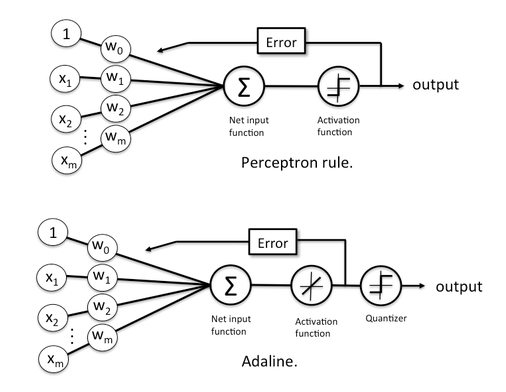
|  |  |  |
| --- | --- | --- |
| Imię Nazwisko Grupa  Michał Słowikowski  Gr 4 | Temat  Scenariusz 1 | Data  10.11.2017r. |

Celem ćwiczenia było nauczenie jednowarstwowej sieci neuronowej do rozpoznawania dużych i małych liter

Do ćwiczenia wykorzystałem perceptron oraz Adaptive Linear Neuron czyli Adaline.

**Syntetyczny opis algorytmu uczenia**

Do wykonania ćwiczenia w pierwszej kolejności wykorzystałem sieć neuronową składającą się tylko z jednego perceptronu (wymóg jednej warstwy + chcę sprawdzać tylko czy to litera duża czy mała). Perceptron podczas nauki wykorzystywał perceptron rule.

W drugim przypadku podobnie wykorzystałem sieć składającą się z jednego neuronu Adaline wykorzystujący gradient descent.

Perceptron:

Został zbudowany zgodnie z modelem podanym na wykładzie. Posiada własne wagi oraz bias = 1.  
Metody publiczne:  
-GetResult: Przyjmuje dane wejściowe, zwraca odpowiedź (1 lub 0)

-Learn: Zajmuje się nauką neuronu. Modyfikuje wagi.

Adaline:  
Budową przypomina perceptron, jednak ma jedną zasadniczą różnicę. Błąd jest obliczany przed właściwą funkcją aktywacji (wcześniej jest tylko funkcja liniowa). Dlatego błąd obliczamy na rzeczywistych wynikach.

Metody publiczne:

