wsi Perceptron Wielowarstowy

Tomasz Truszkowski, Piotr Lenczewski

December 2023

1 Perceptron Wielowarstowy

Perceptron wielowarstwowy, zdefiniowany w pliku multiLayerPerceptron.py, jest siecia neuronowa przeznaczona do aproksymacji funkcji. Został zaimplementowany jako klasa MultiLayerPerceptron, która wykorzystuje nastepujace elementy:

1.1 Inicjalizacja

Klasa MultiLayerPerceptron jest inicjalizowana z parametrami określajacymi rozmiar warstwy wejściowej, architekture warstw ukrytych (czyli liczbe neuronów w każdej warstwie) oraz rozmiar warstwy wyjściowej. Używa stałego ziarna losowości (seed), aby wyniki były powtarzalne, i inicjalizuje wagi oraz biasy (obciażenia) w sposób losowy.

1.2 Funkcja aktywacji

Funkcja sigmoidalna jest popularna funkcja aktywacji, która przypisuje każdej sumie ważonej wartość wyjściowa pomiedzy 0 a 1. Jest to przydatne w przypadkach, gdy potrzebujemy określić prawdopodobieństwa. Wyglad tej funkcji oraz jej pochodna sa matematycznie określone nastepujacymi wzorami:

1.2.1 Funkcja sigmoidalna

$\sigma(x) = \frac{1}{1+e^{-x}}$ gdzie:

- $\sigma(x)$ to wartość funkcji sigmoidalnej dla danego x,
- e to podstawa logarytmu naturalnego,
- x to suma ważona wejść do neuronu.

1.2.2 Pochodna funkcji sigmoidalnej

$\frac{d\sigma}{dx}(x) = \sigma(x)(1 - \sigma(x))$

1.3 Uczenie

Perceptron może być trenowany przy użyciu dwóch metod: metody gradientowej lub algorytmu ewolucyjnego. W zależności od wybranej metody, wywoływana jest odpowiednia wewnetrzna funkcja realizujaca proces treningu.

1.3.1 Metoda gradientowa

Podczas treningu metoda gradientowa sieć wykonuje pasma w przód, aby obliczyć prognozy, a nastepnie pasma wstecz, aby zaktualizować wagi i biasy na podstawie obliczonych strat i współczynnika uczenia sie.

1.3.2 Algorytm ewolucyjny

Dla metody treningu ewolucyjnego ewoluuje sie populacje sieci neuronowych przez pokolenia. Obejmuje to operacje selekcji, krzyżowania i mutacji, aby stworzyć potomstwo z potencjalnie lepsza zdolnościa predykcyjna.

1.4 Predykcja

Po treningu funkcja predict jest używana do obliczenia wyjścia sieci dla danych wejściowych poprzez przeprowadzenie pasma w przód przez sieć.

2 Opis przeprowadzonych eksperymentów numerycznych

2.1 Dane do badania

Niech dana bedzie funkcja $f : [-10, 10] \to R$ o nastepujacej postaci:

$$f(x) = x^2 \cdot \sin(x) + 100 \cdot \sin(x) \cdot \cos(x)$$

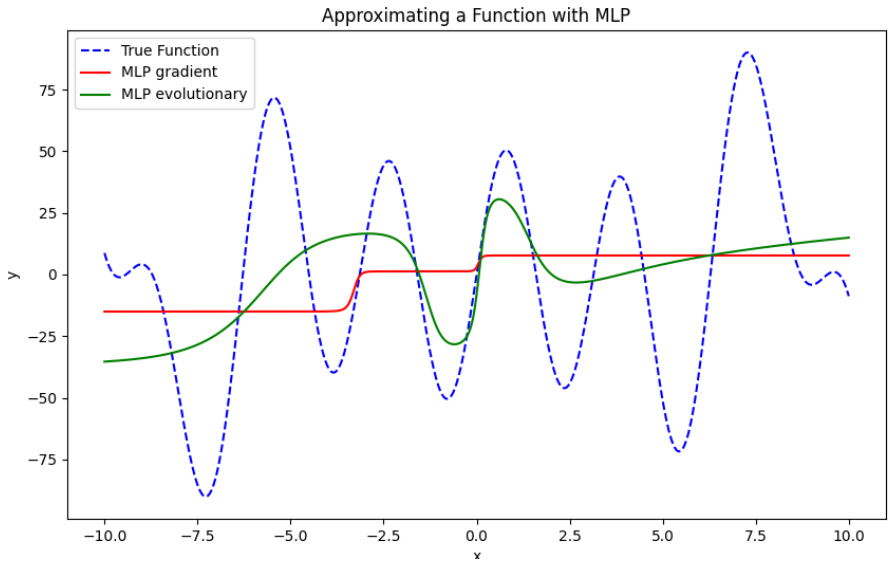
Na podstawie tej funkcji przeprowadzone zostana badania majace na celu analize zachowania i skuteczności aproksymacji przy użyciu perceptronu wielowarstwowego.

