**Sprawozdanie –Scenariusz2**

**Temat:** Budowa i działanie sieci jednowarstwowej.

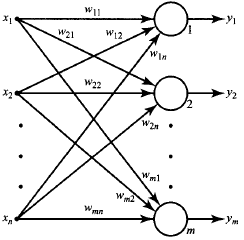
**Cel:** Celem ćwiczenia jest poznanie budowy i działania jednowarstwowych sieci neuronowych oraz uczenie rozpoznawania wielkości liter.

1. **Dane uczące i testujące**

Dane uczące to litery zapisane w specjalny sposób. Wymiar matrycy dla pojedynczej litery to 7x5. Każde pole ma swoją wartość, są to liczby 0 lub 1, w zależności czy w polu jest litera czy nie. W moim programie jedna litera to ciąg 35 zer lub jedynek w jednym wierszu tablicy, co daje wymiar 10x35, dla 10 różnych liter. Dane uczące to także ciąg liczb 0 i 1, gdzie wartość 1 odpowiada testowanej literze, a pozostałe 25 wartości równa jest 0. Moje dane testujące to tablica o wymiarze 10x26, gdzie w każdym z dziesięciu wierszy znajduje się ciąg zer i jedynek, odpowiadający kolejnej literze w alfabecie. Szczegółowy opis danych uczących i testujących znajduje się w osobnym pliku.

1. **Uczenie sieci jednowarstwowej**

Sieć jednowarstwowa tworzą neurony ułożone w jednej warstwie. W swoim programie wykorzystałem dwa algorytmy uczenia sieci. W obu podejściach dostosowałem się do schematu pokazanego na obrazku:



Suma zaś została wyliczona na podstawie powyższego wzoru , po czym była przekazywana do funkcji aktywacji dla każdego z 26 sztucznych neuronów.

Pierwszy algorytm obliczał wagi w sposób:

*wi += u \** (*d-y*) *\* xi*

*Θ= Θ-(d-y),*

gdzie u jest niewielkim współczynnikiem uczenia (u > 0), d -oczekiwana odpowiedź, y -odpowiedź neuronu.

Drugi algorytm obliczał wagi w podobny sposób z tym że do powyzszego wzoru dodoatkowo dodawana była różnica wag obecej oraz poprzedniej pomnozona przez zmienną momentum.

1. **Zestawienie wyników**

Przykładowa zmiana wag litery B (np. zmiana wag wartości wejściowej x o ineksie 1)

