

Dokumentacja – laboratorium nr 5

Wprowadzenie do Sztucznej Inteligencji

Dominika Wyszyńska 318409

Bartosz Kowalski 318382

3 stycznia 2024

1 Opis treści zadania

Celem zadania było zaimplementowanie perceptronu wielowarstwowego, który posłuży aproksymacji funkcji. Funkcja podana w zadaniu

$$f :: [-10, 10] \rightarrow \mathbb{R}$$

o następującej postaci:

$$f(x) = x^2 * \sin(x) + 100 * \sin(x) * \cos(x) \quad (1)$$

W pierwszym kroku, należało stworzyć strukturę perceptronu, uwzględniając liczbę warstw ukrytych, ich rozmiar oraz funkcje aktywacji. Kolejnym etapem była inicjalizacja wag sieci, która odbywa się losowo.

Następnie, przewidziano badanie wpływu dwóch czynników na jakość aproksymacji. W zadaniu zaimplementowano funkcje aktywacji sigmoidalną oraz arctan.

Implementacja programu została umieszczona na [gitlabie wydziałowym](#).

2 Opis planowanych eksperymentów numerycznych

Ekspertymenty zostały zaprojektowane w następujący sposób:

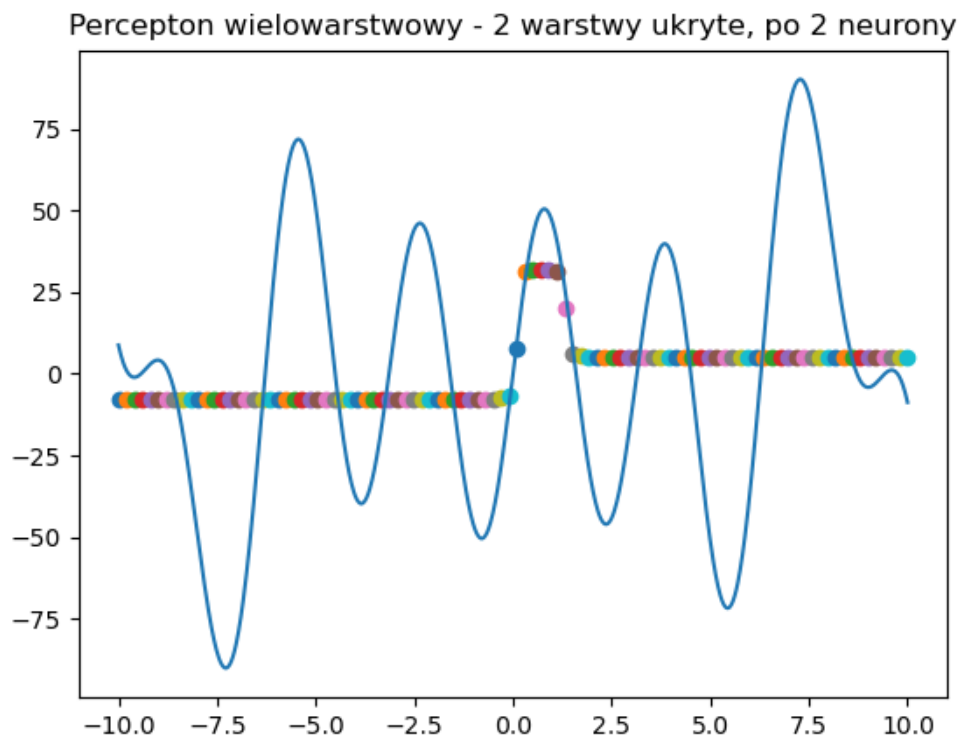
- a) Badanie wpływu liczby neuronów w warstwie na jakość uzyskanej aproksymacji
- b) Badanie wpływu metod optymalizacyjnych do znajdowania wag sieci
 - Metoda gradientowa
 - Metoda ewolucyjna

3 Opis uzyskanych wyników

Zwiększenie liczby iteracji w procesie uczenia perceptronu wielowarstwowego może poprawić jakość aproksymacji funkcji. Dłuższy czas treningu pozwala na lepsze dostosowanie wag sieci do danych, co może skutkować dokładniejszym odwzorowaniem złożonych zależności w funkcji. Jednak istnieje ryzyko przetrenowania, a także większa liczba iteracji wiąże się z obciążeniem obliczeniowym. Stąd w trakcie eksperymentów liczba iteracji była odpowiednio dostosowywana. Dodatkowo na potrzeby eksperymentów beta w metodzie gradientowej została ustawiona na wartość 0.001 oraz wielkość batchy - na 20.

