

Algorytmy Ewolucyjne

Projekt 3

Michał Kwarciański

Sprawozdanie

Kacper Marchlewicz

1. Treść zadania

Zaimplementuj algorytm uczenia się liniowego klasyfikatora binarnego w przestrzeni \mathbb{R}^3 .

- Założenia:

- Zbiór treningowy $D = \{(\bar{x}_1, y_1), (\bar{x}_2, y_2), \dots, (\bar{x}_N, y_N)\}$ dla $N = 20$, $\bar{x}_N \in \mathbb{R}^3$ i $y_N \in \{+1, -1\}$ ma zostać wygenerowany za funkcji `AEproj3_data`,
- dozwolone jest korzystanie z języków MATLAB, C++ i Python.

- Zaimplementuj algorytm uczenia się perceptronu opisany w: L. Hamel „Knowledge discovery with support vector machines”, rozdział 5.1. Książka jest dostępna on-line przez stronę Biblioteki Głównej PW.

- Rozbuduj algorytm o walidację, która pozwoli badać dokładność klasyfikacji (na podstawie liczby błędów) w trakcie uczenia się perceptronu. Walidacja może polegać np. na klasyfikacji danych ze zbioru treningowego po zakończeniu każdej iteracji lub każdego cyklu uczenia się perceptronu.

- Przeprowadź uczenie się perceptronu o początkowych parametrach odpowiadającym płaszczyźnie stycznej do osi rzędnych i odciętych dla trzech różnych wartości współczynnika szybkości uczenia się. Na podstawie badań wskaż najlepszą wartość tego współczynnika.

