# Sprawozdanie z implementacji perceptronu i adaline

**Wojciech Kozłowski**

## Abstract

Celem pierwszej listy była samodzielna implementacja algorytmów perceptronu oraz adaptacyjnego neuronu liniowego. Zadanie miało na celu nie tylko poznanie, ale także dogłębne zrozumienie oraz poznanie dobrych i słabych stron wyżej wymienionych algorytmów. Do ich implementacji użyłem języka python, oraz biblioteki numpy. Kod dostępny jest na serwisie GitHub (Link). Poniższy raport ma na celu przedstawienie wyników badań dotyczących wpływu hiperparametrów modeli na ich jakość, a także porównanie obu modeli.

# 1 Perceptron

## 1.1 Badanie wpływu początkowych wartości progu na szybkość zbieżności perceptronu

### 1.1.1 Parametry eksperymentu

Tabela 1: Wartości parametrów eksperymentu

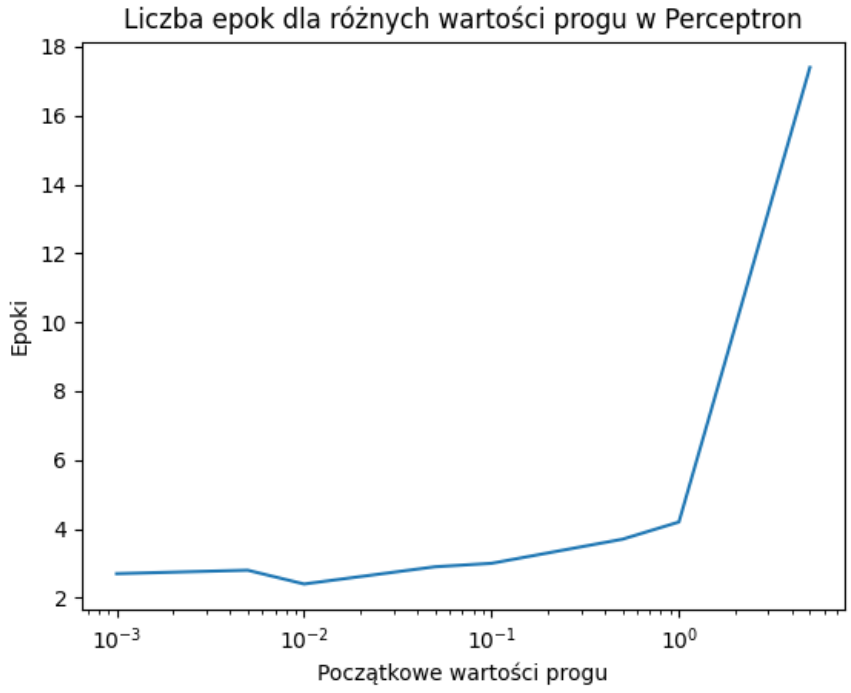
| Parametr | Wartość |
|---|---|
| Stałe eksperymentu | |
| Model | Perceptron |
| Współczynnik uczenia | 0.05 |
| Zakres początkowych wartości wag | (-0.01, 0.01) |
| Funkcja progowa | Bipolarna |
| Dane treningowe | bramka logiczna AND |
| Maksymalna liczba epok | 100 |
| Ilość powtórzeń eksperymentu | 10 |
| Ziarno losowości | 0 |
| Zmienne eksperymentu | |
| Początkowe wartości progu | 0.001; 0.005; 0.01; 0.05; 0.1; 0.5; 1; 5 |

### 1.1.2 Wyniki eksperymentu

Tabela 2: Wyniki eksperymentu

| Próg | 0.001 | 0.005 | 0.01 | 0.05 | 0.1 | 0.5 | 1 | 5 |
|---|---|---|---|---|---|---|---|---|
| Epoki | 2.7 | 2.8 | 2.4 | 2.9 | 3.0 | 3.7 | 4.2 | 17.4 |