3.1 Badanie wpływu liczby neuronów w warstwie na jakość uzyskanej aproksymacji

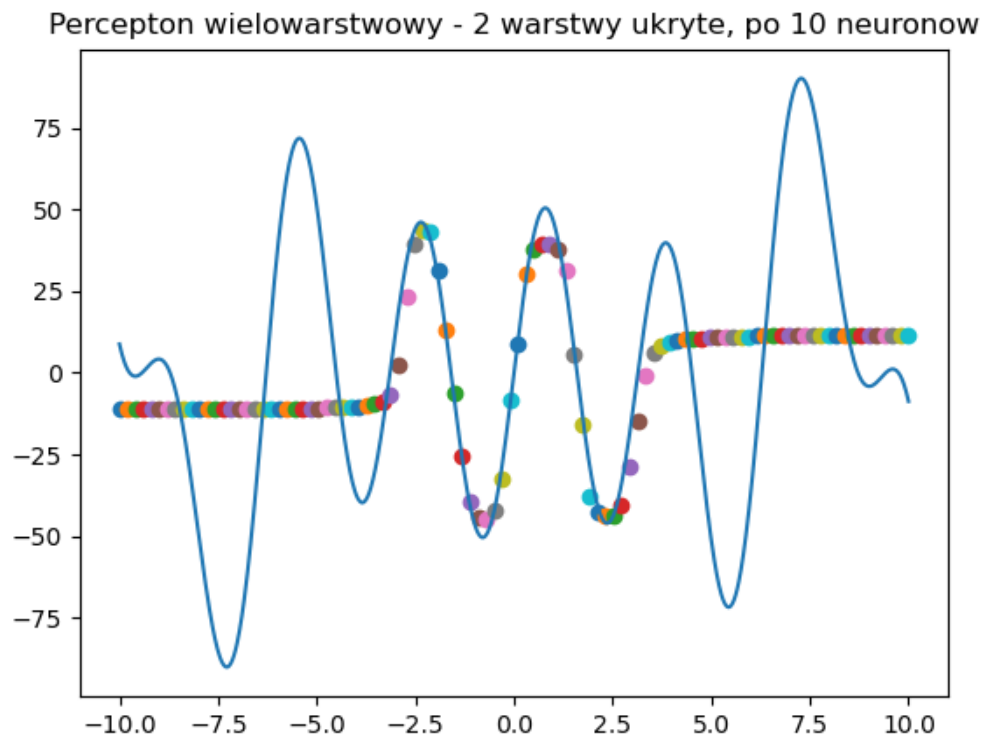
- Perceptron wielowarstwowy z dwoma warstwami ukrytymi - po 2 neurony w każdej warstwie
Jako liczbę iteracji ustawiono 4000.



Rysunek 1: Niebieski wykres - rzeczywisty wykres funkcji podanej w zadaniu

Kolorowy wykres - aproksymowany wykres przy pomocy perceptronu wielowarstwowego

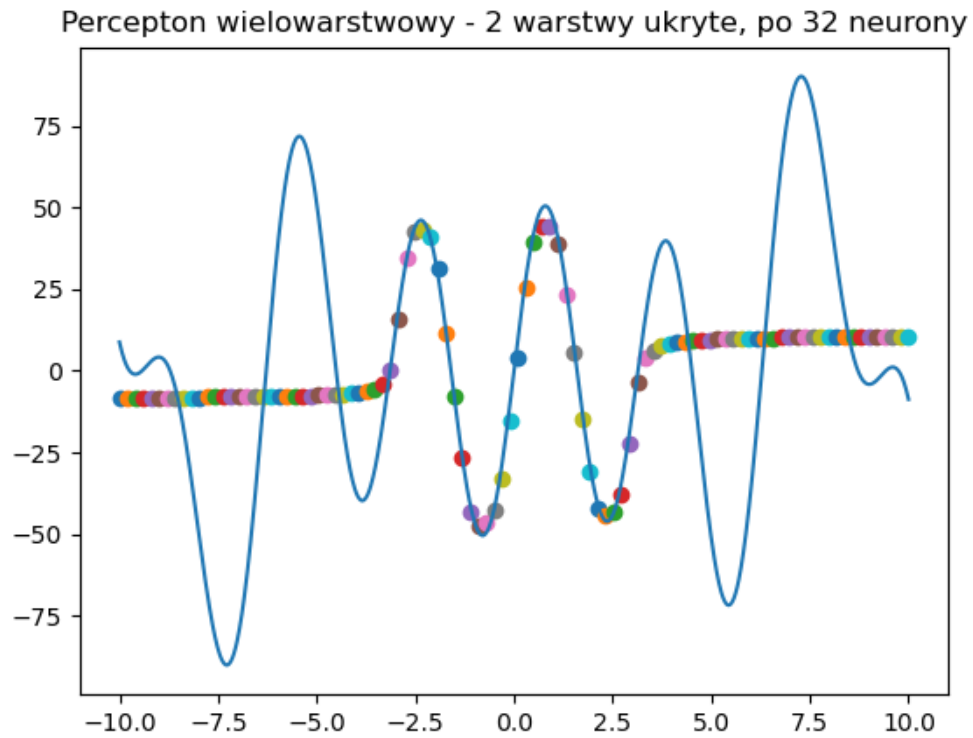
- Perceptron wielowarstwowy z dwoma warstwami ukrytymi - po 10 neuronów w każdej warstwie
Jako liczbę iteracji ustawiono 4000.