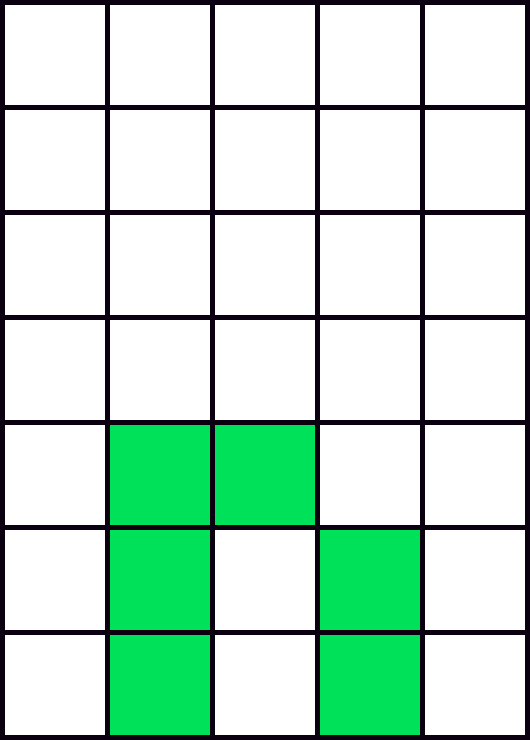
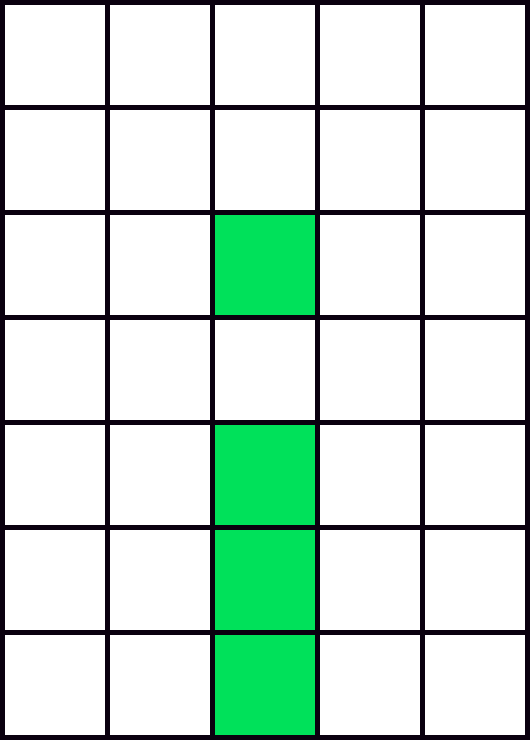
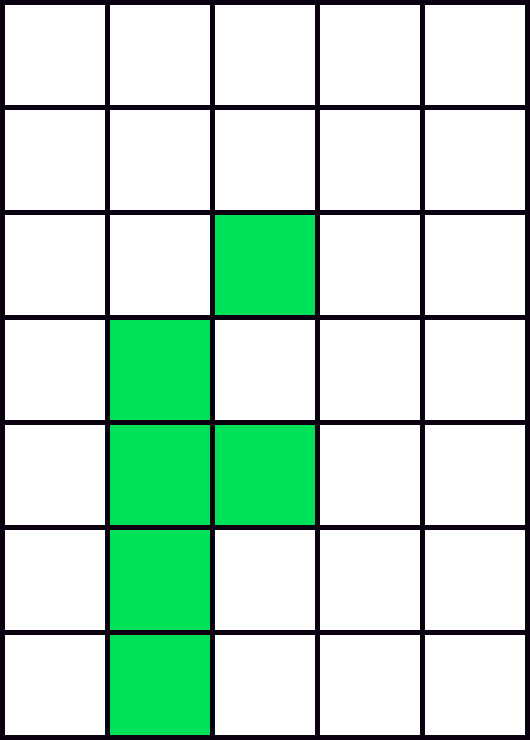
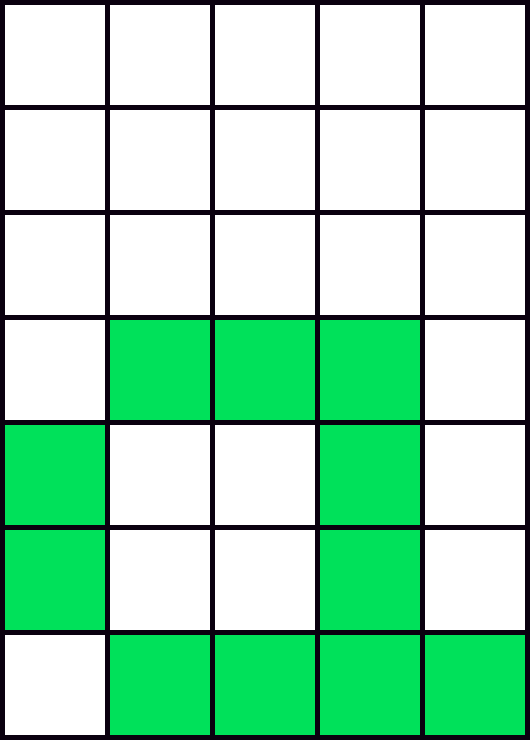
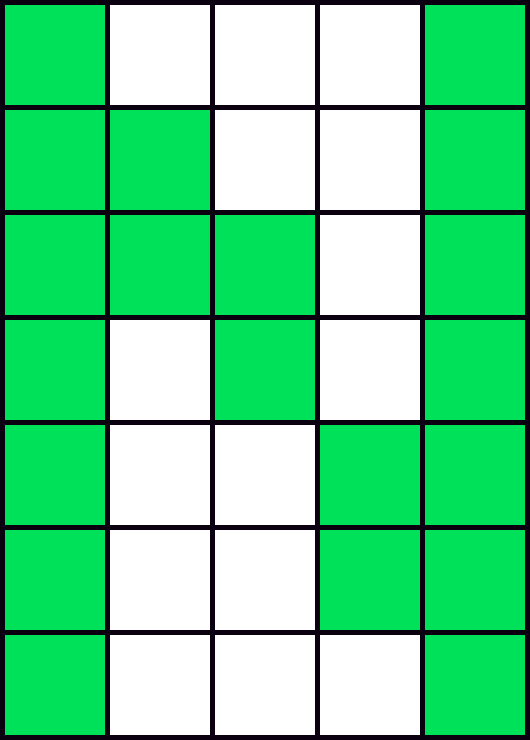