### 1.1.3 Wykres wyników



### 1.1.4 Wnioski

Wyniki eksperymentu wskazują, że gdy wartości początkowe progu są większe, to ilość epok potrzebnych do uzyskania optymalnego modelu znacząco się zwiększa. Jest to najprawdopodobniej spowodowane faktem, że małe początkowe wartości wag generują małą wartość iloczynu skalarnego z wejściami, a to z kolei sprawia (przy dużej wartości progu), że wszystkie próbki są klasyfikowane jako klasa negatywna. Dopiero duża liczba epok zwiększy wagi oraz zmniejszy próg, na tyle, że wszystkie parametry modelu osiągną stan równowagi.

## 1.2 Badanie wpływu początkowych wartości wag na szybkość zbieżności perceptronu

### 1.2.1 Parametry eksperymentu

Tabela 3: Wartości parametrów eksperymentu

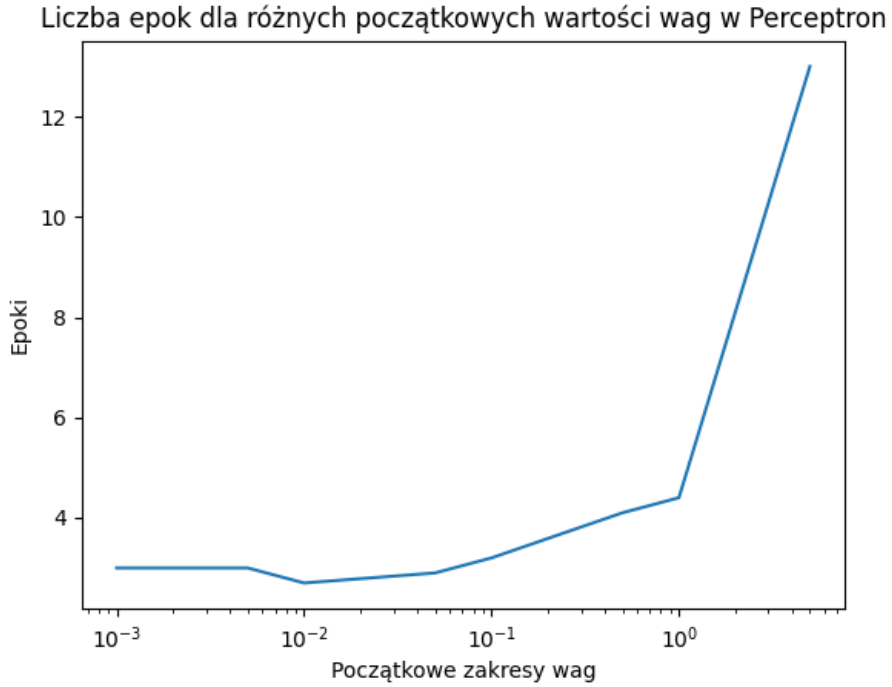
| Parametr | Wartość |
|---|---|
| Stałe eksperymentu | |
| Model | Perceptron |
| Współczynnik uczenia | 0.05 |
| Funkcja progowa | Bipolarna |
| Zakres początkowych wartości progu | (-0.01, 0.01) |
| Dane treningowe | bramka logiczna AND |
| Maksymalna liczba epok | 100 |
| Ilość powtórzeń eksperymentu | 10 |
| Ziarno losowości | 0 |
| Zmienne eksperymentu | |
| Zakresy początkowych wartości wag | (-0.001, 0.001)<br>(-0.005, 0.005)<br>(-0.01, 0.01)<br>(-0.05, 0.05)<br>(-0.1, 0.1)<br>(-0.5, 0.5)<br>(-1, 1)<br>(-5, 5) |

