

Wprowadzenie do sztucznej inteligencji - ćwiczenie 5

Zadanie wykonywane w parach

Proszę zaimplementować perceptron dwuwarstwowy i nauczyć go reprezentować funkcję $J : [-5, 5] \rightarrow \mathbb{R}$, daną wzorem: $J(x) = \sin(x \cdot \sqrt{p[0]+1}) + \cos(x \cdot \sqrt{p[1]+1})$, gdzie $p[0]$ i $p[1]$ to najmłodsze cyfry numerów indeksów wykonawców.

W sprawozdaniu powinny znaleźć się wykresy funkcji aproksymowanej i jej aproksymacji. Powinny również znaleźć się wskaźniki jakości aproksymacji. Jak liczba neuronów w warstwie ukrytej wpływa na jakość aproksymacji?

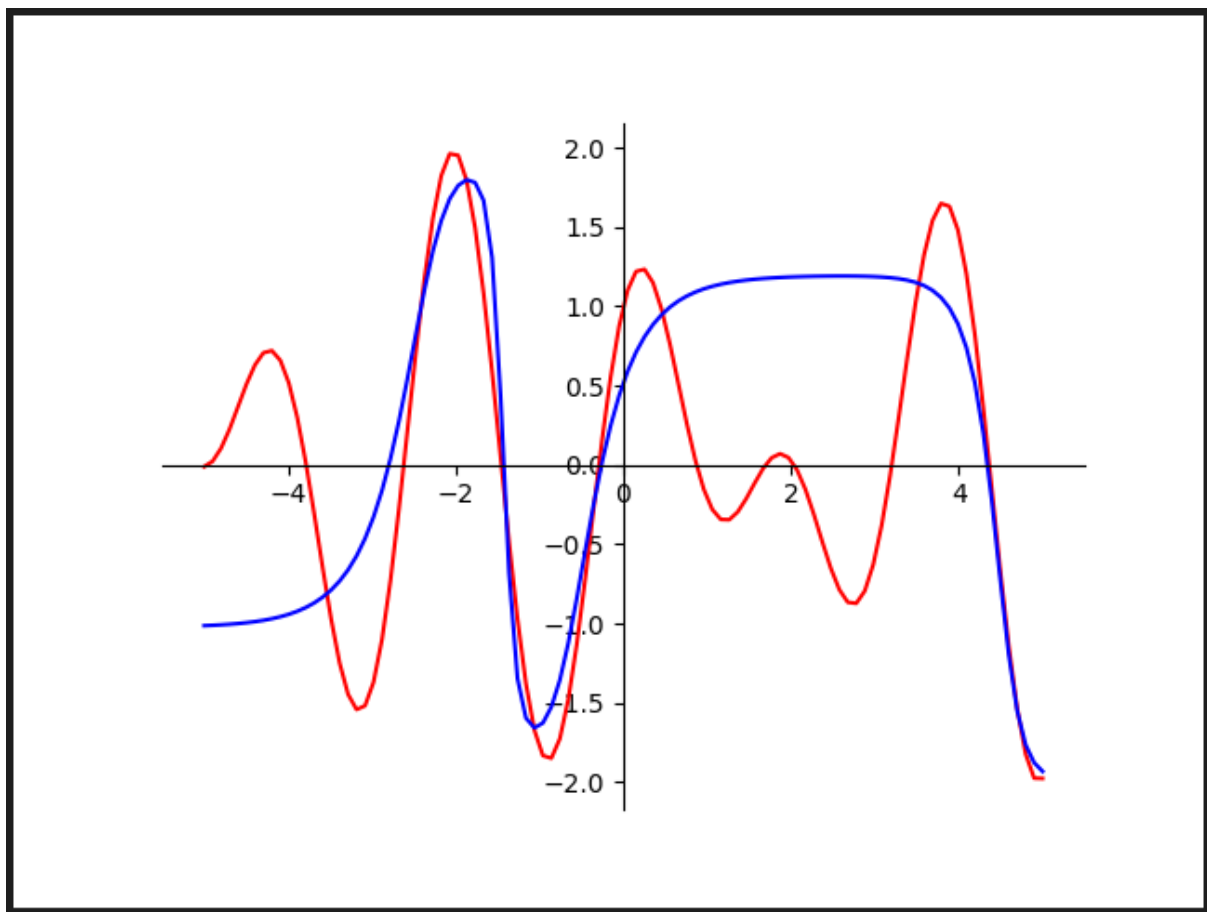
Przygotowałem dla Państwa [kod](#), który powinien ułatwić wykonanie zadania. Zamiast mojego kodu, osoby chętne mogą napisać własny kod pomocniczy. Nie można używać kodu z Internetu, czy bardziej ogólnie, kodu, którego nie jest się autorem.

