
Laboratorium

Sieci neuronowe – ciąg dalszy

Zadanie 3 (z poprzednich zajęć)

Dokonaj klasyfikacji (diagnoza cukrzycy) za pomocą prostej jednokierunkowej sieci neuronowej, czyli wielowarstwowego perceptronu (Multilayer Perceptron, MLP).

Pomocne linki:

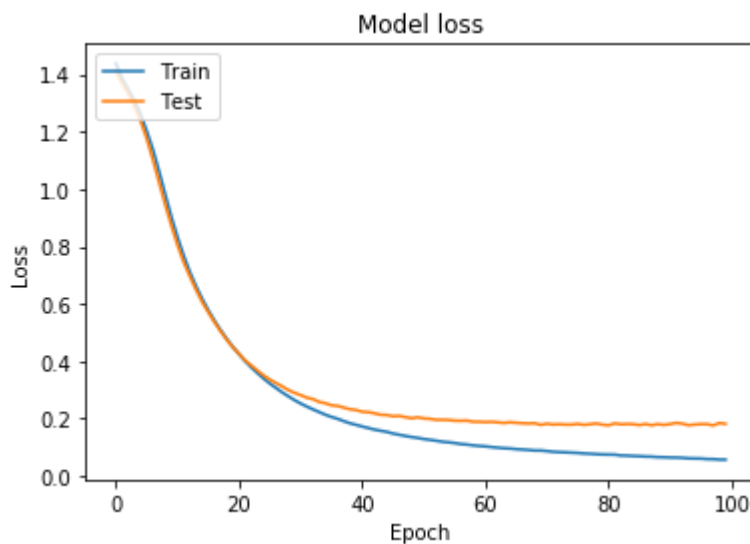
- https://scikit-learn.org/stable/modules/generated/sklearn.neural_network.MLPClassifier.html (dokumentacja i przykłady)
- <https://www.pluralsight.com/guides/machine-learning-neural-networks-scikit-learn> (samouczek)
- <https://analyticsindiamag.com/a-beginners-guide-to-scikit-learns-mlpclassifier/> (samouczek)

- a) Podziel zbiór danych na testowy (30%) i treningowy (70%).
- b) Zbuduj model sieci o dwóch warstwach ukrytych – pierwsza ma 6 neuronów, a druga 3 neurony, z funkcją aktywacji ReLU.
- c) Wytrenuj model na zbiorze treningowym na maksymalnie 500 iteracjach.
- d) Dokonaj ewaluacji na zbiorze testowym: podaj dokładność i macierz błędów.

Zadanie 4 (rozszerzające poprzednie zajęcia)

Wykonaj zadanie 3 wykorzystując inną paczkę: keras. Jest to bardziej zaawansowana biblioteka dedykowana sieciom neuronowym.

- a) Wykonaj kroki b-d z poprzedniego zadania.
- b) Nanieś na wykres krzywą błędów (loss curve) dla zbioru treningowego i zbioru walidującego (u nas zbiorem walidującym może być zbiór testowy). Wykres powinien wyglądać mniej więcej tak:



- c) Czy trenowanie powinno się przerwać w pewnym momencie, by uniknąć przeuczenia? Jak odczytać to z wykresu? A może mamy do czynienia z niedouczeniem? Podpowiedzi:
- Wykład
 - <https://machinelearningmastery.com/learning-curves-for-diagnosing-machine-learning-model-performance/>
 - <https://rstudio-conf-2020.github.io/dl-keras-tf/notebooks/learning-curve-diagnostics.nb.html>
- d) Przetestuj jak sieć będzie uczyła się z różnymi optyimizerami (adam, itd.) i różnymi funkcjami aktywacji (ReLU, itp.). Wyświetl i porównaj wykresy krzywych uczenia się dla różnych konfiguracji.
- e) Dodatkowo, w miarę możliwości, zwizualizuj sieć neuronową w formie grafu, obrazka. Jeśli się da to najlepiej z uwzględnieniem poszczególnych neuronów, krawędzi i wag. Być może przydatne będą linki:
- <https://datascience.stackexchange.com/questions/12851/how-do-you-visualize-neural-network-architectures>
 - <https://towardsdatascience.com/visualizing-artificial-neural-networks-anns-with-just-one-line-of-code-b4233607209e>