

# Metody Inteligencji Obliczeniowej - Sprawozdanie lab 1

Karol Hoerner de Roithberg

13 Marzec 2023

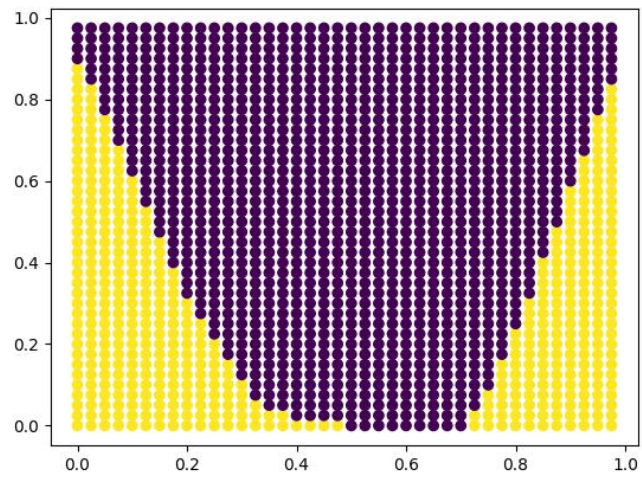
## 1 Analiza nowego leku za pomocą sieci neuronowych

W pierwszym zadaniu należało znormalizować oraz podzielić na uczące oraz trenujące dane zawierające analizę nowego leku zachowując proporcje klas, następnie zaproponowano sieć neuronową oraz zbadano jej skuteczność w zależności od ilości warstw oraz ilości neuronów w tych warstwach.

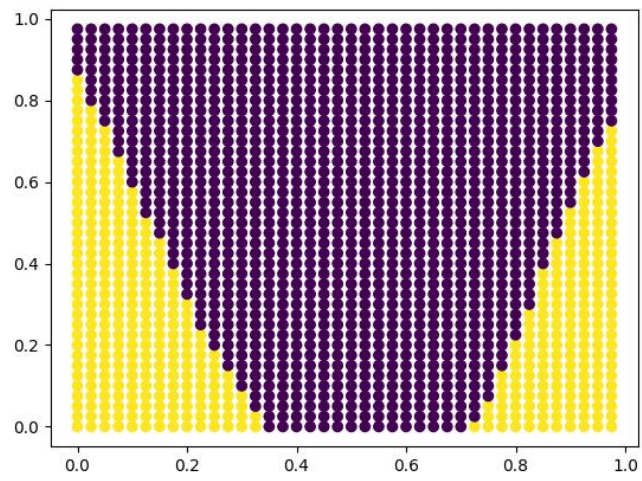
Ilość neuronów w kolejnych warstwach	Wynik klasyfikacji
[7, 29, 40, 9]	0.91
[24, 22, 15]	0.89
[10, 18, 22]	0.89
[37, 6, 39]	0.92
[10, 17, 31]	0.94
[27, 15]	0.90
[5, 29, 31]	0.92
[14, 38, 5, 36]	0.92

Tabela 1: Caption

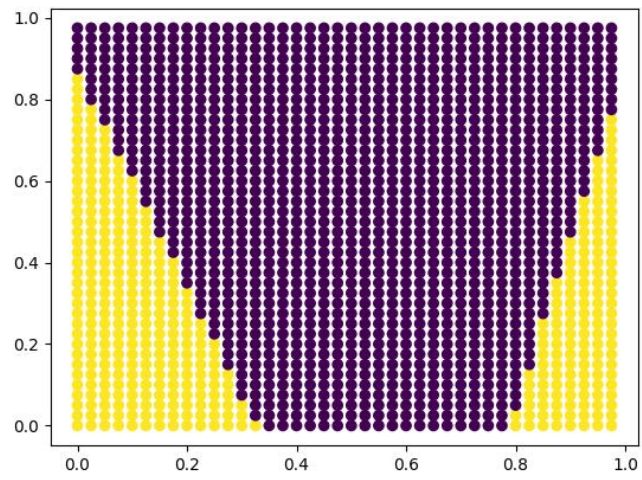
Wyniki klasyfikacji przedstawiono na wykresach poniżej.