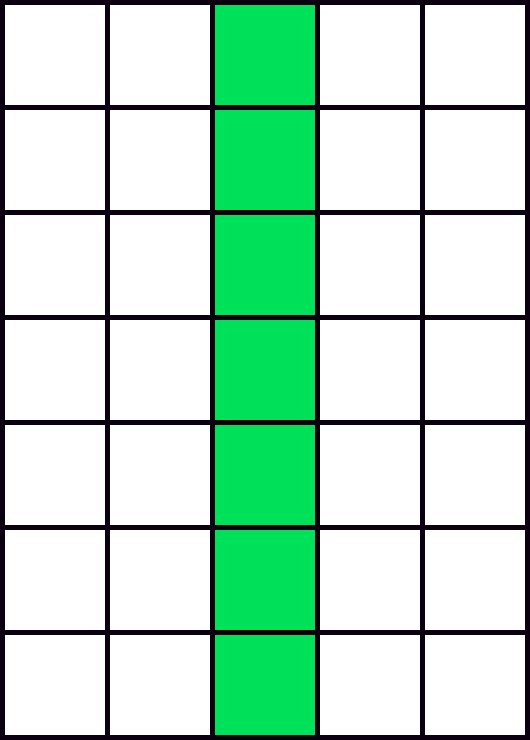
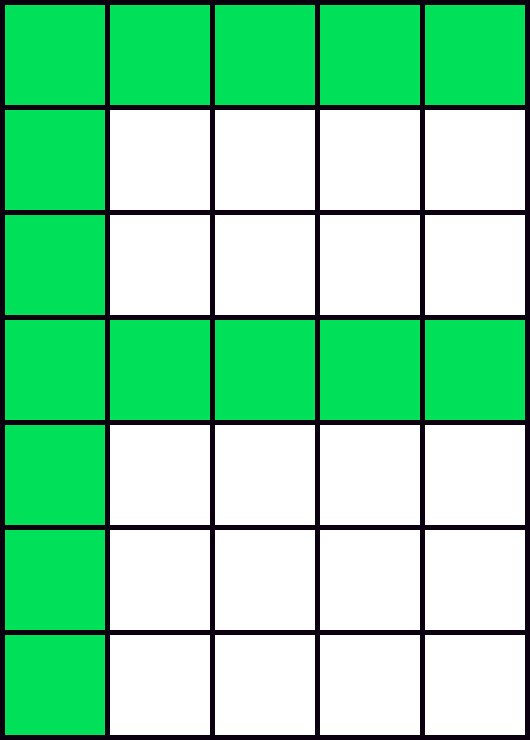
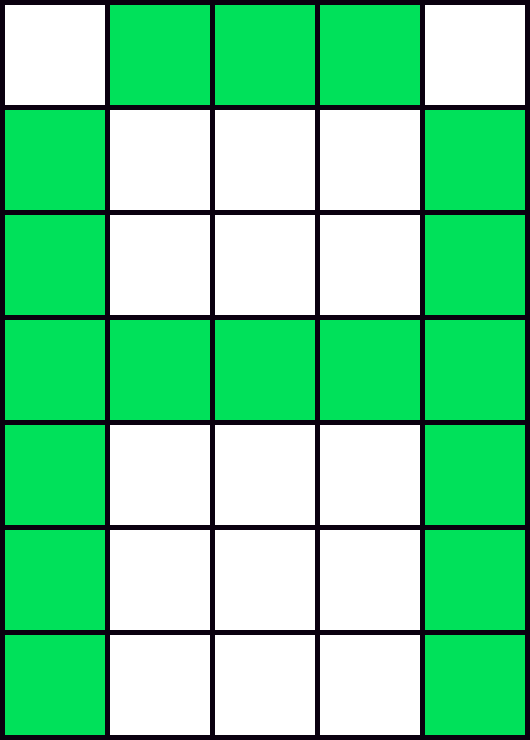
-GetResult: Przyjmuje dane wejściowe, zwraca faktyczny wynik (przed funkcją aktywacji)