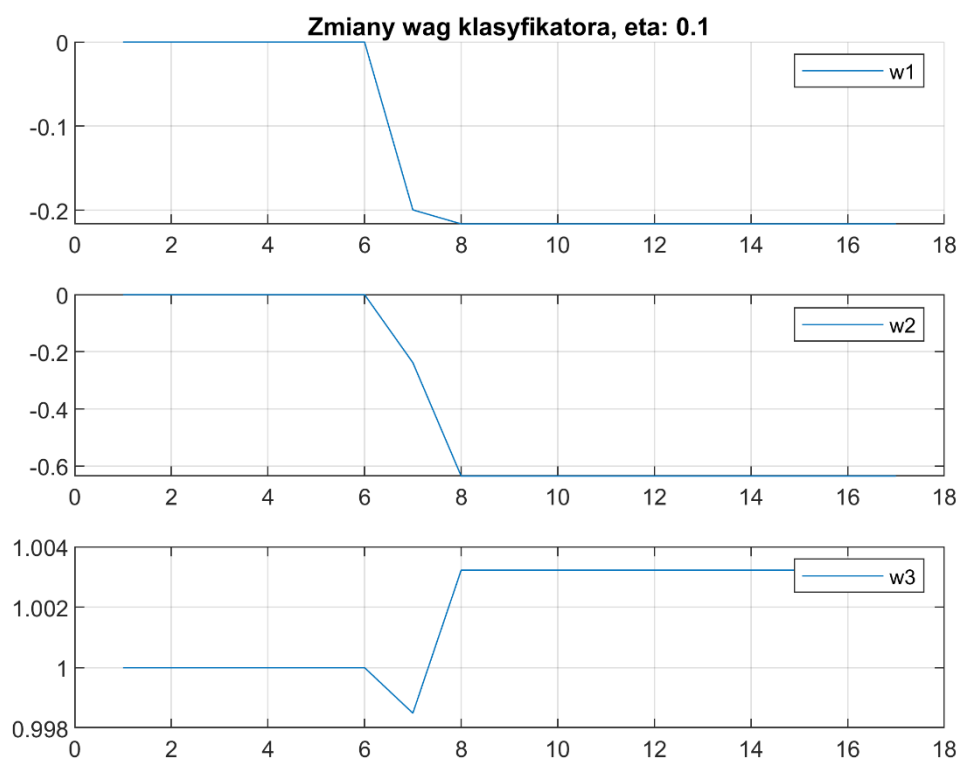
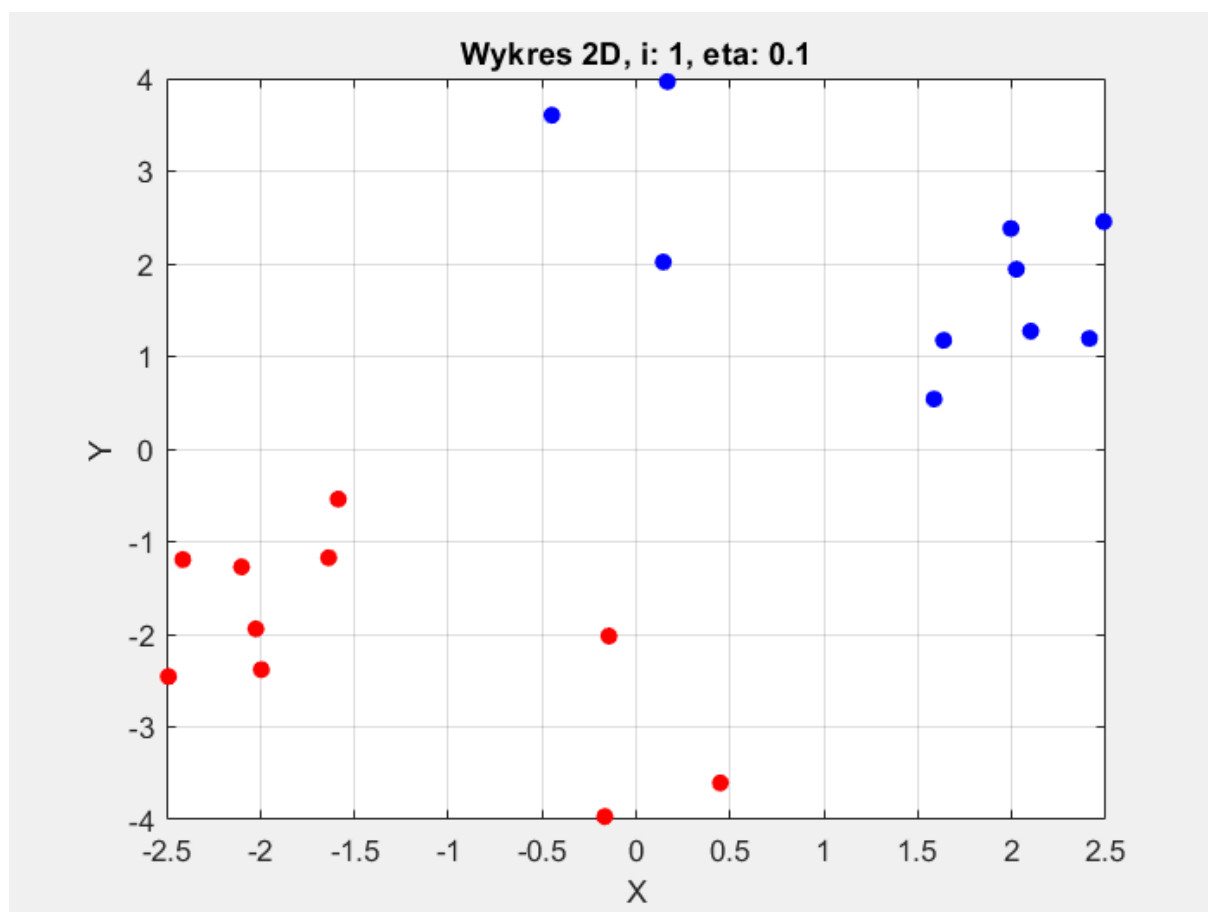
2. Prezentacja wyników