Rysunek 2: Niebieski wykres - rzeczywisty wykres funkcji podanej w zadaniu

Kolorowy wykres - aproksymowany wykres przy pomocy perceptronu wielowarstwowego

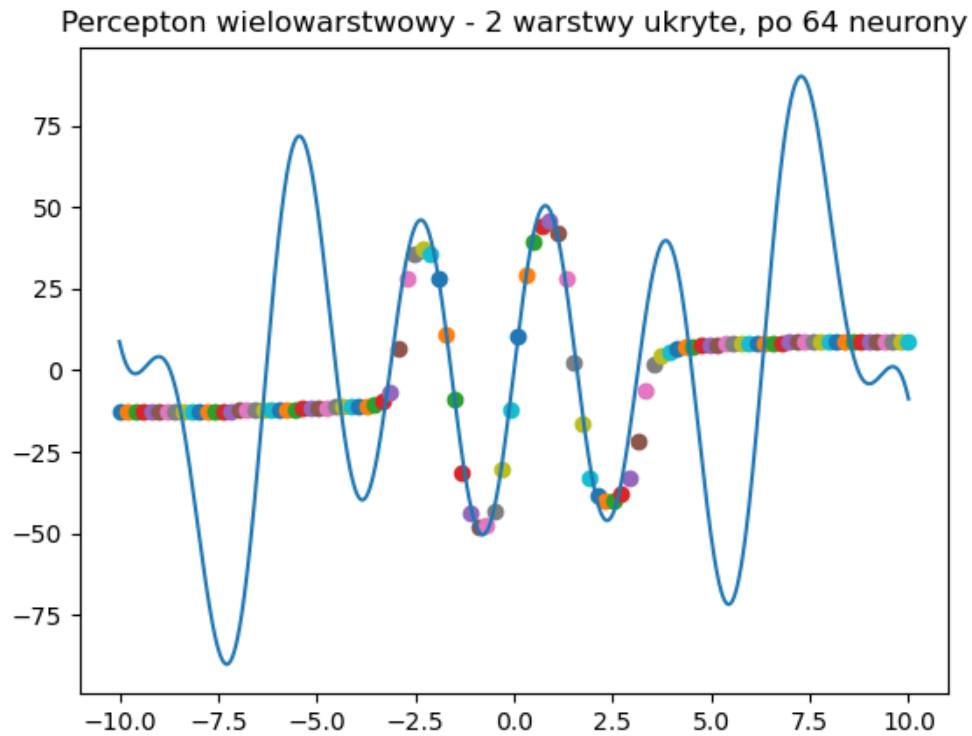
- Perceptron wielowarstwowy z dwoma warstwami ukrytymi - po 32 neurony w każdej warstwie
Jako liczbę iteracji ustawiono 4000.



