# Wprowadzenie do Sztucznej Inteligencji

## Laboratorium 1

Metoda Gradientu Prostego

Piotr Niedziałek

304474

## 1. Wstęp

Metoda gradientu prostego jest to algorytm numeryczny służący do odnajdywania minimum danej funkcji. Metodę tę wykorzystuje się do uczenia sieci neuronowej podczas optymalizacji funkcji kosztu. Działanie metody polega na wyliczaniu gradientu w punkcie startowym i poruszaniu się w kierunku przeciwnym, niż wskazuje gradient. Robi się tak, ponieważ gradient wskazuje kierunek najszybszego wzrostu w danym punkcie funkcji.
W kolejnych krokach liczy się wartość funkcji dla punktu przesuniętego o wartość gradientu pomnożoną przez stałą uczenia. Parametr uczenia $\beta$ wynosi zazwyczaj od 0.1 do 0.0001. Poniżej przedstawiono kolejne kroki działania algorytmu:

1. Wybierz punkt początkowy $x_0$.
2. $x_{k+1} = x_k - \beta \nabla f(x_k)$.
3. Sprawdź kryterium stopu.
4. Powtarzaj punkty 2 i 3 przez zadaną ilość iteracji lub do momentu spełnienia kryterium stopu.
5. Zwróć szukaną wartość $x$ i wartość funkcji w tym punkcie $f(x)$.

Kryterium stopu może być wartość normy gradientu w punkcie mniejsza od bardzo małego epsilon:

$$||\nabla f(x_k)|| \leq \varepsilon$$

## 2. Zadanie

Celem laboratorium było zaimplementowanie metody gradientu prostego dla trzech funkcji oraz przeprowadzenie eksperymentów zbieżności manipulując parametrami uczenia oraz punktami inicjalizacji. Wszystkie z nich mają minimum globalne w $f(0,0) = 0$
Zastosowane funkcje to:

- Suma kwadratów:

$$f(\boldsymbol{x}) = x_1^2 + x_2^2$$

Gradient:

$$\nabla f = [2 \cdot x_1 , 2 \cdot x_2]$$

Zakres:

$$x \in [-10, 10], x \in R^2$$

- Funkcja Rastrigina:

$$f(\boldsymbol{x}) = 20 + x_1^2 + x_2^2 - 10 \cdot (\cos(2 \cdot \pi \cdot x_1) + \cos(2 \cdot \pi \cdot x_2))$$

Gradient:

$$\nabla f = [2 \cdot x_1 + 20 \cdot \pi \cdot \sin(2 \cdot \pi \cdot x_1) , 2 \cdot x_2 + 20 \cdot \pi \cdot \sin(2 \cdot \pi \cdot x_2) ]$$

Zakres:

$$x \in [-5.12, 5.12], x \in R^2$$

- Funkcja Griewanka:

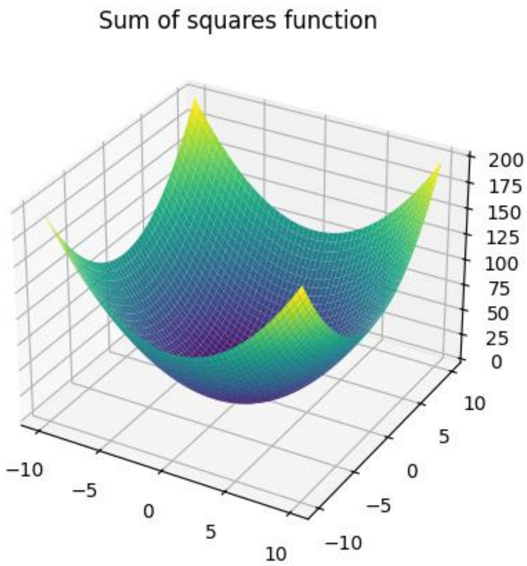
$$f(\boldsymbol{x}) = \frac{x_1^2}{4000} + \frac{x_2^2}{4000} - \cos(x_1) \cdot \cos\left(\frac{x_2}{\sqrt{2}}\right) + 1$$