-Test: Zwraca dane, które przeszły przez funkcję aktywacji.(-1 lub 1)

-Learn: Zajmuje się nauką neuronu. Modyfikuje wagi.

W obydwu przypadkach zastosowano progową funkcję aktywacji (albo litera jest mała, albo duża).

**Zestaw danych testowych**

Zgodnie z propozycją przedstawioną na zajęciach danymi wejściowymi wykorzystywane podczas nauki były tablice o wymiarach 7x5. Wykorzystałem 20 liter (10 dużych, 10 małych). Poniżej obrazy (biały - 0, zielony - 1)  


(przykładowe litery)

**Testy**

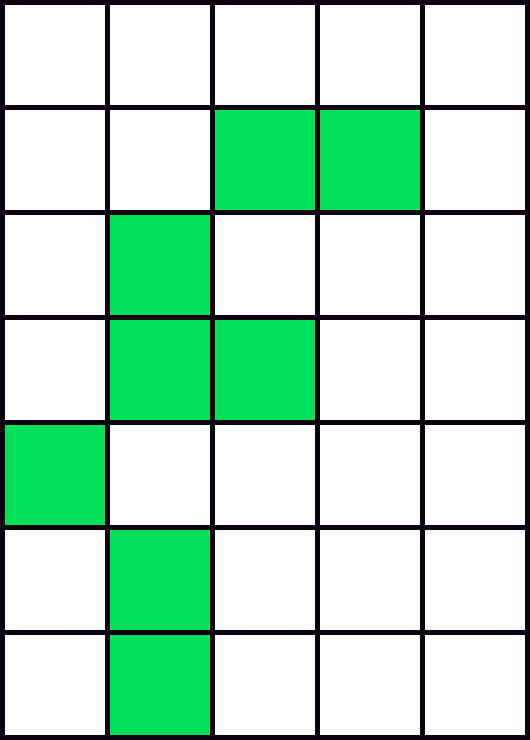
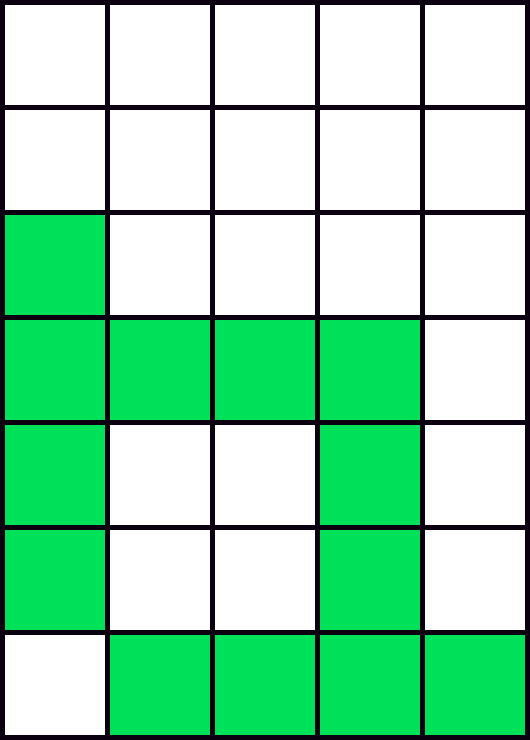
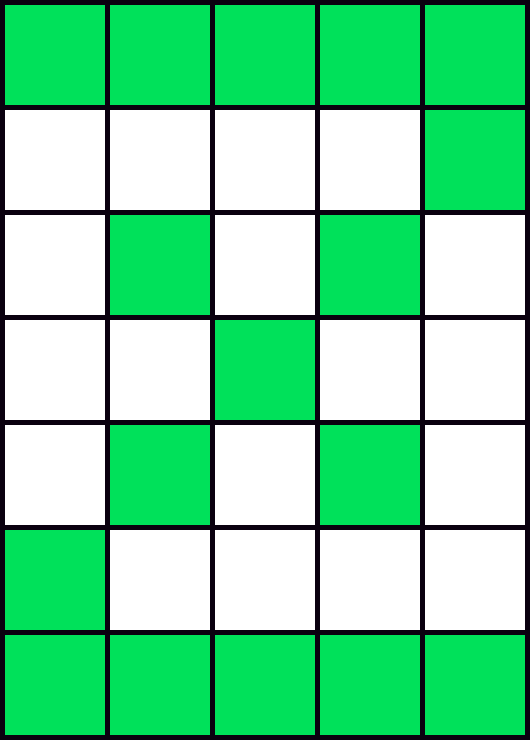
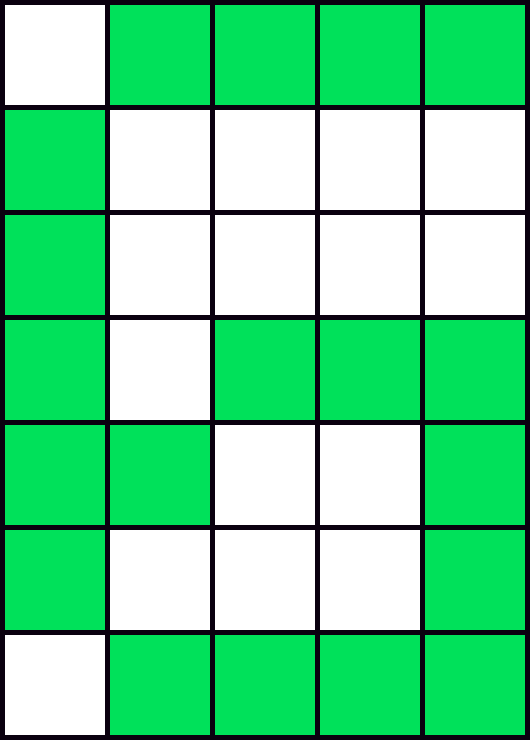
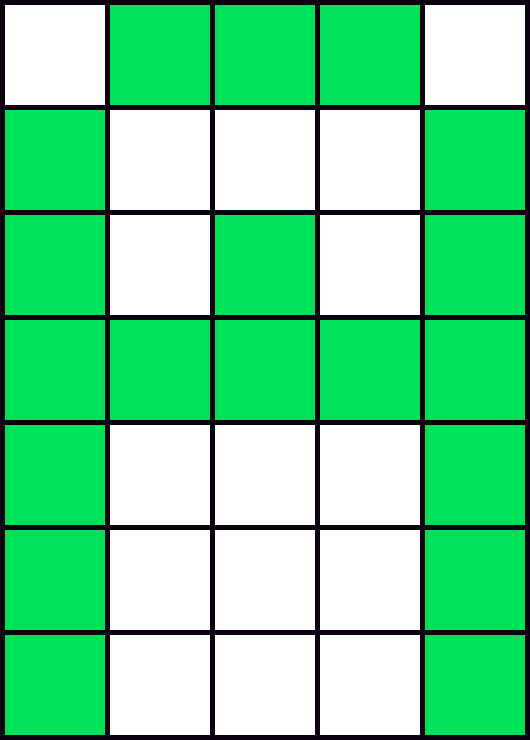
Testy zostały przeprowadzane ze względu na 2 kryteria:

-współczynnik uczenia się

-ilość obszarów

Obszary były wyznaczane poprzez podzielenie tablicy na wycinki. Jeżeli w danym wycinku znajdował się zakolorowany piksel, wtedy wycinek zwraca wartość 1, jeżeli wszystkie piksele były białe, zwracał wartość 0.