### 1.2.2 Wyniki eksperymentu

Tabela 4: Wyniki eksperymentu

| Zakres pocz. wartości wag | $\pm 0.001$ | $\pm 0.005$ | $\pm 0.01$ | $\pm 0.05$ | $\pm 0.1$ | $\pm 0.5$ | $\pm 1$ | $\pm 5$ |
|---|---|---|---|---|---|---|---|---|
| Epoki | 3.0 | 3.0 | 2.7 | 2.9 | 3.2 | 4.1 | 4.4 | 13.0 |

### 1.2.3 Wykres wyników




### 1.2.4 Wnioski

Jak widać na tabeli wyników oraz wykresie, im większy zakres początkowych wartości wag, tym większa liczba epok potrzebnych do stworzenia modelu poprawnie klasyfikującego wszystkie próbki. Hiperpłaszczyznę rozdzielającą dla dwuwymiarowych danych możemy rozumieć jako prostą o wzorze:

$$w_1 \cdot x_1 + w_2 \cdot x_2 + b = 0 \tag{1}$$

Żeby znaleźć dobrą prostą (rozdzielającą obie klasy) potrzebne jest znalezienie dobrego stosunku wartości $w_1$, $w_2$ oraz $b$. Gdy wszystkie te wartości będą o rząd wielkości większe lub mniejsze - prosta będzie dalej taka sama. Natomiast, gdy pracujemy na większych wartościach, to różnice między wagami są większe, a ich zmiany są stałe, co sprawia, że potrzeba większej liczby epok, by znaleźć dobry stosunek w przypadku, kiedy wartości początkowe są większe.

## 1.3 Badanie wpływu wartości współczynnika uczenia na szybkość zbieżności perceptronu

### 1.3.1 Parametry eksperymentu

Tabela 5: Wartości parametrów eksperymentu

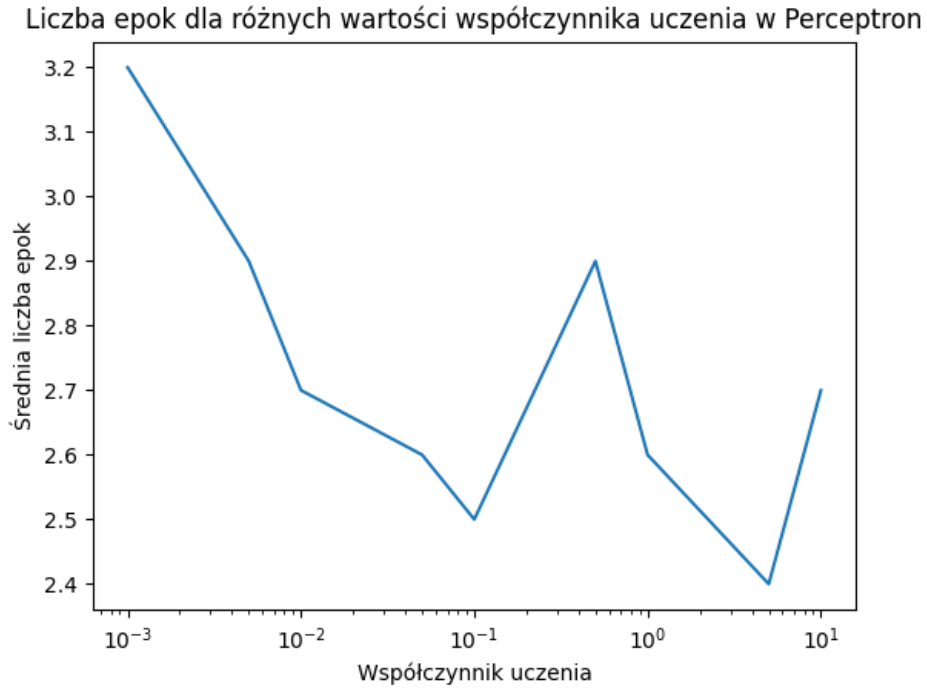
| Parametr | Wartość |
|---|---|
| Stałe eksperymentu | |
| Model | Perceptron |
| Zakres początkowych wartości progu | (-0.01, 0.01) |
| Zakres początkowych wartości wag | (-0.01, 0.01) |
| Funkcja progowa | Bipolarna |
| Dane treningowe | bramka logiczna AND |
| Maksymalna liczba epok | 100 |
| Ilość powtórzeń eksperymentu | 10 |
| Ziarno losowości | 0 |
| Zmienne eksperymentu | |
| Współczynnik uczenia | 0.001; 0.005; 0.01; 0.05; 0.1; 0.5; 1; 5; 10 |