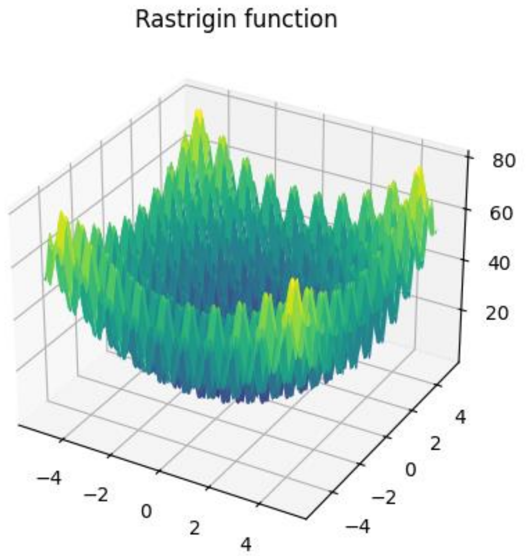
Gradient:

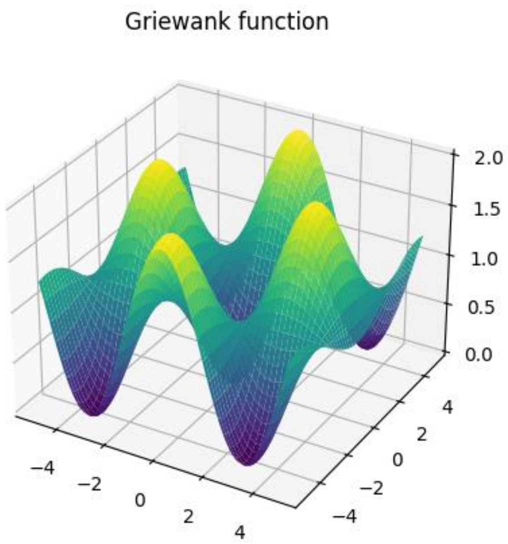
$$\nabla f = \left[\frac{x_1}{2000} + \sin(x_1) \cdot \cos\left(\frac{x_2}{\sqrt{2}}\right), \frac{x_2}{2000} + \cos(x_1) \cdot \sin\left(\frac{x_2}{\sqrt{2}}\right) \cdot \frac{1}{\sqrt{2}}\right]$$

Zakres:

$$x \in [-5, 5], x \in R^2$$



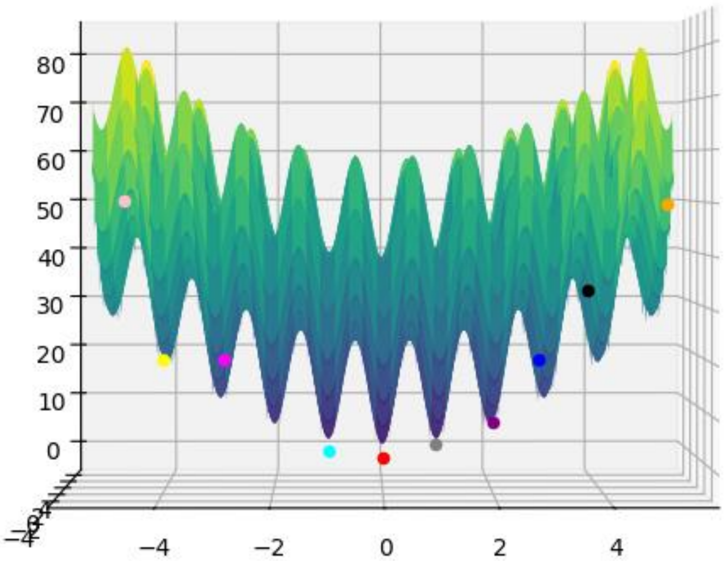



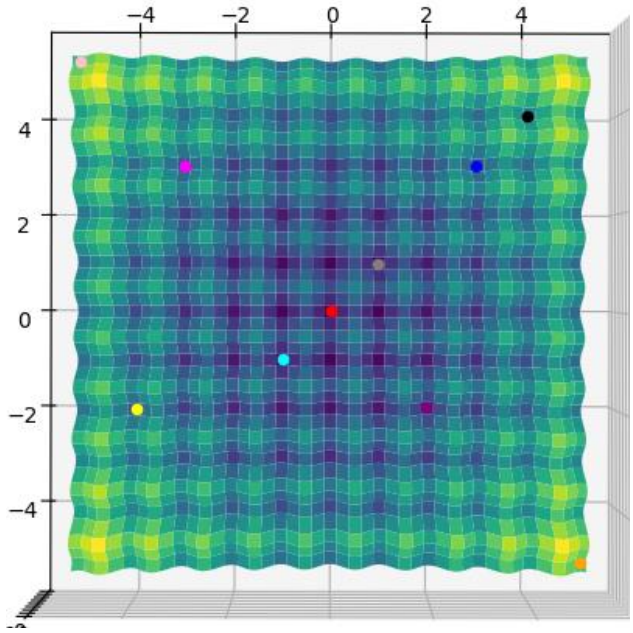
## 3. Wyniki

### 3.1. Badanie różnych punktów inicjalizacji

Dla każdej funkcji przetestowano zbieżność algorytmu w zależności od punktu inicjalizacji. Algorytm został uruchomiony dla 10 różnych punktów. Wyniki przedstawiono na wykresach i w tabeli. Wartość kroków była stała i wyniosła 1000.

- **Funkcja Rastrigina**
  Przyjęto wartość współczynnika $\beta = 0.001$.