2.2 Eksperymenty

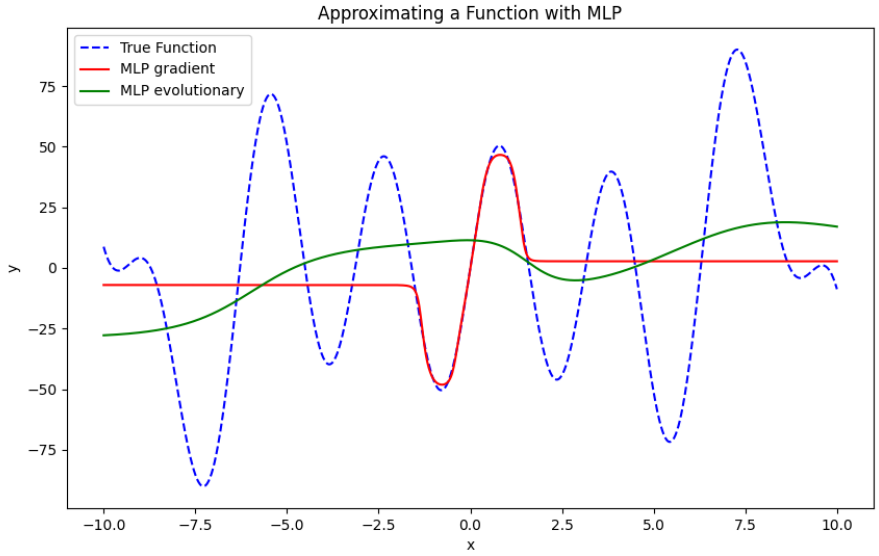
Sprawdzone zostanie, jaki wpływ na wynik ma liczba neuronów w warstawie na jakość uzyskanej aproksymacji, w tym celu liczbe neuronów dla warstw ukrytych bedzie zwiekszana, zaczynajac od małej liczby (4, 8), zakończac na dużej (128, 256). Wszystkie liczby neuronów beda sprawdzane dla obydwu metod i tych samych danych testowych. Dzieki temu sprawdze też jakość aproksymacji zależnie od metody uczenia.