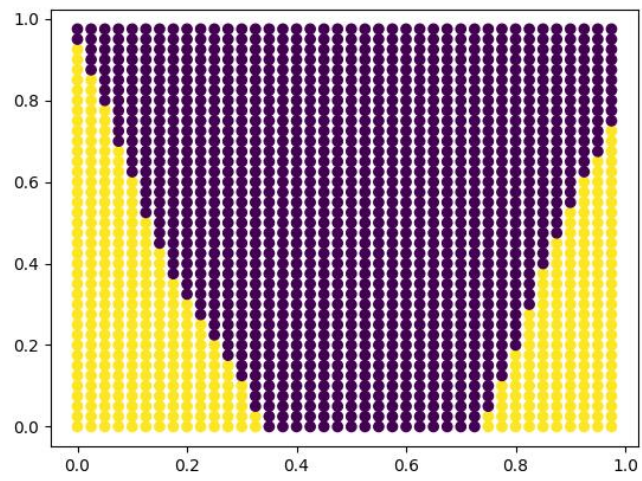
Rysunek 1: Wykres klasyfikacji dla sieci neuronowej [7, 29, 40, 9]



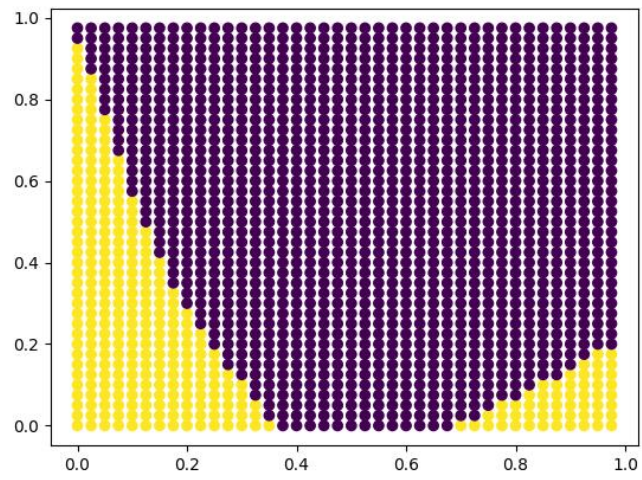
Rysunek 2: Wykres klasyfikacji dla sieci neuronowej [24, 22, 15]



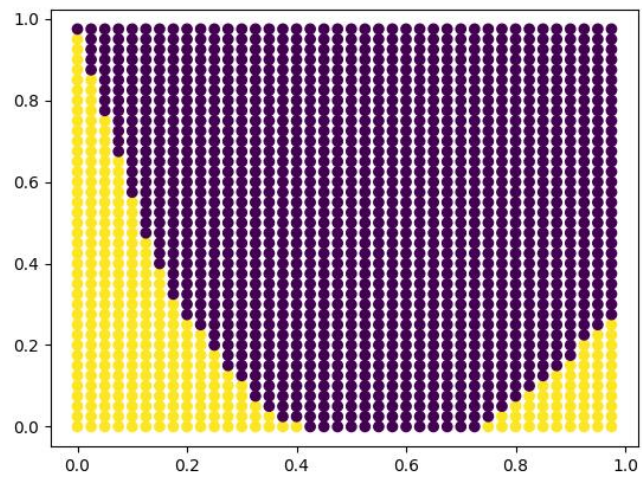
Rysunek 3: Wykres klasyfikacji dla sieci neuronowej [10, 18, 22]



Rysunek 4: Wykres klasyfikacji dla sieci neuronowej [37, 6, 39]

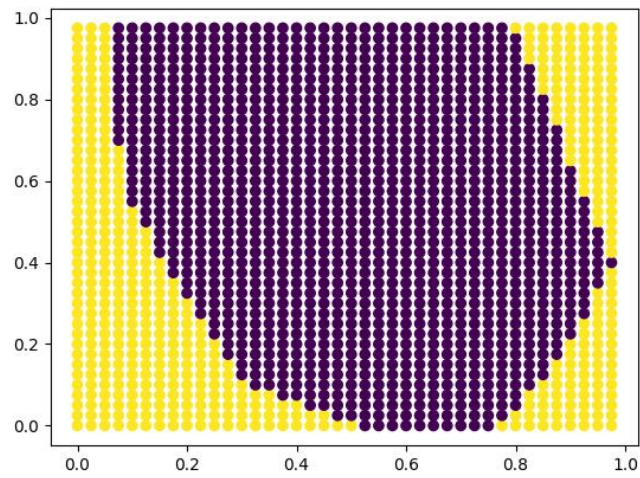


Rysunek 5: Wykres klasyfikacji dla sieci neuronowej [10, 17, 31]