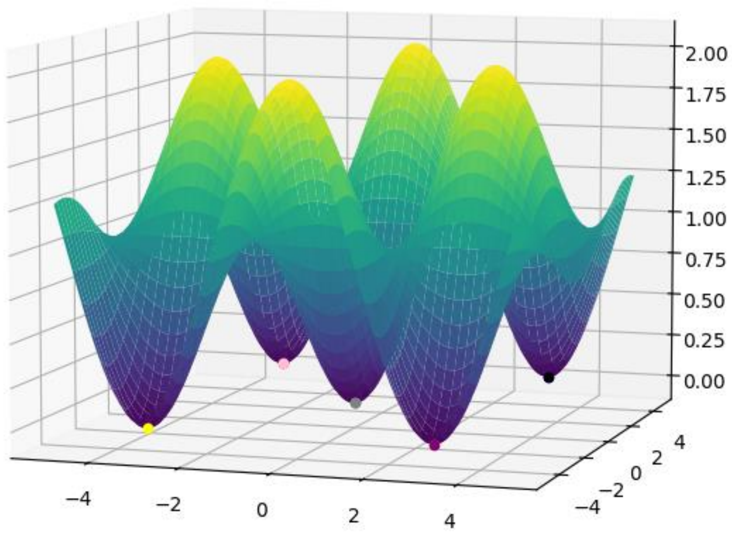


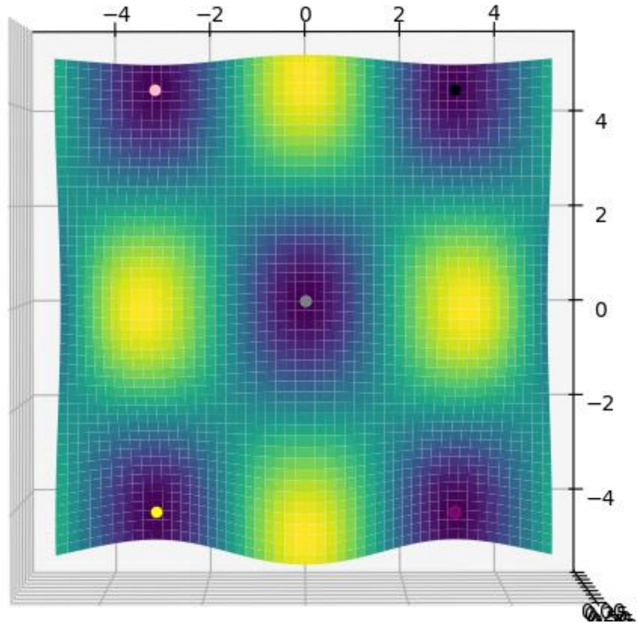




| Punkt inicjalizacji | Punkt osiągnięty | Wartość funkcji w punkcie | Wartość oczekiwana | Błąd SE | Błąd L1 |
|---|---|---|---|---|---|
| 0.5, -0.5 | 1.61e-217, -1.61e-217 | 0.0 | 0.0 | 0.0 | 0.0 |
| 3 , 3 | 2.98, 2.98 | 17.91 | - | 320.7681 | 17.91 |
| 4, 4 | 3.98, 3.98 | 31.84 | - | 1013.786 | 31.84 |
| 5, -5 | 4.97, -4.97 | 49.75 | - | 2475.063 | 49.75 |
| -4, -2 | -3.98, -1.99 | 19.9 | - | 396.01 | 19.9 |
| -1, -1 | -0.99, -0.99 | 1.99 | - | 3.9601 | 1.99 |
| 2, -2 | 1.99, -1.99 | 7.96 | - | 63.3616 | 7.96 |
| 1, 1 | 0.99, 0.99 | 1.99 | - | 3.9601 | 1.99 |
| -3, 3 | -2.98, 2.98 | 17.91 | - | 320.7681 | 17.91 |
| -5, 5 | -4.97, 4.97 | 49.75 | - | 2475.063 | 49.75 |

$$MSE = \frac{\sum SE}{N} = 707{,}27386$$

- **Funkcja Griewanka**
  Przyjęto wartość współczynnika $\beta = 0.1$




| Punkt inicjalizacji | Punkt osiągnięty | Wartość funkcji w punkcie | Wartość oczekiwana | Błąd SE | Błąd L1 |
|---|---|---|---|---|---|
| 0.5, -0.5 | 9.13e-47, -2.63e-23 | 0.0 | 0.0 | 0.0 | 0.0 |
| 3 , 3 | 3.14, 4.44 | 0.0074 | - | 0.00005476 | 0.0074 |
| 4, 4 | 3.14, 4.44 | 0.0074 | - | 0.00005476 | 0.0074 |
| 5, -5 | 3.14, -4.44 | 0.0074 | - | 0.00005476 | 0.0074 |
| -4, -2 | -3.14, -4.44 | 0.0074 | - | 0.00005476 | 0.0074 |
| -1, -1 | -2.6e-46, -6.25e-23 | 0.0 | - | 0.0 | 0.0 |
| 2, -2 | 3.14, -4.44 | 0.0074 | - | 0.00005476 | 0.0074 |
| 1, 1 | -2.6e-46, 6.25e-23 | 0.0 | - | 0.0 | 0.0 |
| -3, 3 | -3.14, 4.44 | 0.0074 | - | 0.00005476 | 0.0074 |
| -5, 5 | -3.14, 4.44 | 0.0074 | - | 0.00005476 | 0.0074 |

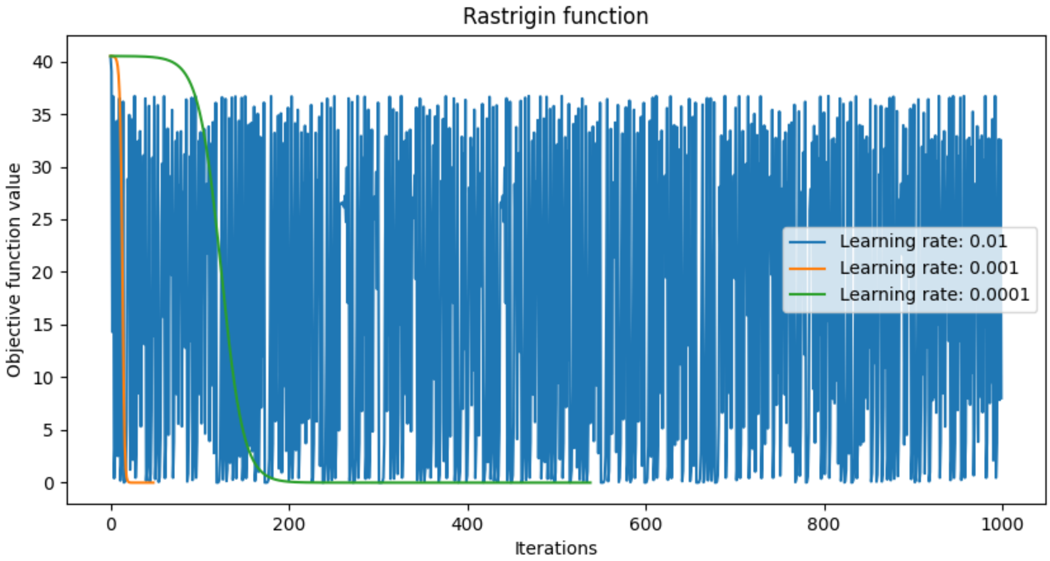
$$MSE = \frac{\sum SE}{N} = 0{,}000038$$

### 3.2. Badanie różnych wartości parametru $\beta$

Dla obu funkcji przeprowadzono badanie zbieżności do minimum w zależności od ustawionego parametru $\beta$. Ilość iteracji ustawiona została na 1000. Ponadto zastosowano kryterium stopu: norma gradientu w punkcie mniejsza od $\varepsilon = 10^{-5}$.