### 1.3.2 Wyniki eksperymentu

Tabela 6: Wyniki eksperymentu

| współczynnik uczenia | 0.001 | 0.005 | 0.01 | 0.05 | 0.1 | 0.5 | 1 | 5 | 10 |
|---|---|---|---|---|---|---|---|---|---|
| Epoki | 3.2 | 2.9 | 2.7 | 2.6 | 2.5 | 2.9 | 2.6 | 2.4 | 2.7 |

### 1.3.3 Wykres wyników



### 1.3.4 Wnioski

Jak wynika z przeprowadzonego eksperymentu, im większy współczynnik uczenia tym model szybciej zbiega. Małe wartości współczynnika uczenia sprawiają, że potrzeba wiele kroków, by dojść do optimum. Duże wartości sprawiają z kolei, że wagi aktualizują się o większe wartości i dobra prosta jest znaleziona wcześniej. Nie jest to jednak reguła, gdyż wiele zależy tu od danych oraz modelu. Dane są bardzo proste, a margines między klasami jest na tyle szeroki, że mimo dużych zmian parametrów, model trafia na prostą leżącą na marginesie.

### 1.4 Badanie wpływu funkcji aktywacyjnej na szybkość zbieżności perceptronu

#### 1.4.1 Parametry eksperymentu

Tabela 7: Wartości parametrów eksperymentu

| Parametr | Wartość |
|---|---|
| Stałe eksperymentu | |
| Model | Perceptron |
| Współczynnik uczenia | 0.05 |
| Zakres początkowych wartości progu | (-0.01, 0.01) |
| Zakres początkowych wartości wag | (-0.01, 0.01) |
| Dane treningowe | bramka logiczna AND |
| Maksymalna liczba epok | 100 |
| Ilość powtórzeń eksperymentu | 10 |
| Ziarno losowości | 0 |
| Zmienne eksperymentu | |
| Funkcja progowa | Unipolarna; Bipolarna |

#### 1.4.2 Wyniki eksperymentu

Tabela 8: Wyniki eksperymentu

| Funkcja | Unipolarna | Bipolarna |
|---|---|---|
| Epoki | 6.4 | 2.5 |

#### 1.4.3 Wnioski

Funkcja bipolarna sprawia, że algorytm zbiega znacznie szybciej niż w przypadku funkcji unipolarnej. Są dwa powody, dlaczego tak się dzieje. Po pierwsze w przypadku funkcji bipolarnej błędy klasyfikacji należą do $\{-2, 0, 2\}$, a w przypadku unipolarnej do $\{-1, 0, 1\}$, co sprawia, że aktualizacja wag jest dwukrotnie większa w przypadku funkcji bipolarnej. A jak przekonaliśmy się w poprzednim eksperymencie, zbiór danych oraz model są na tyle proste, że duża wartość współczynnika uczenia nie pogarsza czasu zbiegania. Dodatkowo podczas używania funkcji bipolarnej zbiór danych zawierał wartości $\{-1, 1\}$, co sprawiło, że środek ciężkości danych był w punkcie (0, 0). Z racji na fakt, że wartości początkowe progu są bliskie zeru, model z funkcją bipolarną był bliżej dobrego rozwiązania już na samym początku, niż model z funkcją unipolarną, gdzie środek danych był w punkcie (0.5, 0.5).

