

Programowanie Sieciowe

Perceptron, Adeline

Imię i nazwisko: Dominik Ćwikowski

Indeks: 248914

Data zajęć: 10.03.2021

1. Opis ćwiczenia

Ćwiczenie ma na celu poznanie i zrozumienie działania prostej sieci neuronowej, i zaobserwowanie otrzymanych wyników.

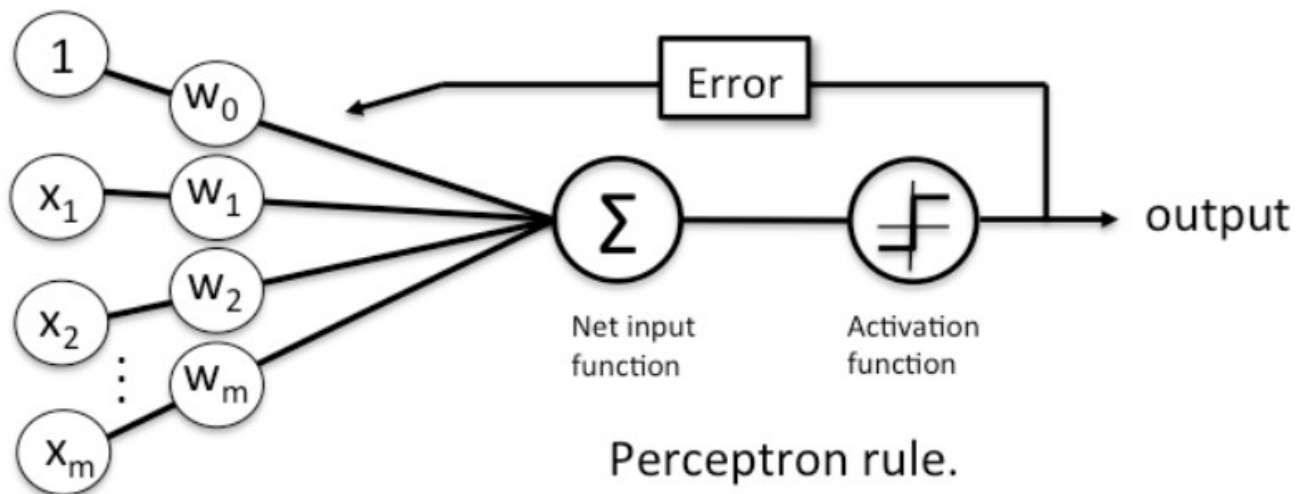
2. Opis rozwiązania problemu

Zasada działania Perceptronu opiera się na wymnażaniu wektora wartości wejściowych $x_j^{(i)}$ dla każdej próbki (i) przez wektor wag w_j , następnie dzięki przepuszczeniu przez funkcję aktywacji utrzymuje się wartość wyjściową $output(i)$. Poźniej należy zmodyfikować wartości wag o wartość Δw_j obliczoną ze wzoru:

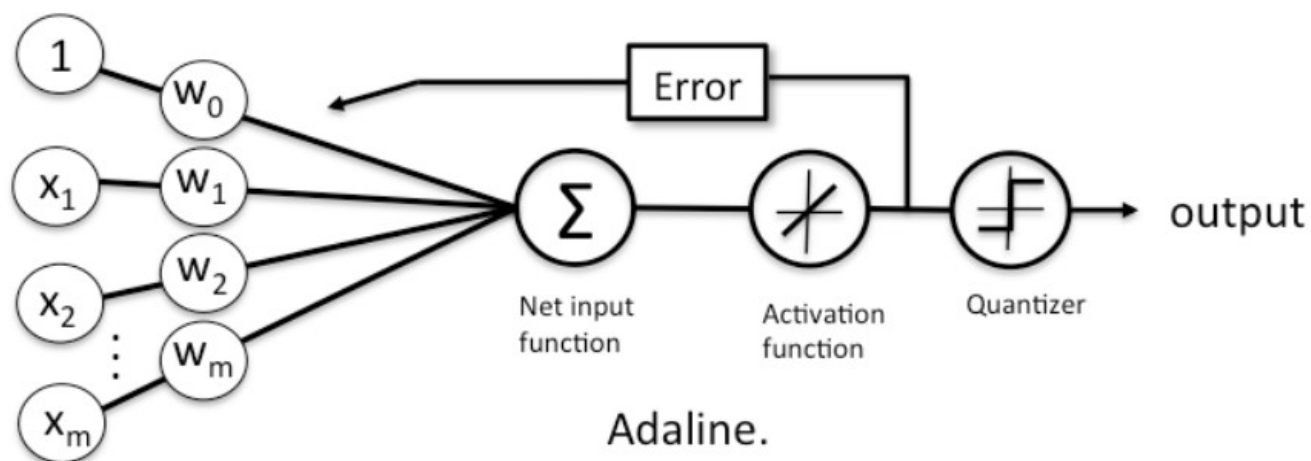
$$\Delta w_j = \eta (target^{(i)} - output^{(i)}) x_j^{(i)}$$