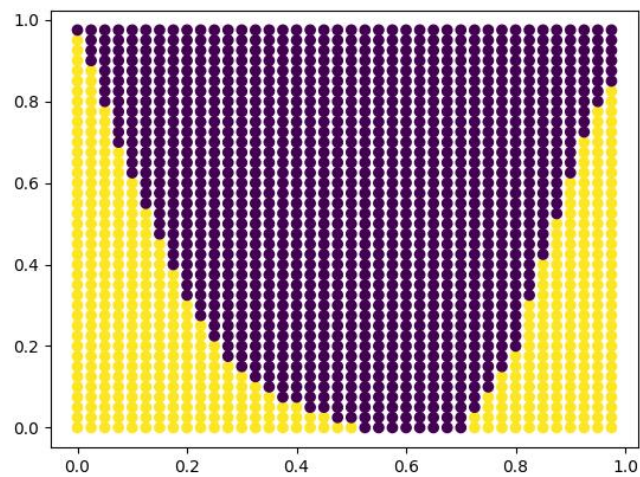


Rysunek 6: Wykres klasyfikacji dla sieci neuronowej [27, 15]





Rysunek 7: Wykres klasyfikacji dla sieci neuronowej [5, 29, 31]



Rysunek 8: Wykres klasyfikacji dla sieci neuronowej [14, 38, 5, 36]

## 1.1 Wnioski

- Wynik klasyfikacji jest dokładniejszy dla większej liczby neuronów oraz warstw
- Aby otrzymać optymalną strukturę należy przetestować wiele opcji sieci
- Normalizacja danych jest konieczna aby sieć neuronowa otrzymała dobre wyniki

## 2 Analiza zbioru ręcznie pisanych cyfr za pomocą sieci neuronowych

W kolejnym zadaniu pobralno zbiór ręcznie pisanych cyfr oraz dokonano jej klasyfikacji z opcjami sieci neuronowych jak w zadaniu pierwszym.

Ilość neuronów w kolejnych warstwach	Wynik klasyfikacji
[7, 29, 40, 9]	0.93
[24, 22, 15]	0.87
[10, 18, 22]	0.95
[37, 6, 39]	0.97
[10, 17, 31]	0.96
[27, 15]	0.97
[5, 29, 31]	0.91
[14, 38, 5, 36]	0.94

Tabela 2: Caption

Poniżej przedstawiono macierze pomyłek dla powyższych wyników klasyfikacji.

```

[[142  0  0  0  0  0  0  0  0  0]
 [  0 143  0  0  0  0  0  0  2  0]
 [  0  0 141  1  0  0  0  0  0  0]
 [  0  0  0 146  0  0  0  0  0  0]
 [  1  0  0  0 144  0  0  0  0  0]
 [  0  0  0  0  0 145  0  0  1  0]
 [  0  0  0  0  0  1 143  0  1  0]
 [  0  0  0  0  1  0  0 142  0  0]
 [  0  2  0  0  0  1  1  0 135  0]
 [  0  0  0  1  0  0  0  0  0 143]]

```

Rysunek 9: Macierz pomyłek dla sieci neuronowej [7, 29, 40, 9]

```

[[142  0  0  0  0  0  0  0  0  0]
 [  0 146  0  0  0  0  0  0  0  0]
 [  0  0 142  0  0  0  0  0  0  0]
 [  0  0  0 146  0  0  0  0  0  0]
 [  0  0  0  0 145  0  0  0  0  0]
 [  0  0  0  0  0 145  0  0  0  0]
 [  0  0  0  0  0  0 145  0  0  0]
 [  0  0  0  0  0  0  0 143  0  0]
 [  0  0  0  0  0  0  0  0 139  0]
 [  0  0  0  0  0  0  0  0  0 144]]

```

Rysunek 10: Macierz pomyłek dla sieci neuronowej [24, 22, 15]

```

[[142  0  0  0  0  0  0  0  0  0]
 [  0 144  0  0  0  0  0  0  1  0]
 [  0  0 142  0  0  0  0  0  0  0]
 [  0  0  0 145  0  0  0  0  1  0]
 [  0  0  0  0 144  0  0  1  0  0]
 [  0  0  0  0  0 146  0  0  0  0]
 [  0  0  0  0  0  0 145  0  0  0]
 [  0  0  0  0  0  0  0 143  0  0]
 [  0  3  0  0  0  0  0  0 136  0]
 [  0  0  0  0  0  0  0  0  0 144]]

```

Rysunek 11: Macierz pomyłek dla sieci neuronowej [10, 18, 22]

```

[[142  0  0  0  0  0  0  0  0  0]
 [  0 145  0  0  0  0  0  0  0  0]
 [  0  0 142  0  0  0  0  0  0  0]
 [  0  0  0 145  0  1  0  0  0  0]
 [  0  0  0  0 145  0  0  0  0  0]
 [  0  0  0  0  0 146  0  0  0  0]
 [  0  0  0  0  0  0 145  0  0  0]
 [  0  0  0  0  0  0  0 143  0  0]
 [  0  1  0  0  0  0  0  0 138  0]
 [  0  0  0  0  0  0  0  0  0 144]]