# 2 Adaline

## 2.1 Badanie wpływu początkowych wartości wag na szybkość zbieżności Adaline

### 2.1.1 Parametry eksperymentu

Tabela 9: Wartości parametrów eksperymentu

| Parametr | Wartość |
|---|---|
| Stałe eksperymentu | |
| Model | Adaline |
| Współczynnik uczenia | 0.05 |
| Funkcja progowa | Bipolarna |
| Zakres początkowych wartości progu | (-0.01, 0.01) |
| Dane treningowe | bramka logiczna AND |
| Maksymalna liczba epok | 100 |
| Tolerancja błędu | $0.3 \cdot \lvert X \rvert$ [1] |
| Ilość powtórzeń eksperymentu | 10 |
| Ziarno losowości | 0 |
| Zmienne eksperymentu | |
| Zakresy początkowych wartości wag | (-0.001, 0.001)<br>(-0.005, 0.005)<br>(-0.01, 0.01)<br>(-0.05, 0.05)<br>(-0.1, 0.1)<br>(-0.5, 0.5)<br>(-1, 1)<br>(-5, 5) |

### 2.1.2 Wyniki eksperymentu

Tabela 10: Wyniki eksperymentu

| Zakres pocz. wartości wag | $\pm 0.001$ | $\pm 0.005$ | $\pm 0.01$ | $\pm 0.05$ | $\pm 0.1$ | $\pm 0.5$ | $\pm 1$ | $\pm 5$ |
|---|---|---|---|---|---|---|---|---|
| Epoki | 8.0 | 8.0 | 8.0 | 7.9 | 7.7 | 7.9 | 9.4 | 13.5 |

[1] Rozmiar danych treningowych

### 2.1.3 Wykres wyników


### 2.1.4 Wnioski

Jak wynika z badań, gdy zakres początkowych wartości wag jest większy, model potrzebuje większej ilości epok do uzyskania dobrych rezultatów. Powód jest dokładnie ten sam jak w przypadku perceptronu tzn. dla małych wartości wag łatwiej jest znaleźć dobry stosunek wartości, gdyż różnice między wartościami są mniejsze, a prędkość aktualizacji wag jest stała.

## 2.2 Badanie wpływu wartości współczynnika uczenia na szybkość zbieżności Adaline

### 2.2.1 Parametry eksperymentu

Tabela 11: Wartości parametrów eksperymentu

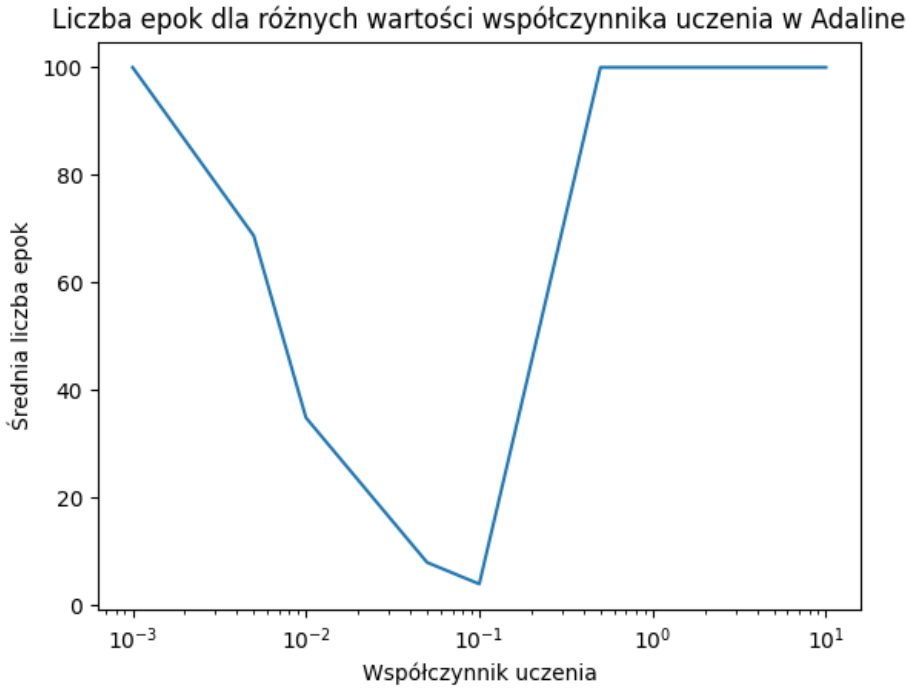
| Parametr | Wartość |
|---|---|
| Stałe eksperymentu | |
| Model | Adaline |
| Zakres początkowych wartości progu | (-0.01, 0.01) |
| Zakres początkowych wartości wag | (-0.01, 0.01) |
| Funkcja progowa | Bipolarna |
| Dane treningowe | bramka logiczna AND |
| Maksymalna liczba epok | 100 |
| Tolerancja błędu | $0.3 \cdot \lvert X \rvert$ |
| Ilość powtórzeń eksperymentu | 10 |
| Ziarno losowości | 0 |
| Zmienne eksperymentu | |
| Współczynnik uczenia | 0.001; 0.005; 0.01; 0.05; 0.1; 0.5; 1; 5; 10 |