Rysunek 3: Niebieski wykres - rzeczywisty wykres funkcji podanej w zadaniu

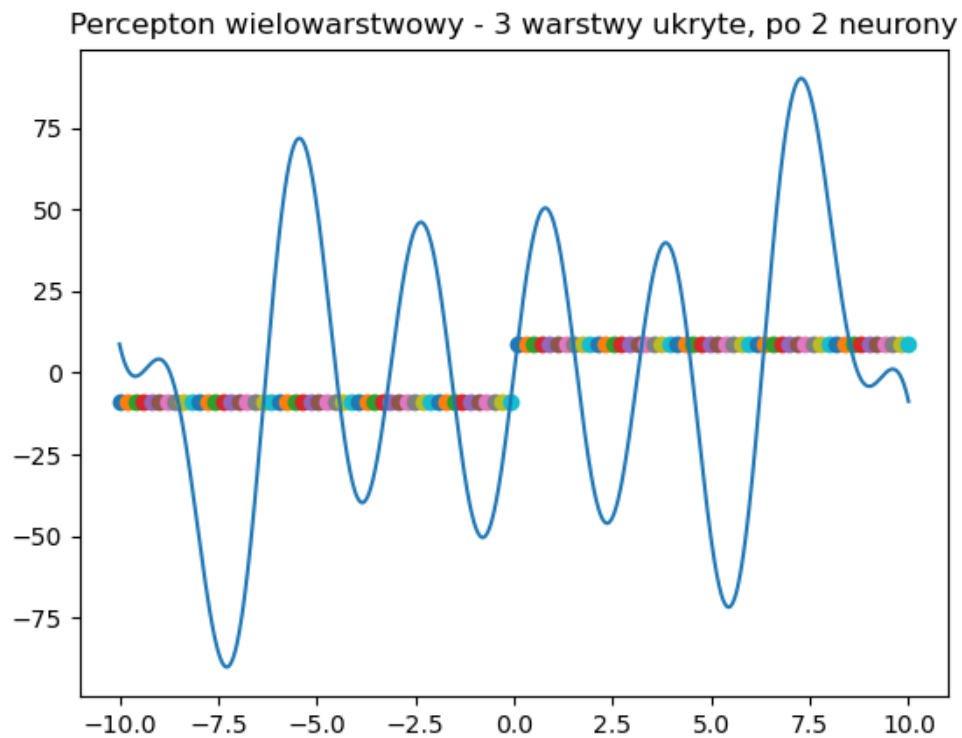
Kolorowy wykres - aproksymowany wykres przy pomocy perceptronu wielowarstwowego

- Perceptron wielowarstwowy z dwoma warstwami ukrytymi - po 64 neurony w każdej warstwie
Jako liczbę iteracji ustawiono 2000, gdyż więcej iteracji trwałoby dość długo.



Rysunek 4: Niebieski wykres - rzeczywisty wykres funkcji podanej w zadaniu
Kolorowy wykres - aproksymowany wykres przy pomocy perceptronu wielowarstwowego

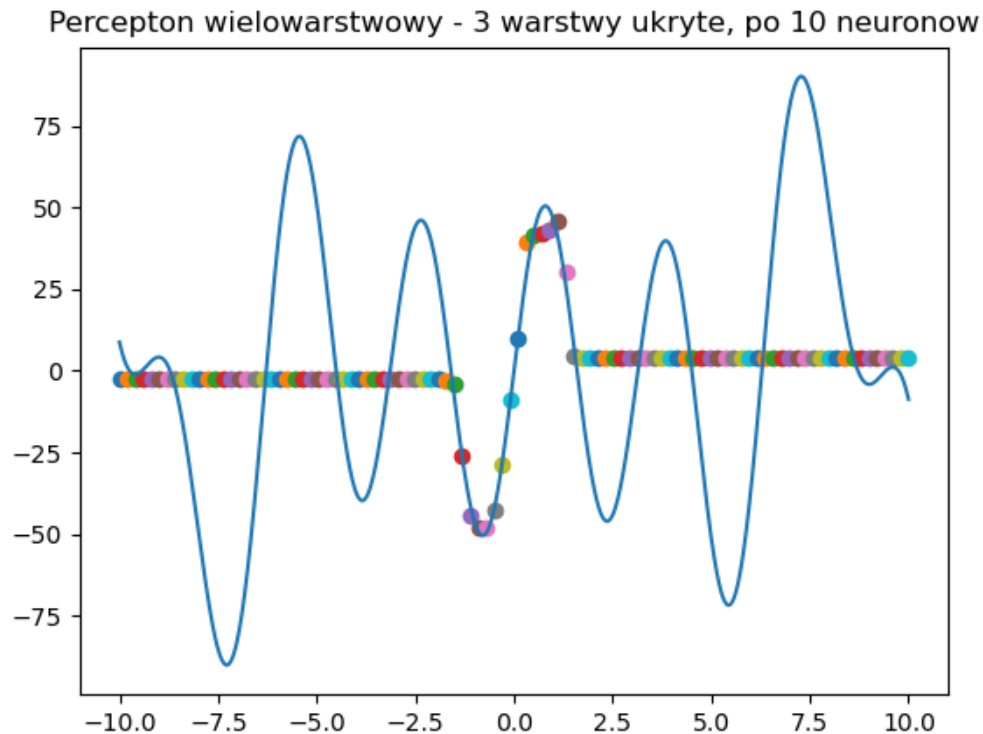
- Perceptron wielowarstwowy z trzema warstwami ukrytymi - po 2 neurony w każdej warstwie Jako liczbę iteracji ustawiono 4000.