```

Rysunek 12: Macierz pomyłek dla sieci neuronowej [37, 6, 39]

```

[[142  0  0  0  0  0  0  0  0  0]
 [  0 145  0  0  0  0  0  0  0  0]
 [  0  0 142  0  0  0  0  0  0  0]
 [  0  0  0 146  0  0  0  0  0  0]
 [  0  0  0  0 145  0  0  0  0  0]
 [  0  0  0  0  0 146  0  0  0  0]
 [  0  0  0  0  0  0 145  0  0  0]
 [  0  0  0  0  0  0  0 142  0  1]
 [  0  0  0  0  0  0  0  0 139  0]
 [  0  0  0  0  0  0  0  0  0 144]]

```

Rysunek 13: Macierz pomyłek dla sieci neuronowej [10, 17, 31]



```

[[142  0  0  0  0  0  0  0  0  0]
 [  0 145  0  0  0  0  0  0  0  0]
 [  0  0 142  0  0  0  0  0  0  0]
 [  0  0  0 146  0  0  0  0  0  0]
 [  0  0  0  0 144  0  0  0  1  0]
 [  0  0  0  0  0 146  0  0  0  0]
 [  0  0  0  0  0  0 145  0  0  0]
 [  0  0  0  0  0  0  0 142  0  1]
 [  0  0  0  0  0  0  0  0 139  0]
 [  0  1  0  0  0  0  0  0  0 142]]

```

Rysunek 14: Macierz pomyłek dla sieci neuronowej [27, 15]

```

[[142  0  0  0  0  0  0  0  0  0]
 [  0 143  0  0  0  0  0  0  3  0]
 [  0  4 136  0  0  0  0  0  2  0]
 [  0  0  2 140  0  1  0  0  2  1]
 [  0  0  0  0 145  0  0  0  0  0]
 [  0  0  0  0  0 145  0  0  0  0]
 [  0  0  1  0  0  0 144  0  0  0]
 [  0  0  0  0  1  0  0 142  0  0]
 [  0  4  0  2  0  0  0  0 133  0]
 [  0  0  0  4  0  0  0  1  0 139]]

```

Rysunek 15: Macierz pomyłek dla sieci neuronowej [5, 29, 31]

[	142	0	0	0	0	0	0	0	0]
[	0	144	0	0	0	0	0	1	0]
[	0	0	142	0	0	0	0	0	0]
[	0	0	0	146	0	0	0	0	0]
[	0	0	0	0	142	0	0	3	0]
[	0	0	0	0	0	145	0	0	1]
[	0	0	0	0	0	0	145	0	0]
[	0	0	0	0	0	0	0	143	0]
[	0	3	0	0	0	0	0	0	136]
[	0	0	0	0	0	0	0	0	144]]

Rysunek 16: Macierz pomyłek dla sieci neuronowej [14, 38, 5, 36]

## 2.1 Wnioski

- Wielowarstwowe sieci neuronowe potrafią uzyskać bardzo wysokie wyniki klasyfikacji
- Duży wpływ na skuteczność sieci mają wpływ parametry sieci oraz normalizacja danych

## 3 Testowanie różnych ustawień sieci neuronowych

W kolejnym zadaniu należało przedtestować różne parametry sieci neuronowych oraz zbadać ich wyniki klasyfikacji. Wyniki klasyfikacji dla różnych parametrów zostały przedstawione w pliku "zad3.txt" ze względu na ich ilość.

### 3.1 Wnioski

- Najlepiej sprawdził się algorytm *ADAM* z maksymalną dokładnością 0.96
- Dla algorytmu *SGD* najlepiej sprawdziła się funkcja aktywacji *identity*

## 4 Analiza zbioru yeasts za pomocą sieci neuronowych

W kolejnym zadaniu pobrano zestaw *yeast* oraz dokonano jego analizy za pomocą różnych algorytmów oraz funkcji aktywacji. Wyniki zostały zapisane w pliku "zad4.txt"

### 4.1 Wnioski

- Skuteczność klasyfikacji na poziomie 0.5 nie jest wystarczająca, model działa losowo
- Najlepsza skuteczność jaką uzyskano to 0.60
- Algorytm *ADAM* wypada znacznie lepiej niż algorytm *SGD*