Dla algorytmu 1, współczynnik uczenia = 0,5

|  |  |
| --- | --- |
| **Wagi przed** | **Wagi po** |
| Waga(1,0)=0.1565893200143741  Waga(1,1)=0.6399146617909677  Waga(1,2)=0.3147413590640138  Waga(1,3)=0.8957136892178424  Waga(1,4)=0.8467447037452872  Waga(1,5)=0.3451998616532711  Waga(1,6)=0.7514955946007834  Waga(1,7)=0.8198364477080263  Waga(1,8)=0.8492453979512035  Waga(1,9)=0.11176608310771163  Waga(1,10)=0.2737129051408145  Waga(1,11)=0.6462371719632962  Waga(1,12)=0.7911244045993152  Waga(1,13)=0.5106743572594223  Waga(1,14)=0.9193951781204864  Waga(1,15)=0.9541950802930366  Waga(1,16)=0.6303009008574919  Waga(1,17)=0.14908249159677667  Waga(1,18)=0.5084338876613089  Waga(1,19)=0.37734052398065054  Waga(1,20)=0.5983475359043218  Waga(1,21)=0.8328966608351952  Waga(1,22)=0.29654072103789186  Waga(1,23)=0.015818831082587193  Waga(1,24)=0.8044222226400697  Waga(1,25)=0.7322391365475803  Waga(1,26)=0.13435156704928997  Waga(1,27)=0.13631969442974456  Waga(1,28)=0.0011269838470991944  Waga(1,29)=0.0829658714634115  Waga(1,30)=0.19936194863794432  Waga(1,31)=0.4671315876701069  Waga(1,32)=0.09215860251606589  Waga(1,33)=0.7631317960537147  Waga(1,34)=0.46307734990275795 | Waga(1,0)=0.6565893200143741  Waga(1,1)=1.1399146617909677  Waga(1,2)=0.8147413590640138  Waga(1,3)=1.3957136892178426  Waga(1,4)=0.8467447037452872  Waga(1,5)=0.8451998616532711  Waga(1,6)=0.7514955946007834  Waga(1,7)=0.8198364477080263  Waga(1,8)=0.8492453979512035  Waga(1,9)=0.6117660831077116  Waga(1,10)=0.7737129051408145  Waga(1,11)=0.6462371719632962  Waga(1,12)=0.7911244045993152  Waga(1,13)=0.5106743572594223  Waga(1,14)=1.4193951781204865  Waga(1,15)=1.4541950802930366  Waga(1,16)=1.1303009008574918  Waga(1,17)=0.6490824915967767  Waga(1,18)=1.008433887661309  Waga(1,19)=0.37734052398065054  Waga(1,20)=1.0983475359043218  Waga(1,21)=0.8328966608351952  Waga(1,22)=0.29654072103789186  Waga(1,23)=0.015818831082587193  Waga(1,24)=1.3044222226400697  Waga(1,25)=1.2322391365475802  Waga(1,26)=0.13435156704928997  Waga(1,27)=0.13631969442974456  Waga(1,28)=0.0011269838470991944  Waga(1,29)=0.5829658714634115  Waga(1,30)=0.6993619486379443  Waga(1,31)=0.9671315876701069  Waga(1,32)=0.5921586025160659  Waga(1,33)=1.2631317960537147  Waga(1,34)=0.46307734990275795 |

Przykładowa zmiana wag litery B (np. zmiana wag wartości wejściowej x o ineksie 1)

Dla algorytmu 2, współczynnik uczenia = 0,5, momentum=0,1

|  |  |
| --- | --- |
| **Wagi przed** | **Wagi po** |
| Waga(1,0)=0.9806256990428082  Waga(1,1)=0.33538034668179406  Waga(1,2)=0.8080216174264792  Waga(1,3)=0.057291164781185455  Waga(1,4)=0.6702999711256926  Waga(1,5)=0.5650174598526088  Waga(1,6)=0.08447848292013505  Waga(1,7)=0.29434602789878284  Waga(1,8)=0.3377780529848382  Waga(1,9)=0.3048537053427143  Waga(1,10)=0.9776041667682287  Waga(1,11)=0.2586853504881358  Waga(1,12)=0.9170810535110856  Waga(1,13)=0.19001761258876237  Waga(1,14)=0.8116422817807155  Waga(1,15)=0.48460771490276855  Waga(1,16)=0.9063455789294776  Waga(1,17)=0.6825563988480332  Waga(1,18)=0.8424314758271021  Waga(1,19)=0.5098371006388671  Waga(1,20)=0.7196023801037146  Waga(1,21)=0.7487370719830209  Waga(1,22)=0.952598729546287  Waga(1,23)=0.7532220992195054  Waga(1,24)=0.4425550634186761  Waga(1,25)=0.1971000661337018  Waga(1,26)=0.25162270413491694  Waga(1,27)=0.8710737358248466  Waga(1,28)=0.16035852187050215  Waga(1,29)=0.3953129984339133  Waga(1,30)=0.17894216938546137  Waga(1,31)=0.7286470291057056  Waga(1,32)=0.6435601778550641  Waga(1,33)=0.49109972393491397  Waga(1,34)=0.967073659898836 | Waga(1,0)=1.4806256990428082  Waga(1,1)=0.8353803466817941  Waga(1,2)=1.3080216174264792  Waga(1,3)=0.5572911647811855  Waga(1,4)=0.6702999711256926  Waga(1,5)=1.0650174598526088  Waga(1,6)=0.08447848292013505  Waga(1,7)=0.29434602789878284  Waga(1,8)=0.3377780529848382  Waga(1,9)=0.8048537053427143  Waga(1,10)=1.4776041667682287  Waga(1,11)=0.2586853504881358  Waga(1,12)=0.9170810535110856  Waga(1,13)=0.19001761258876237  Waga(1,14)=1.3116422817807156  Waga(1,15)=0.9846077149027685  Waga(1,16)=1.4063455789294776  Waga(1,17)=1.1825563988480332  Waga(1,18)=1.342431475827102  Waga(1,19)=0.5098371006388671  Waga(1,20)=1.2196023801037146  Waga(1,21)=0.7487370719830209  Waga(1,22)=0.952598729546287  Waga(1,23)=0.7532220992195054  Waga(1,24)=0.9425550634186761  Waga(1,25)=0.6971000661337018  Waga(1,26)=0.25162270413491694  Waga(1,27)=0.8710737358248466  Waga(1,28)=0.16035852187050215  Waga(1,29)=0.8953129984339133  Waga(1,30)=0.6789421693854614  Waga(1,31)=1.2286470291057054  Waga(1,32)=1.1435601778550641  Waga(1,33)=0.991099723934914  Waga(1,34)=0.967073659898836 |