### 2.2.2 Wyniki eksperymentu

Tabela 12: Wyniki eksperymentu

| współczynnik uczenia | 0.001 | 0.005 | 0.01 | 0.05 | 0.1 | 0.5 | 1 | 5 | 10 |
|---|---|---|---|---|---|---|---|---|---|
| Epoki | 100.0 | 68.5 | 34.9 | 8.0 | 4.0 | 100.0 | 100.0 | 100.0 | 100.0 |

### 2.2.3 Wykres wyników



### 2.2.4 Wnioski

Jak widać na wykresie zarówno zbyt mała jak i duża wartość współczynnika uczenia sprawia, że liczba potrzebnych epok znacząco rośnie, lub model wcale nie jest zbieżny. Dla małych wartości liczba epok jest duża z tego samego powodu, co w przypadku perceptronu - małe zmiany wag sprawiają, że potrzeba wiele uaktualnień, by dotrzeć do minimum. Gdy wartości współczynnika uczenia są zbyt duże adaline zachowuje się zupełnie inaczej, ponieważ wielkość zmian wag nie jest stała, ale zależna od błędu, co daje ryzyko eksplozji gradientu, czyli sytuacji, w której wartości parametrów przeskakują nad optimum i lądują po drugiej stronie z jeszcze większym błędem, co sprawia, że wartości wag rosną do nieskończoności i model jest rozbieżny. Dlatego w przypadku adaline odpowiedni wybór wartości współczynnika uczenia jest bardzo istotny.

