

WSI - ćwiczenie 5.

Sztuczne sieci neuronowe

11 grudnia 2024

1 Sprawy organizacyjne

1. Ćwiczenie realizowane jest w zespołach dwuosobowych.
2. Ćwiczenie wykonywane jest w języku Python.
3. Ćwiczenie powinno zostać oddane najpóźniej przed terminem 12. zajęć. Ćwiczenia należy zademonstrować prowadzącemu działanie kodu oraz utworzyć pull request (z kodem oraz raportem) który prowadzący będzie mógł komentować.
4. Raport powinien być w postaci pliku .pdf, .html albo być częścią noteboka jupyterowego. Powinien zawierać opis eksperymentów, uzyskane wyniki wraz z komentarzem oraz wnioski.
5. Na ocenę wpływa poprawność oraz jakość kodu i raportu.
6. Można korzystać z pakietów do obliczeń numerycznych, takich jak *numpy*
7. Implementacja algorytmu powinna być ogólna. W szczególności: powinno być możliwość zastosowania rozwiązania do dowolnego zbioru danych o wartościach rzeczywistych, liczba warstw i neuronów w każdej warstwie powinna być parametryzowalna, powinna być możliwość wyboru różnych funkcji straty oraz aktywacji.
8. Można skorzystać z pakietu *scikit-learn* w celu załadowania zbioru danych oraz jego podziału na części.

2 Ćwiczenie

Celem ćwiczenia jest implementacja perceptronu wielowarstwowego oraz wybranego algorytmu optymalizacji gradientowej z algorytmem propagacji wstecznej.

Następnie należy wytrenować perceptron wielowarstwowy do klasyfikacji zbioru danych MNIST (<http://yann.lecun.com/exdb/mnist/>). Zbiór MNIST dostępny jest w pakiecie *scikit-learn* (`sklearn.datasets.load_digits()`).

3 Uwagi

1. Zbiór MNIST jest zbiorem obrazków - należy je najpierw „spłaszczyć” do postaci wektorów.
2. Polecam najpierw sprawdzić poprawność implementacji na mniejszych, testowych danych (np. funkcja xor dla 3 neuronów w 1 warstwie ukrytej) - obliczenia na zbiorze MNIST mogą być czasochłonne i trudne do ręcznego przeanalizowania.
3. Wymagany jest podział danych na zbiór trenujący, walidacyjny i testowy. Można skorzystać w tym celu z gotowych funkcji.