Na koniec (po zakończeniu uczenia) neurony były poddawane jeszcze testowi, w którym litery były zaszumiane (niektóre piksele miały złe wartości). Ten zestaw danych nie był wykorzystywany podczas uczenia.

  
  
Wyniki:

Dla 1 fragmentu nie udało się ukończyć nauki pozytywnie, dla żadnego współczynnika uczenia.

(Dla wszystkich testów 50% trafionych).

Dla 9 fragmentów (3X i 3Y)

Nie udało się pozytywnie zakończyć nauki w żadnym przypadku.

Współczynnik uczenia 0.001.

|  |  |
| --- | --- |
| Perceptron  1 epoka - 50%  51 epoka - 70%  52 epoka - 80%  72 epoka - 90%  Wynik testu - 50% (wszystkie 0) | Adaline  1 epoka - 50%  13 epoka - 65%  14 epoka - 75%  15 epoka - 80%  Wynik testu - 75% |

Dla współczynnika 0.01

|  |  |
| --- | --- |
| Perceptron  1 epoka - 75%  2 epoka - 80%  7 epoka - 95%  11 epoka - 90%  Wynik testu - 50% (wszystkie 0) | Adaline  1 epoka - 50%  3 epoka - 60%  4 epoka - 75%  5 epoka - 80%  Wynik testu - 75% |

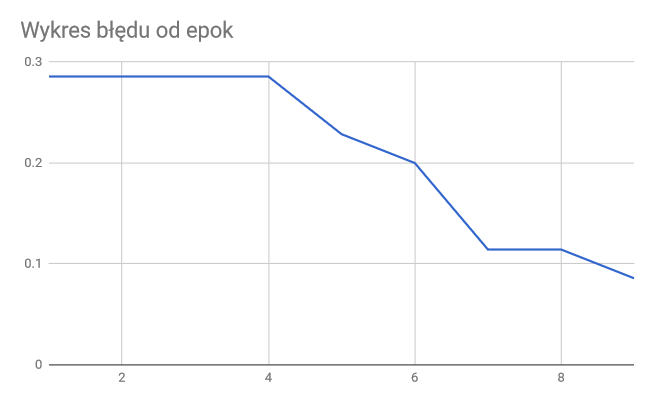
Dla współczynnika 0.01

|  |  |
| --- | --- |
| Perceptron  1 epoka - 75%  2 epoka - 90%  3 epoka - 85%  4 epoka - 90%  Wynik testu - 50% (wszystkie 0) | Adaline  1 epoka - 90%  30 epoka - 95%  Wynik testu - 75% |

Dla 35 fragmentów (5X i 7Y)