### 2.3 Badanie wpływu wartości tolerancji błędu na szybkość zbieżności Adaline

#### 2.3.1 Parametry eksperymentu

Tabela 13: Wartości parametrów eksperymentu

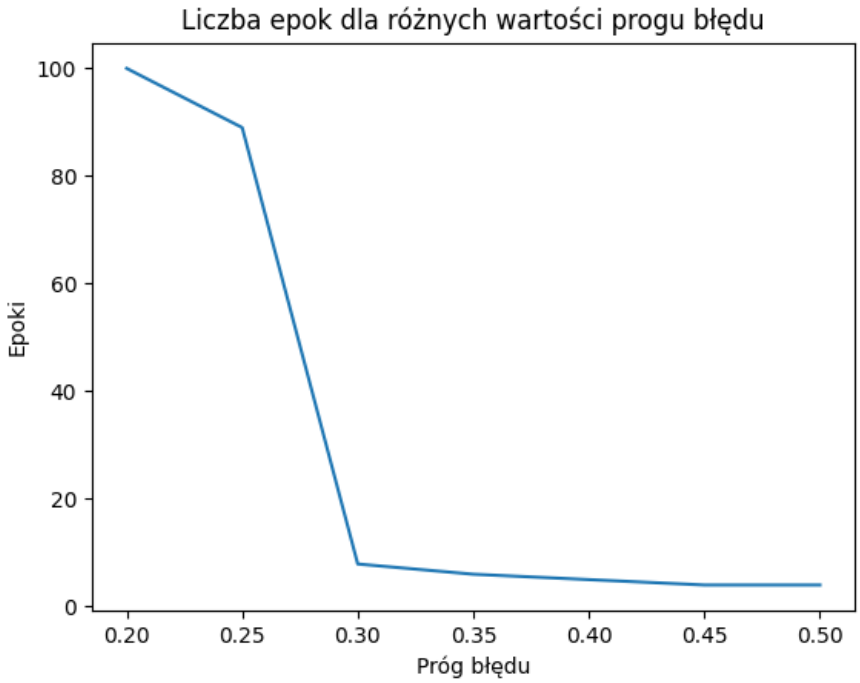
| Parametr | Wartość |
|---|---|
| Stałe eksperymentu | |
| Model | Adaline |
| Współczynnik uczenia | 0.05 |
| Zakres początkowych wartości progu | (-0.01, 0.01) |
| Zakres początkowych wartości wag | (-0.01, 0.01) |
| Funkcja progowa | Bipolarna |
| Dane treningowe | bramka logiczna AND |
| Maksymalna liczba epok | 100 |
| Ilość powtórzeń eksperymentu | 10 |
| Ziarno losowości | 0 |
| Zmienne eksperymentu | |
| Tolerancja błędu | $0.2 \cdot \lvert X \rvert$<br>$0.25 \cdot \lvert X \rvert$<br>$0.3 \cdot \lvert X \rvert$<br>$0.35 \cdot \lvert X \rvert$<br>$0.4 \cdot \lvert X \rvert$<br>$0.45 \cdot \lvert X \rvert$<br>$0.5 \cdot \lvert X \rvert$ |

#### 2.3.2 Wyniki eksperymentu

Tabela 14: Wyniki eksperymentu

| współczynnik uczenia | 0.2 | 0.25 | 0.3 | 0.35 | 0.4 | 0.45 | 0.5 |
|---|---|---|---|---|---|---|---|
| Epoki | 100.0 | 89.1 | 8.0 | 6.0 | 5.0 | 4.1 | 4.0 |

### 2.3.3 Wykres wyników




### 2.3.4 Wnioski

Dla małych wartości progu błędu, model mimo znalezienia bardzo dobrego rozwiązania dalej jest aktualizowany, co sprawia, że proces nauki jest kilkakrotnie dłuższy niż mógłby być. Dla dużych wartości progu błędu trening kończy się o wiele szybciej, jednak hiperpłaszczyzna rozdzielająca może być gorsza (tzn bliżej jednej z klas) niż w przypadku modelu o niskiej wartości progu błędu.

# 3 Porównanie modelu Perceptron i Adaline

W tej części na podstawie eksperymentu oraz intuicji porównam oba modele, a także wskaże ich dobre i słabe cechy. Na początek przeprowadzę badanie na tych samych danych i pokażę jak skuteczny jest każdy z modeli.