Rysunek 5: Niebieski wykres - rzeczywisty wykres funkcji podanej w zadaniu

Kolorowy wykres - aproxymowany wykres przy pomocy perceptronu wielowarstwowego

- Perceptron wielowarstwowy z trzema warstwami ukrytymi - po 10 neuronów w każdej warstwie
Jako liczbę iteracji ustawiono 4000.



Rysunek 6: Niebieski wykres - rzeczywisty wykres funkcji podanej w zadaniu
Kolorowy wykres - aproksymowany wykres przy pomocy perceptronu wielowarstwowego

Wnioski:

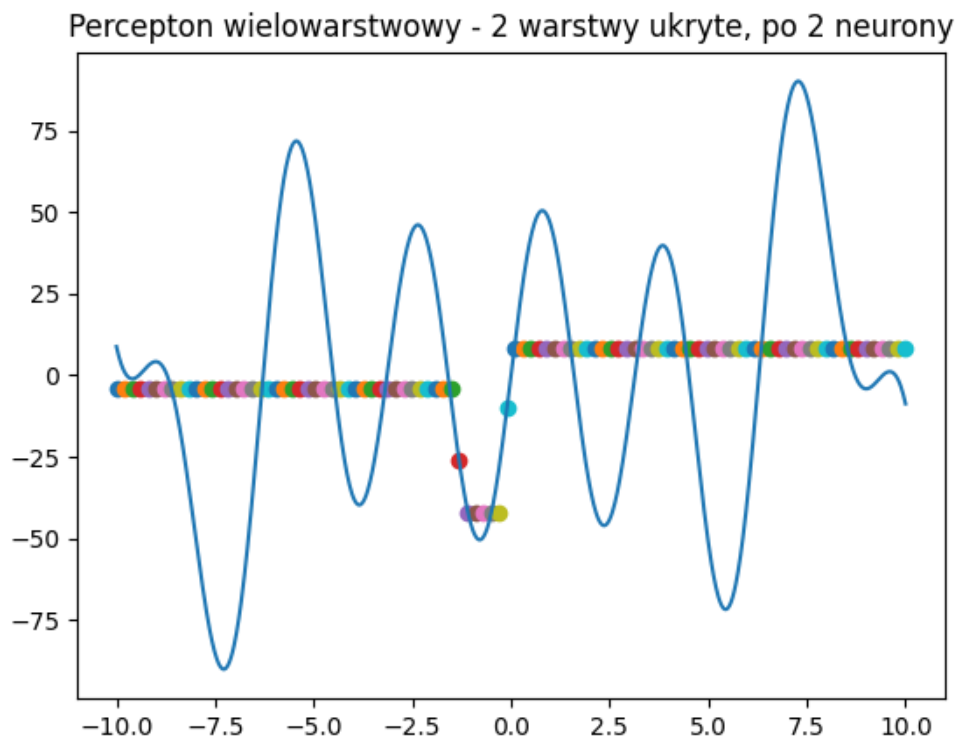
Dodanie większej liczby warstw ukrytych oraz zwiększenie liczby neuronów w tych warstwach pozwala modelowi na lepsze zrozumienie i reprezentację bardziej skomplikowanych zależności w danych. Można powiedzieć, że jest to jak dodanie kolejnych poziomów abstrakcji - każda warstwa może wydobyć bardziej skomplikowane cechy z informacji wejściowych. Jednakże więcej warstw i neuronów może sprawić, że model będzie bardziej złożony, ale także potencjalnie bardziej dopasowany do danych treningowych. Może to prowadzić do przetrenowania. Dodatkowo, dodanie kolejnych warstw i neuronów zwiększa obciążenie obliczeniowe. Oznacza to, że uczenie modelu może wymagać o wiele więcej czasu. W praktyce, optymalna liczba warstw i neuronów zależy od charakteru danych i specyfiki danego zadania.

3.2 Badanie wpływu metod optymalizacyjnych do znajdowania wag sieci

Metoda gradientowa została przetestowana w poprzednim podpunkcie, stąd zostało jedynie przedstawić eksperymenty przy użyciu metody ewolucyjnej. Następnie należało porównać obie metody.

