiór testowy. Następnie pozostałe dane są dzielone na K składowych. Każda z tych składowych jest następnie zbiorem walidacyjnym, a pozostałe $K-1$ składowych składa się na zbiór treningowy. Walidacja następuje na podstawie średnich danych walidacyjnych z K treningów. Na podstawie walidacji wybieramy najlepsze wartości hiperparametrów, a następnie uczymy sieć na całym zbiorze danych oprócz części testowej. Ocena wyuczonej sieci następuje na zbiorze testowym.

K -składowa walidacja krzyżowa ze zbiorem walidacyjnym i testowym. Dane są dzielone na K składowych. Każda z tych składowych jest następnie zbiorem testowym, a z pozostałych $K-1$ składowych każda staje się zbiorem walidacyjnym, a pozostałe $K-2$ składowych składa się na zbiór treningowy. Walidacja następuje na podstawie średnich danych walidacyjnych z $K \cdot (K-1)$ treningów. Na podstawie walidacji wybieramy najlepsze wartości hiperparametrów, a następnie uczymy sieć K -krotnie – za każdym razem na wszystkich $K-1$ składowych, które nie są w danym przebiegu zbiorem testowym. Ocena wyuczonej sieci jest średnim wynikiem dla K zbiorów testowych.

W sprawozdaniu zamieść i opisz:

- wykorzystany kod,
- sposób doboru hiperparametrów,
- wykresy funkcji straty i trafności dla walidacji,
- wyniki uzyskane dla najlepszej kombinacji parametrów oraz rysunek finalnej sieci wykonany przy wykorzystaniu `keras.utils.plot_model`,
- wnioski.

Zadanie 6. Znajdź zbiór danych odpowiedni do zadania **klasyfikacji jednoetykietowej (dyskretnej)**. Zbiór może pochodzić z Kaggle (<https://www.kaggle.com/datasets>) lub UCI Machine Learning Repository (<https://archive.ics.uci.edu/ml/datasets.php>) lub z innego źródła (po uzgodnieniu z prowadzącym). Zastanów się jak wykorzystać sieć neuronową do rozwiązania tego zadania klasyfikacji. Przygotuj się na omówienie zbioru danych i propozycji rozwiązania problemu klasyfikacji z prowadzącym podczas zajęć 10.01.2021. **Zaproponuj strukturę sieci (rozważ także wykorzystanie regularyzacji i porzucania) a następnie eksperymentalnie dobierz wartości hiperparametrów sieci maksymalizujące efektywność sieci na zbiorze testowym. W sprawozdaniu zamieść i opisz:**

- wykorzystany kod,
- sposób doboru hiperparametrów,
- wykresy funkcji straty i trafności dla walidacji,
- wyniki uzyskane dla najlepszej kombinacji parametrów oraz rysunek finalnej sieci wykonany przy wykorzystaniu `keras.utils.plot_model`,
- wnioski.

Zadanie 7. Znajdź zbiór danych odpowiedni do zadania **regresji**. Zbiór może pochodzić z Kaggle (<https://www.kaggle.com/datasets>) lub UCI Machine Learning Repository (<https://archive.ics.uci.edu/ml/datasets.php>) lub z innego źródła (po uzgodnieniu z prowadzącym). Zastanów się jak wykorzystać sieć neuronową do rozwiązania tego zadania regresji. Przygotuj się na omówienie zbioru danych i propozycji rozwiązania problemu regresji z prowadzącym podczas zajęć 10.01.2021. **Zaproponuj strukturę sieci (rozważ także wykorzystanie regularyzacji i porzucania) a następnie eksperymentalnie dobierz wartości hiperparametrów sieci maksymalizujące efektywność sieci na zbiorze testowym. W sprawozdaniu zamieść i opisz:**

- wykorzystany kod,
- sposób doboru hiperparametrów,
- wykresy funkcji straty i trafności dla walidacji,
- wyniki uzyskane dla najlepszej kombinacji parametrów oraz rysunek finalnej sieci wykonany przy wykorzystaniu `keras.utils.plot_model`,
- wnioski.

Literatura:

- [1] F. Chollet, Deep Learning. Praca z językiem Python i biblioteką Keras, Helion, 2019.
- [2] F. Chollet, J. J. Allaire, Deep Learning. Praca z językiem R i biblioteką Keras, Helion, 2019.