### 3.0.1 Parametry Perceptronu

Tabela 15: Wartości parametrów perceptronu

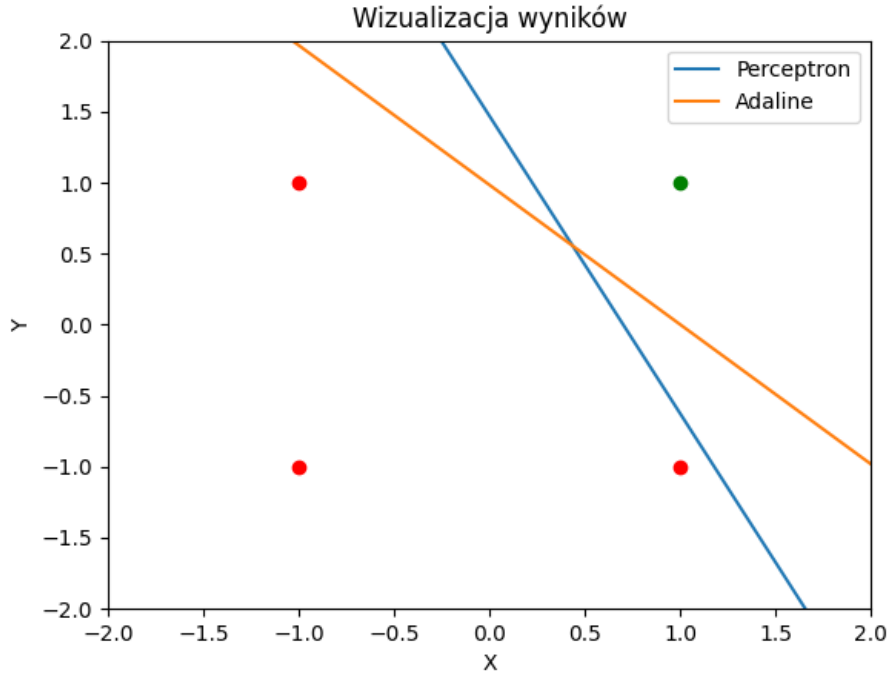
| Parametr | Wartość |
|---|---|
| Model | Perceptron |
| Współczynnik uczenia | 0.001 |
| Zakres początkowych wartości progu | (-0.01, 0.01) |
| Zakres początkowych wartości wag | (-0.01, 0.01) |
| Funkcja progowa | Bipolarna |
| Dane treningowe | bramka logiczna AND |
| Maksymalna liczba epok | 100 |
| Ilość powtórzeń eksperymentu | 1 |
| Ziarno losowości | 0 |

### 3.0.2 Parametry Adaline

Tabela 16: Wartości parametrów Adaline

| Parametr | Wartość |
|---|---|
| Model | Adaline |
| Współczynnik uczenia | 0.001 |
| Zakres początkowych wartości progu | (-0.01, 0.01) |
| Zakres początkowych wartości wag | (-0.01, 0.01) |
| Funkcja progowa | Bipolarna |
| Dane treningowe | bramka logiczna AND |
| Maksymalna liczba epok | 100 |
| Ilość powtórzeń eksperymentu | 1 |
| Ziarno losowości | 0 |

### 3.0.3 Wizualizacja hiperpłaszczyzn rozdzielających




### 3.0.4 Wnioski

Na tym eksperymencie doskonale widać słabą cechę perceptronu. Gdy znajdzie dobrą prostą, która rozdziela obie klasy, to przestaje już ją aktualizować. Wynikiem czego może być sytuacja z wykresu, na którym widać, że prosta perceptronu jest bardzo blisko punktu (1, -1). Adaline natomiast jest aktualizowane wartościami ciągłymi aż do momentu, kiedy błąd modelu jest bardzo mały (mniejszy niż próg tolerancji).
Mimo wszystko Adaline także ma pewne wady. Przede wszystkim jest bardzo wrażliwy na wartość współczynnika uczenia. Może się zdażyć, że przy zbyt dużej jego wartości model nigdy nie zbiegnie. Oprócz tego oba modele są do siebie podobne - są modelami gradientowymi o wymiarze[2] VC równym 3 (rozwiązują tylko problemy separowalne liniowo), a także podobnie reagują na zmiany niektórych hiperparametrów.

[2]Wymiar Wapnika-Czerwonienkisa