Eksperymenty wykonane przy użyciu metody ewolucyjnej: Populacja wynosi 100 osobników, siła mutacji 0.1, a liczba iteracji 2000. Dla większej liczby neuronów większa liczba iteracji powodowała spore obciążenie obliczeniowe.

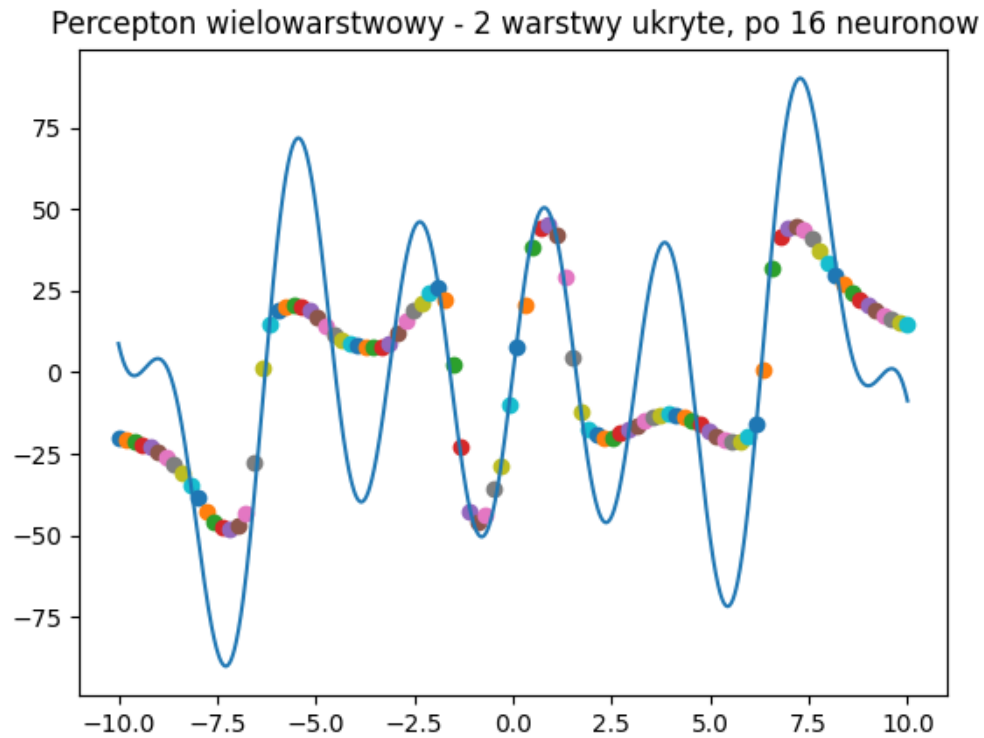
- Perceptron wielowarstwowy przy użyciu metody ewolucyjnej z dwoma warstwami ukrytymi - po 2 neurony w każdej warstwie



Rysunek 7: Niebieski wykres - rzeczywisty wykres funkcji podanej w zadaniu

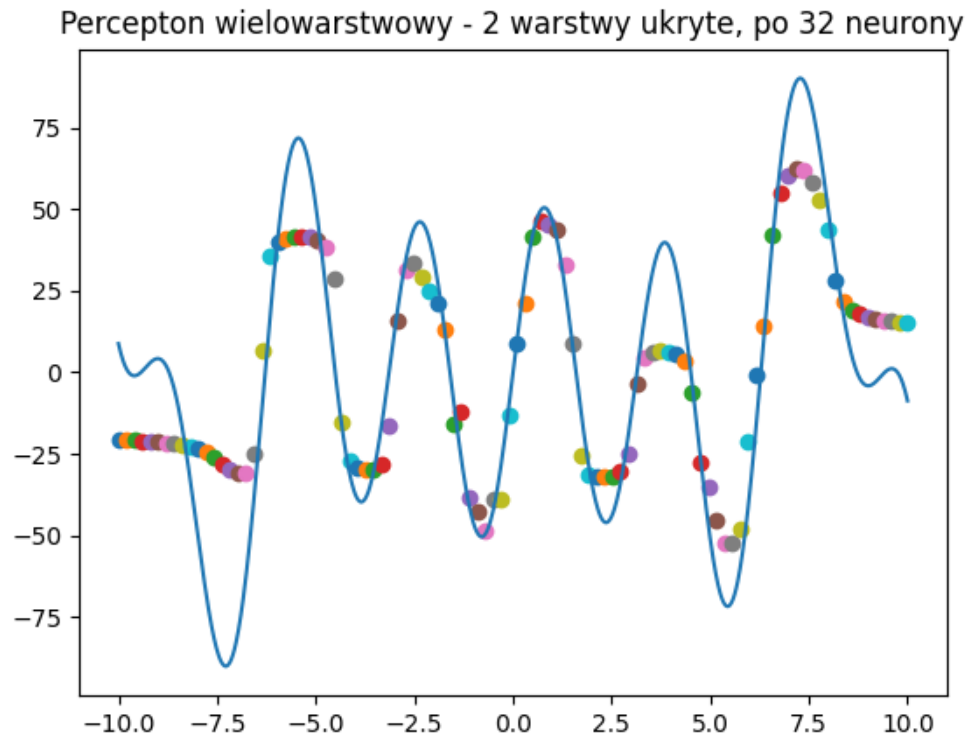
Kolorowy wykres - aproksymowany wykres przy pomocy perceptronu wielowarstwowego