Wynik działania programu dla zmodyfikowanych wag po uczeniu dla litery B

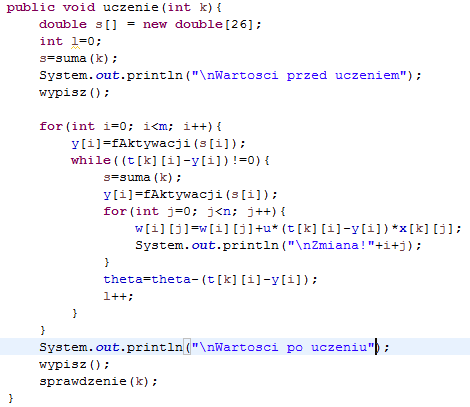
|  |  |
| --- | --- |
| Algorytm1 | Algorytm2 |
| Odpowiedz po uczeniu:  0,  1,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0, | Odpowiedz po uczeniu:  0,  1,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0, |

1. **Wnioski**

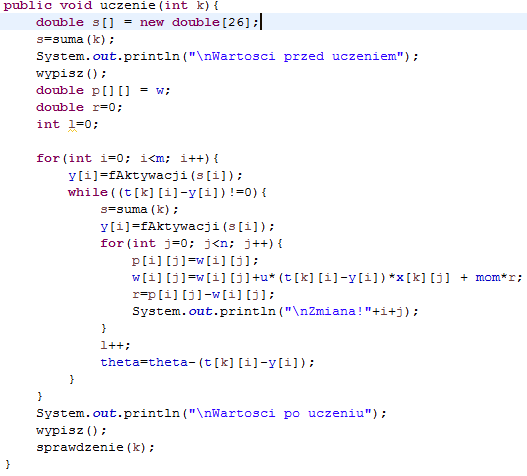
Działanie obu algorytmów różniło się w większości przypadków nieznacznie. Wpływ na działanie miała zarówno liczba momentum jak i wsółczynnik uczenia. Przy niewielkiej liczbie momentum np. momentum=0,1 i takich samych współczynnikach uczenia dla obu algorytmów, szybkość działania, czyli liczba iteracji nie różniła się znacząco. W przypadku jednak, gdy liczba momentum była większa np. momentum=1, szybszy okazywał się algorytm wykorzystujący momentum. Wniosek jest taki, że zastosowanie tej zmiennej przyspiesza uczenie. Im większa jest wartość tego współczynnika , tym składnik z niego wynikający ma większy wpływ na dobór wag. Jeśli chodzi o współczynnik uczenia, to jego mniejsza wartość powodowała zwiększenie ilości iteracji. W tym przypadku uczenia sieci efektywnie byłoby zastosować zarówno większy współczynnik uczenia jak i współczynnik momentum. Algorytm drugi wydaje się być lepszym wyjściem.

1. **Kod programu**

**Algorytm1**



**Algorytm2**



**Sprawdzenie, sumowanie oraz funkcja aktywacji**