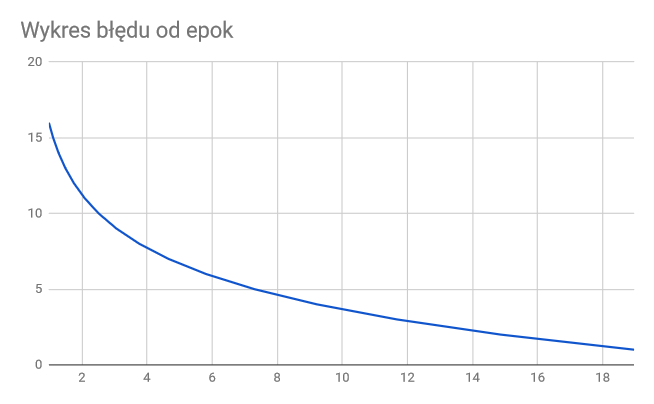
Współczynnik uczenia 0.001

Perceptron



Wynik testu - 85%

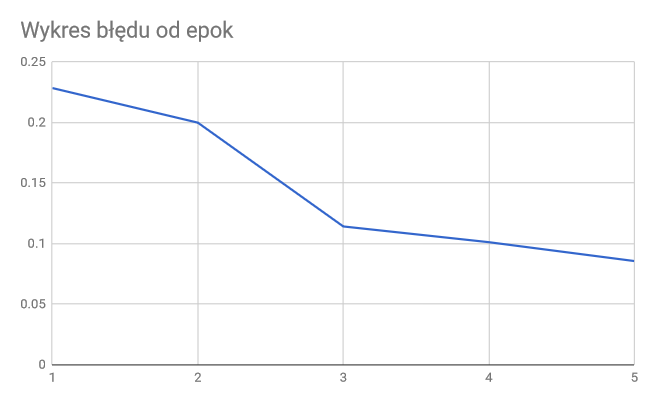
Adaline



Wynik testu 80%

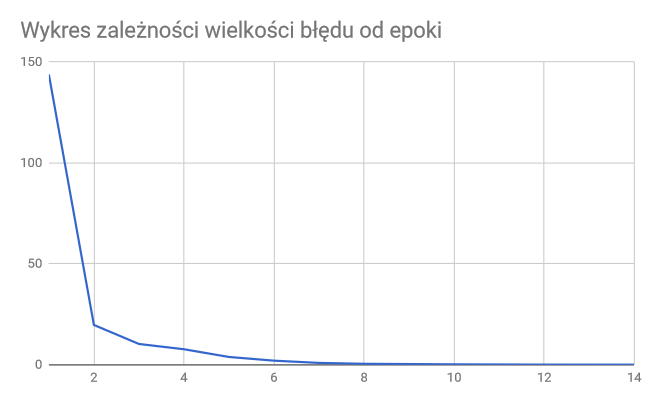
Współczynnik uczenia 0.01

Perceptron



Wynik Testu - 85%

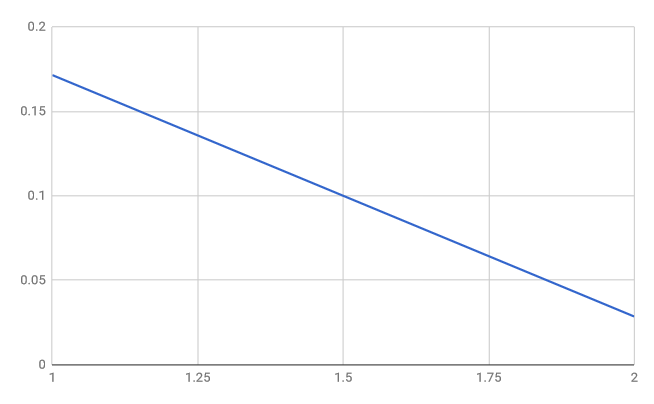
Adaline



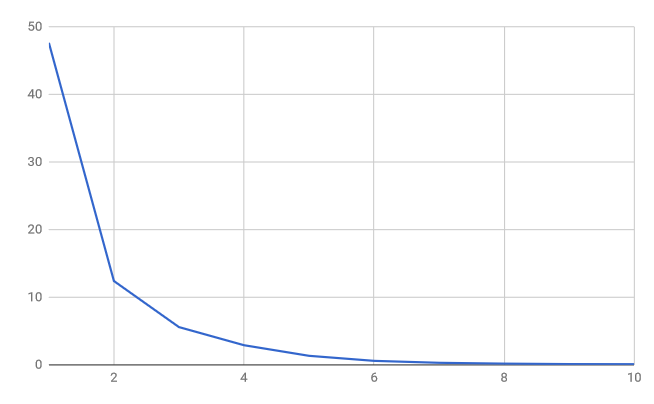
Wynik Testu - 95%

Współczynnik uczenia 0.1

Perceptron



Wynik Testu 90%

Adeline

Wynik Testu 90%