- Perceptron wielowarstwowy przy użyciu metody ewolucyjnej z dwoma warstwami ukrytymi - po 16 neuronów w każdej warstwie



Rysunek 8: Niebieski wykres - rzeczywisty wykres funkcji podanej w zadaniu
Kolorowy wykres - aproksymowany wykres przy pomocy perceptronu wielowarstwowego

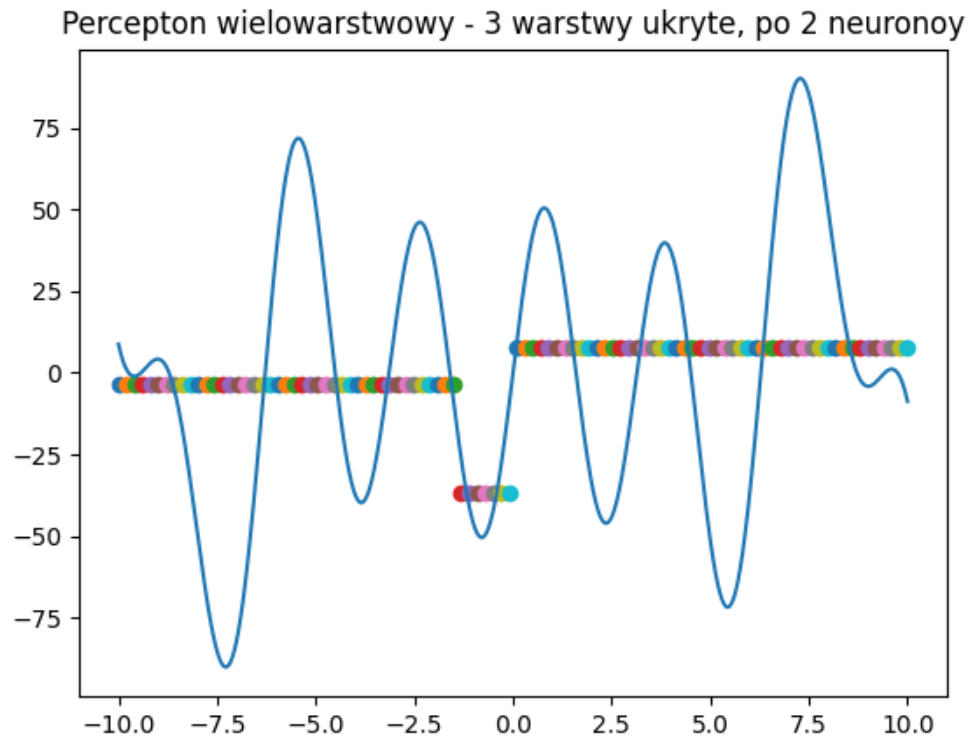
- Perceptron wielowarstwowy przy użyciu metody ewolucyjnej z dwoma warstwami ukrytymi - po 32 neurony w każdej warstwie