$target(i)$ - wartość oczekiwana

η - współczynnik uczenia się (w naszych badaniach równe 0.01)



W przeciwieństwie do perceptronu, Adaline aktualizuje wagi w oparciu o liniową funkcję aktywacji, a nie funkcję kroku jednostkowego. Aktywacja liniowa jest ulepszeniem perceptronu.



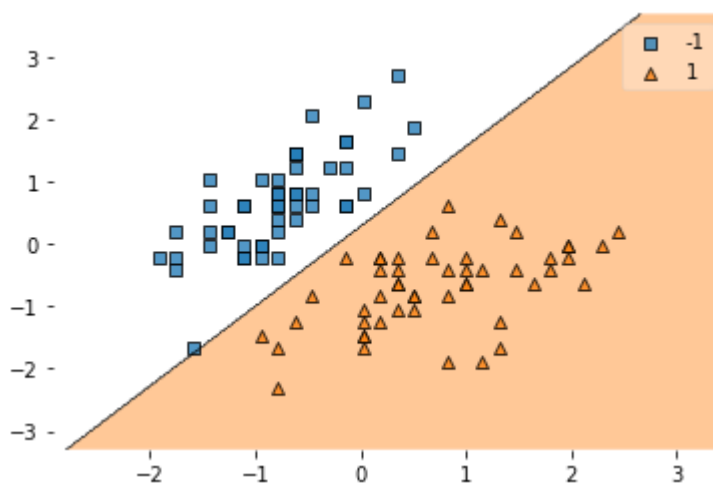
3. Testy

Testy były wykonywane na najbardziej znanej bazie danych wykorzystywanej w dziedzinie uczenia maszynowego "iris.data". Dla perceptronu 2 klasowego podajemy dane do dwóch klasyfikatorów, które następnie są rozdzielane na dane do uczenia (80%) i dane do testu (20%)

Perceptron dla 2 klas

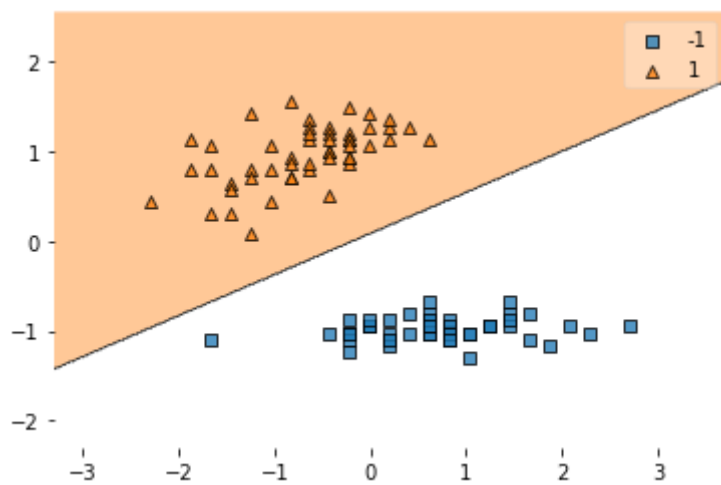
Testy klasyfikatora korzystającego z dwóch klas, wykonywane na zbiorach:

- Iris Setosa, Iris Versicolor, dwie pierwsze kolumny, 100 iteracji:



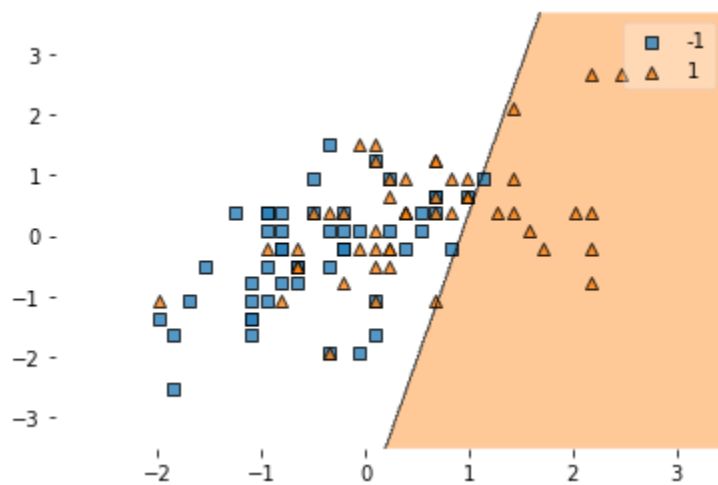
Niepasujące próbki: 0

- Iris Setosa, Iris Versicolor, druga trzecia kolumna, 100 iteracji:



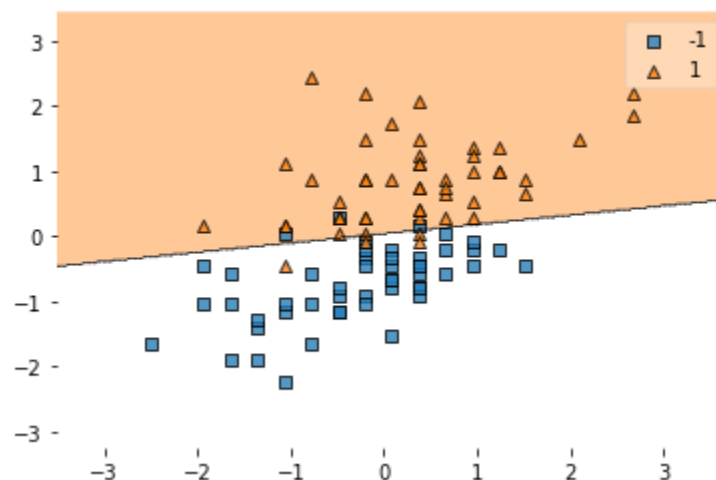
Niepasujące próbki: 0

- Iris Versicolor, Iris Virginica, dwie pierwsze kolumny, 100 iteracji:



Niepasujące próbki: 8

- Iris Versicolor, Iris Virginica, druga trzecia kolumna, 100 iteracji:



Niepasujące próbki: 2

Klasyfikacja przy użyciu perceptronu 3 klasowego, polega na podaniu danej do wszystkich trzech klasyfikatorów i wybraniu tej klasy gdzie wskazanie pozytywne jest największe.

Badane przykłady:

- Wszystkie kwiaty, dwie pierwsze kolumny, 100 iteracji:

```
1.Niepasujące próbki: 0
2.Niepasujące próbki: 7
3.Niepasujące próbki: 8
```

- Wszystkie kwiaty, dwie pierwsze kolumny, 50 iteracji:

```
1.Niepasujące próbki: 0
2.Niepasujące próbki: 7
3.Niepasujące próbki: 8
```

- Wszystkie kwiaty, wszystkie kolumny, 100 iteracji:

```
1.Niepasujące próbki: 0
2.Niepasujące próbki: 7
3.Niepasujące próbki: 1
```

- Wszystkie kwiaty, wszystkie kolumny, 50 iteracji:

```
1.Niepasujące próbki: 0
2.Niepasujące próbki: 7
3.Niepasujące próbki: 1
```

Analogicznie badamy Adaline.

Badane przykłady:

- Wszystkie kwiaty, dwie pierwsze kolumny, 100 iteracji:

```
1.Niepasujące próbki: 1  
2.Niepasujące próbki: 7  
3.Niepasujące próbki: 27
```

- Wszystkie kwiaty, dwie pierwsze kolumny, 50 iteracji:

```
1.Niepasujące próbki: 1  
2.Niepasujące próbki: 7  
3.Niepasujące próbki: 27
```

- Wszystkie kwiaty, wszystkie kolumny, 100 iteracji:

```
1.Niepasujące próbki: 1  
2.Niepasujące próbki: 2  
3.Niepasujące próbki: 4
```

- Wszystkie kwiaty, wszystkie kolumny, 50 iteracji:

```
1.Niepasujące próbki: 1  
2.Niepasujące próbki: 7  
3.Niepasujące próbki: 11
```

4. Wnioski

Na podstawie przeprowadzonych badań można wywnioskować, że jakość przeprowadzanych klasyfikacji rośnie wraz z:

- Liczbą danych do treningu,
- Liczbą klas,
- Ilością iteracji,
- Im “dalej” znajdują się od siebie dane dla poszczególnych parametrów (np. Długość płatk 5cm dla pierwszej klasy i 2cm dla drugiej).

Także od pewnej liczby iteracji wyniki przestają ulegać polepszeniu

Dla większej ilości danych wejściowych Adaline dokonuje klasyfikacji lepiej od Perceptronu.