Pierwsze litery indeksów = [4, 9]

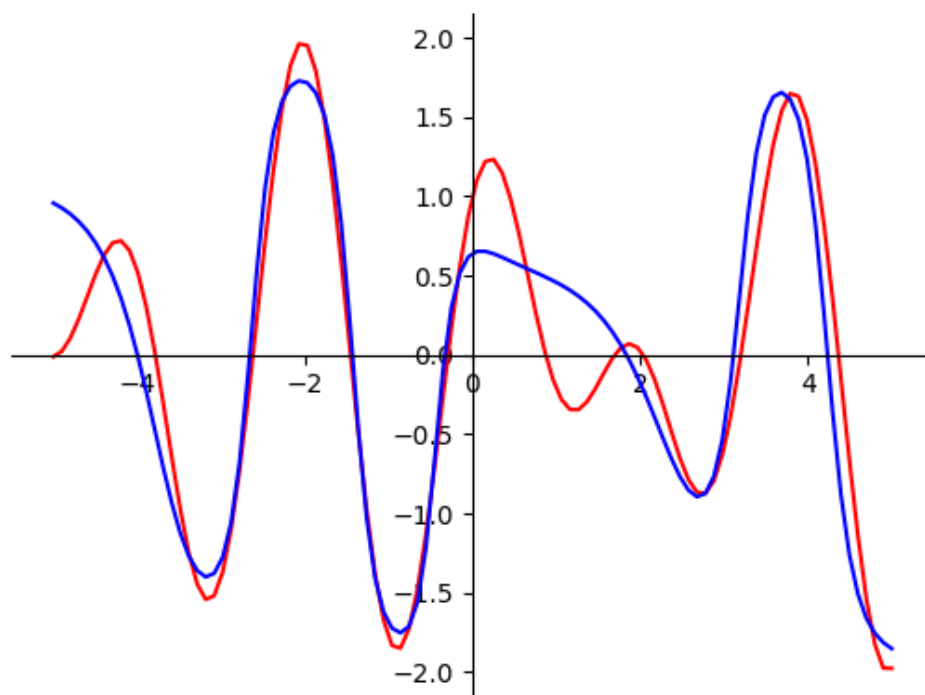
Wyniki:

czzerwony - wykres funkcji aproksymowanej

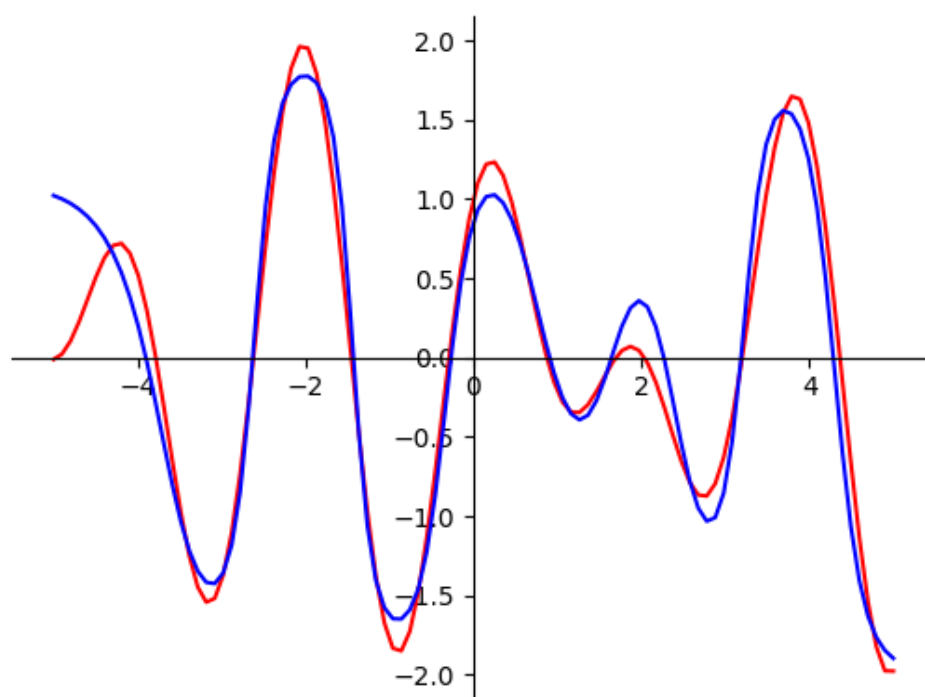
niebieski - wykres aproksymacji



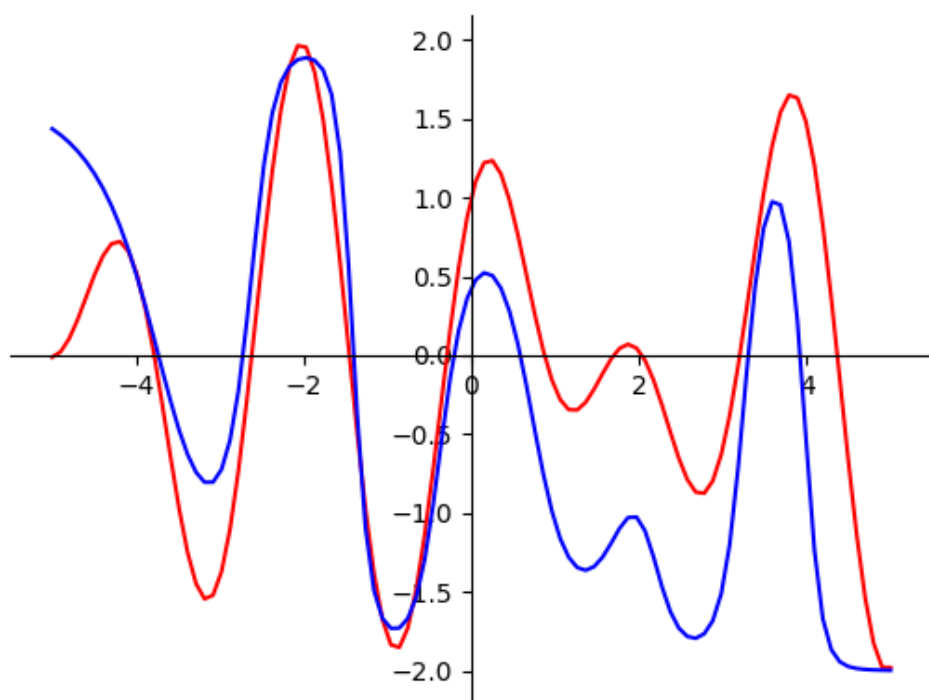
liczba neuronów: 3 | strata: 0.9300381993



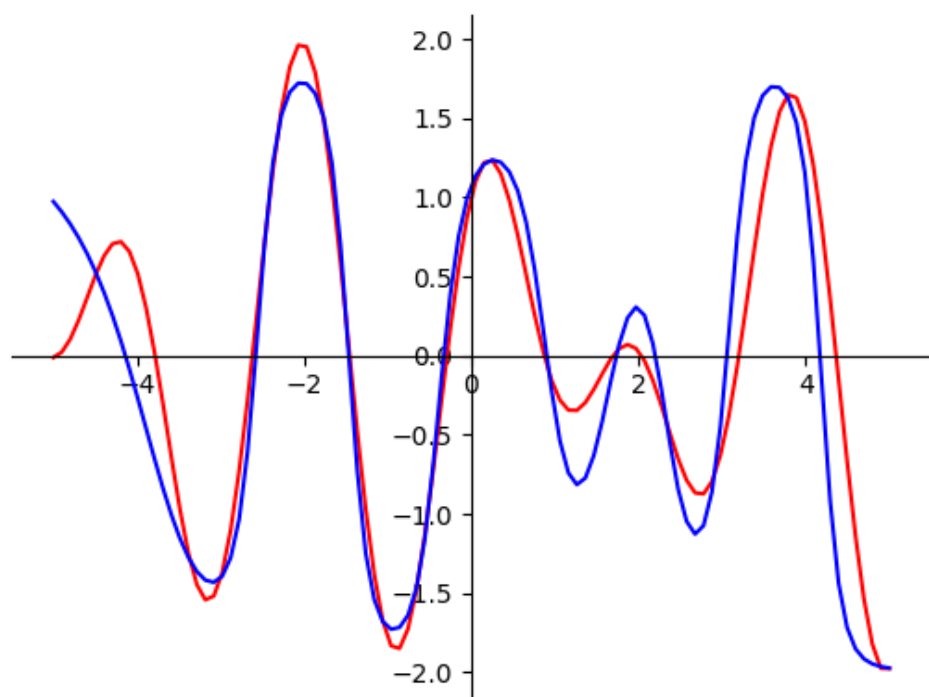
liczba neuronów: 7 | strata: 0.1327877991



liczba neuronów 12 | strata: 0.0706741249



liczba neuronów 23 | strata: 0.7300617799



liczba neuronów 30 | strata: 0.181275585

Wnioski:

Zbyt mała ilość neuronów w warstwie ukrytej powoduje, że sieć jest mało skuteczna, jednak od pewnego momentu zwiększanie ilości neuronów nie przynosi większych zysków, a wręcz może przynieść odwrotne skutki do oczekiwanych. U nas najlepszą sieć udało się uzyskać dla 12 neuronów, innym razem dla 19, a jeszcze innym dla 15 (modyfikowaliśmy ilość iteracji na zbiorze treningowym). Można więc zauważyć, że istnieje tutaj element losowy, który sprawia, że ciężko przewidzieć jak dana sieć będzie się dokładnie zachowywała.