Rysunek 9: Niebieski wykres - rzeczywisty wykres funkcji podanej w zadaniu

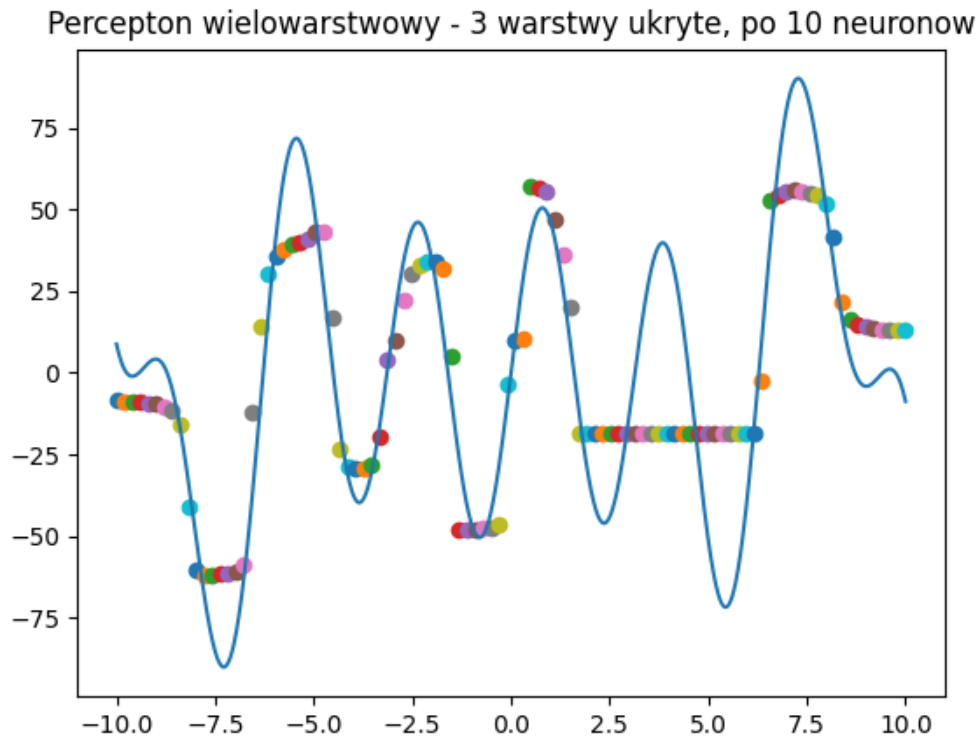
Kolorowy wykres - aproksymowany wykres przy pomocy perceptronu wielowarstwowego

- Perceptron wielowarstwowy przy użyciu metody ewolucyjnej z trzema warstwami ukrytymi - po 2 neurony w każdej warstwie



Rysunek 10: Niebieski wykres - rzeczywisty wykres funkcji podanej w zadaniu
Kolorowy wykres - aproksymowany wykres przy pomocy perceptronu wielowarstwowego

- Perceptron wielowarstwowy przy użyciu metody ewolucyjnej z trzema warstwami ukrytymi - po 10 neuronów w każdej warstwie



Rysunek 11: Niebieski wykres - rzeczywisty wykres funkcji podanej w zadaniu
Kolorowy wykres - aproksymowany wykres przy pomocy perceptronu wielowarstwowego

Porównanie metody gradientowej z metodą ewolucyjną:

Metoda gradientowa dostosowuje wagi zgodnie z gradientem funkcji straty, co może prowadzić do szybszego zbliżania się do minimum lokalnego. Natomiast metoda ewolucyjna bazuje na strategiach genetycznych, przeszukując przestrzeń parametrów bardziej globalnie, co może być korzystne w przypadku złożonych funkcji kosztu. W odniesieniu do wymagań obliczeniowych, metoda gradientowa jest często bardziej efektywna, zwłaszcza przy dużych zbiorach danych, dzięki wykorzystaniu informacji gradientowych do aktualizacji wag. Z kolei metoda ewolucyjna może wymagać większych zasobów, zwłaszcza przy manipulacji populacją rozwiązań i operacjami genetycznymi.

Ostateczny wybór między tymi metodami zależy od specyfiki problemu oraz dostępności zasobów obliczeniowych.

4 Podsumowanie i wnioski

Celem tych eksperymentów było zrozumienie wpływu różnych parametrów oraz metod na jakość uzyskanej aproksymacji przy użyciu perceptronu wielowarstwowego. Zwiększanie liczby warstw i neuronów daje modelowi większą zdolność do nauki złożonych wzorców, ale jednocześnie niesie ze sobą ryzyko przetrenowania i zwiększoną złożoność obliczeniową.

Metoda gradientowa, skupiając się na aktualizacji wag w kierunku lokalnie optymalnego minimum, może prowadzić do lepszej dokładności w okolicach środka przedziału. Jednak, wraz z oddalaniem się od tego obszaru, może tracić zdolność do ogólnego uchwycenia wzorców.

Metoda ewolucyjna, będąc bardziej globalnym podejściem, może prowadzić do bardziej równomiernej dokładności na całym przedziale, ponieważ przeszukuje przestrzeń rozwiązań bez ograniczeń na lokalne ekstrema. Jednak, może być mniej efektywna w dostosowywaniu się do lokalnych szczegółów, co może prowadzić do mniejszej dokładności w okolicach ekstremów.