3 opis uzyskanych wyników

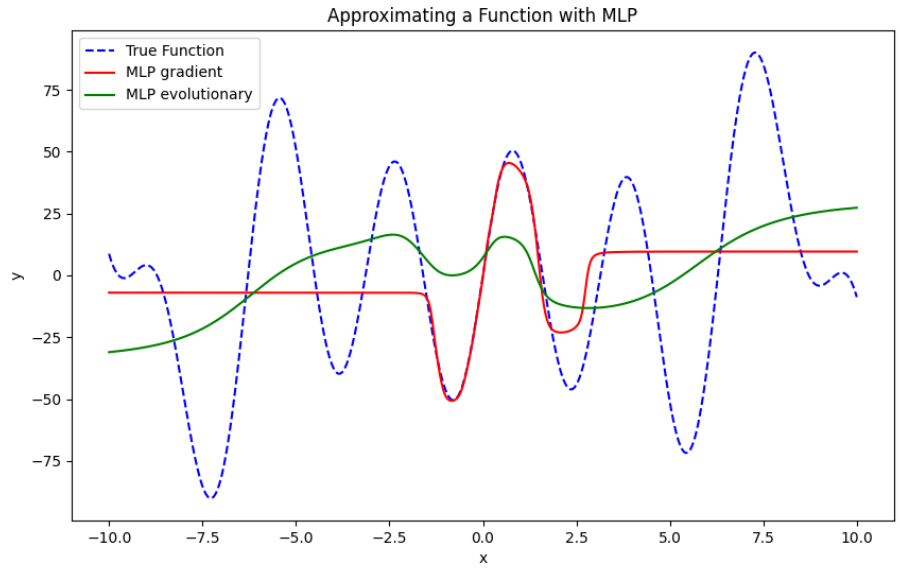
3.1 4, 8 neuronów w warstwie



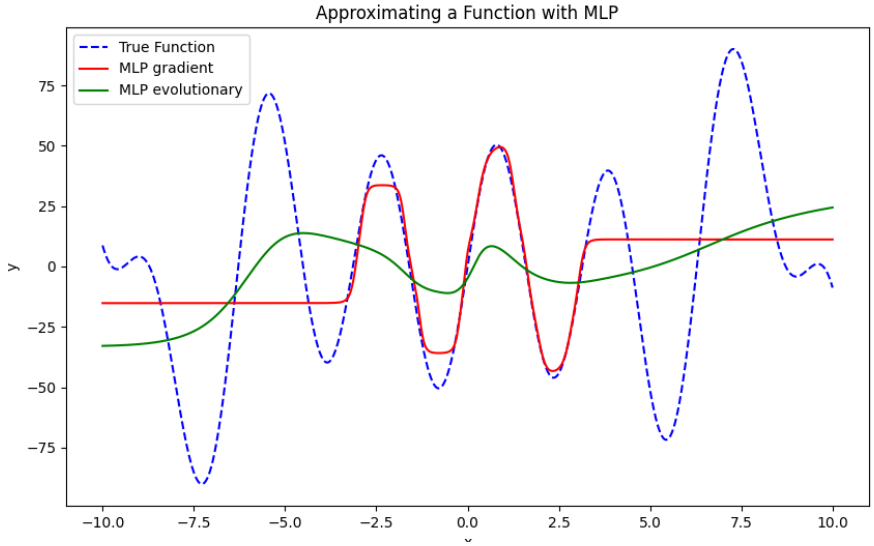
3.2 8, 16 neuronów w warstwie



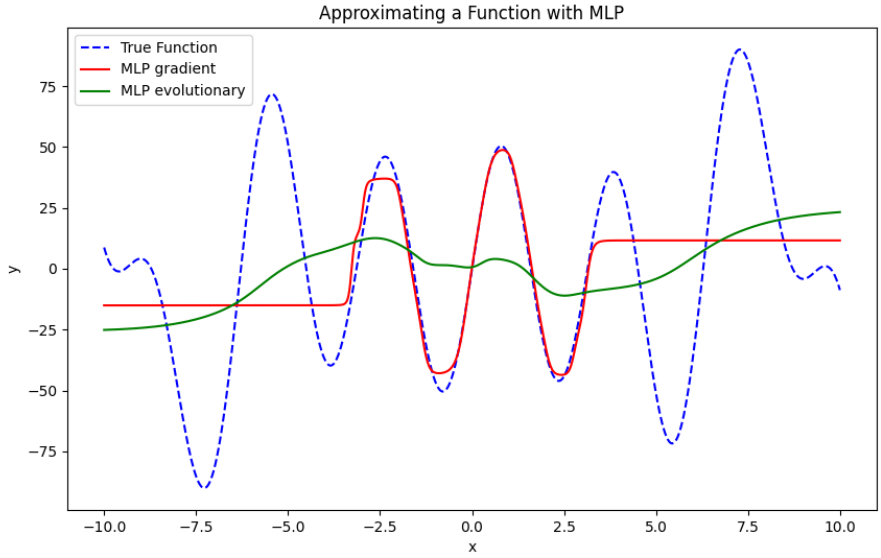
3.3 16, 32 neuronów w warstwie



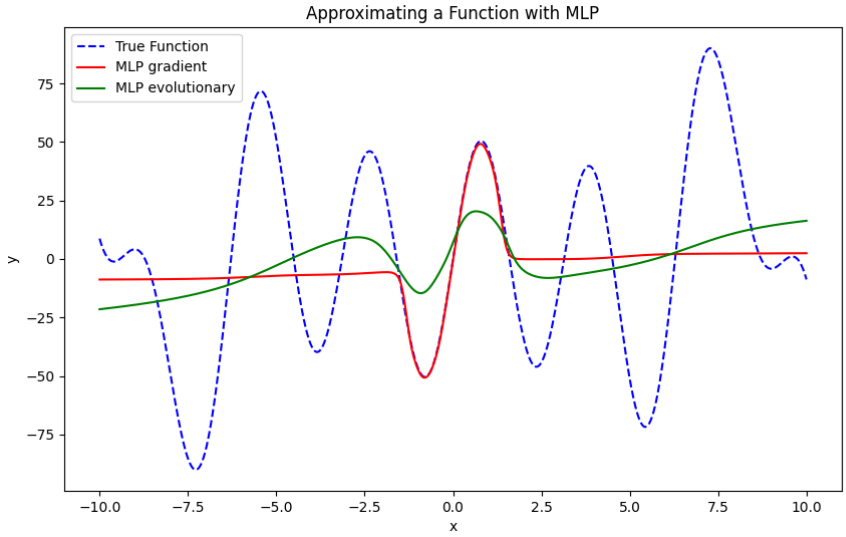
3.4 32, 64 neuronów w warstwie



3.5 64, 128 neuronów w warstwie



3.6 128, 256 neuronów w warstwie



4 Wnioski z przeprowadzonych badań

- Metoda gradientu działa dużo szybciej niż ewolucyjna.
- Zwiekszanie liczby neuronów w warstwie zwieksza jakość aproksymacji dla metody ewolucyjnej.
- Zwiekszanie liczby neuronów w warstwie nie zwieksza jakości aproksymacji dla metody gradientowej.
- Dla bardzo małej liczby neuronów lepiej działa metoda gradientu.