- Funkcja Rastrigina

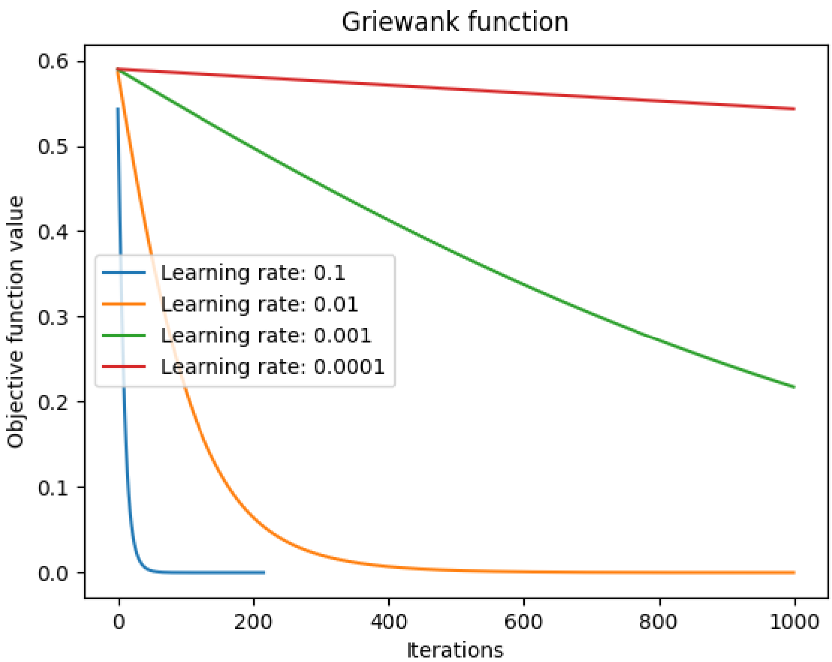
Jako punkt inicjalizacji wybrano (0.5, 0.5)




| **Wartość $\beta$** | **Ilość iteracji** | **Osiągnięta wartość** |
|---|---|---|
| 0.01 | 1000 | 8.02 |
| 0.001 | 49 | 3.55e-14 |
| 0.0001 | 539 | 1.14e-13 |

- Funkcja Griewanka

  Jako punkt inicjalizacji wybrano (1, 1)



| Wartość $\beta$ | Ilość iteracji | Osiągnięta wartość |
|---|---|---|
| 0.1 | 217 | 8.14e-11 |
| 0.01 | 1000 | 1.64e-5 |
| 0.001 | 1000 | 0.22 |
| 0.0001 | 1000 | 0.54 |

## 4. Wnioski

Metoda gradientu prostego jest bardzo wrażliwa na występowanie minimów lokalnych. W przypadku funkcji Rastrigina dla wszystkich punktów początkowych algorytm zatrzymał się w innych minimach lokalnych. Jest to typowy problem dla tego algorytmu – globalne minimum w wielu przypadkach nie zostaje odnalezione.

Manipulowanie parametrem $\beta$ może znacząco przyspieszyć działanie algorytmu. Na przykład w funkcji Griewanka dla wartości $\beta = 0.1$ osiągnięto zadawalający wynik prawie 5 razy szybciej niż dla wartości $\beta = 0{,}01$, a dla jeszcze mniejszych wartości $\beta$ nie osiągnięto zbieżności w zadanej ilości iteracji. Jednakże, w przypadku funkcji Rastrigina parametr $\beta = 0.1$ oraz $\beta = 0.01$ nie osiągnęły zbieżności. Dla tego drugiego algorytm wpadł w oscylacje w okolicach minimum. Dopiero dziesięciokrotne zmniejszenie wartości parametru pozwoliło osiągnąć zbieżność. Dobór wartości parametru zależy więc od rodzaju funkcji, której optimum poszukujemy.