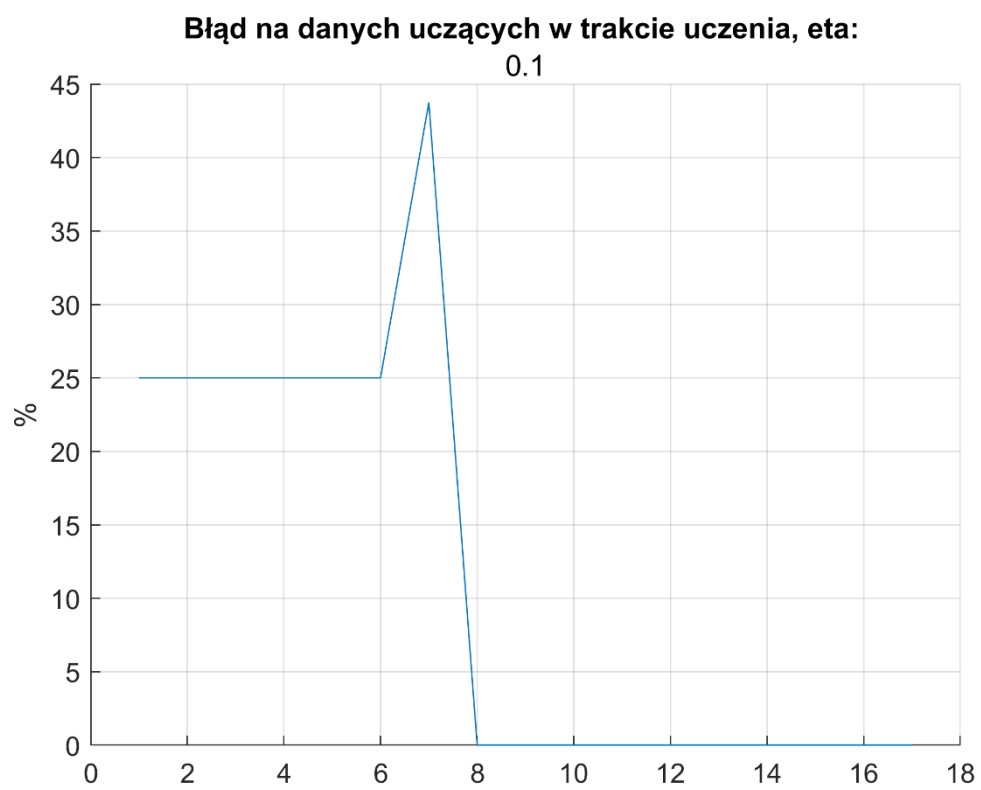
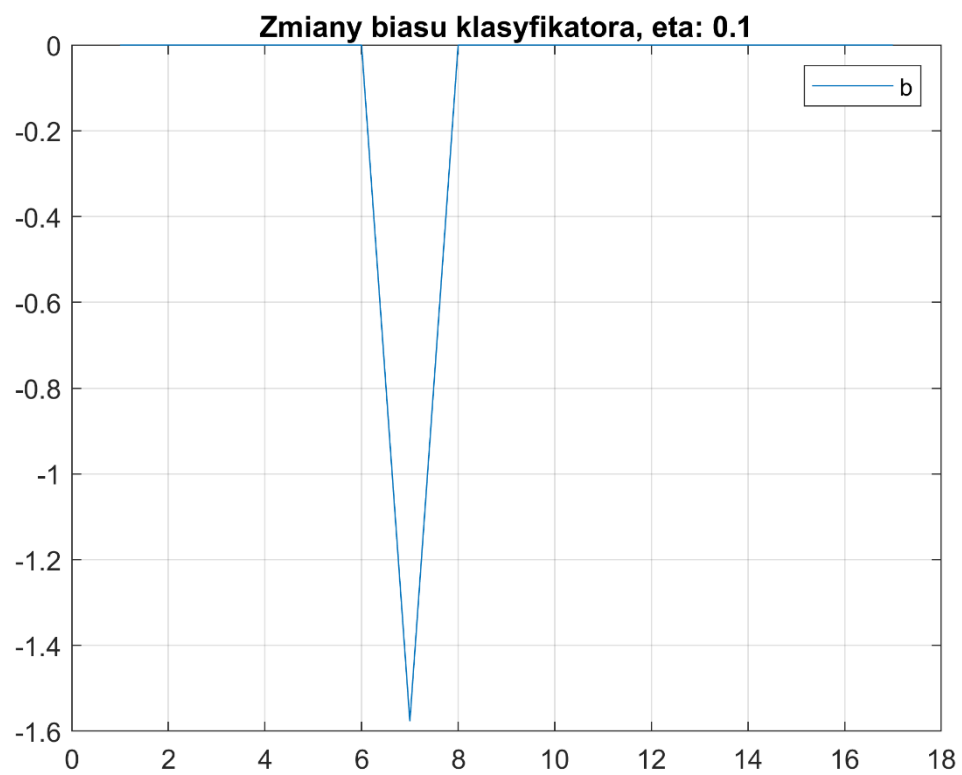
Dane podzieliliśmy na dwa zbiory: uczący (80% wszystkich danych) i testowy (pozostałe dane).

Uczenie przeprowadziliśmy dla trzech wartości współczynnika szybkości uczenia się (0,1; 0,5; 0,9). Jak widać na poniższych zdjęciach początkowe parametry $w=[0, 0, 1]$ i $b=0$ bez nauki poprawnie klasyfikowały 80% wszystkich przypadków, 75% danych uczących i 100% testowych. Wynika to z zbyt małej ilości danych testowych (tylko 4 testowe) i po prostu tak się wylosowały. Dla pierwszych danych błędnie sklasyfikowanych nastąpiła gwałtowna zmiana paramentów, która zwolniła. Po trzech iteracjach uczących (w których nastąpiła zmiana parametrów – błędnie skalsyfikowane) perceptrony nauczyły się poprawnie klasyfikować dany podzbiór. Można powiedzieć, że dane były proste do nauki. Widać, że klasy się nie pokrywały i był spory odstęp między nimi.

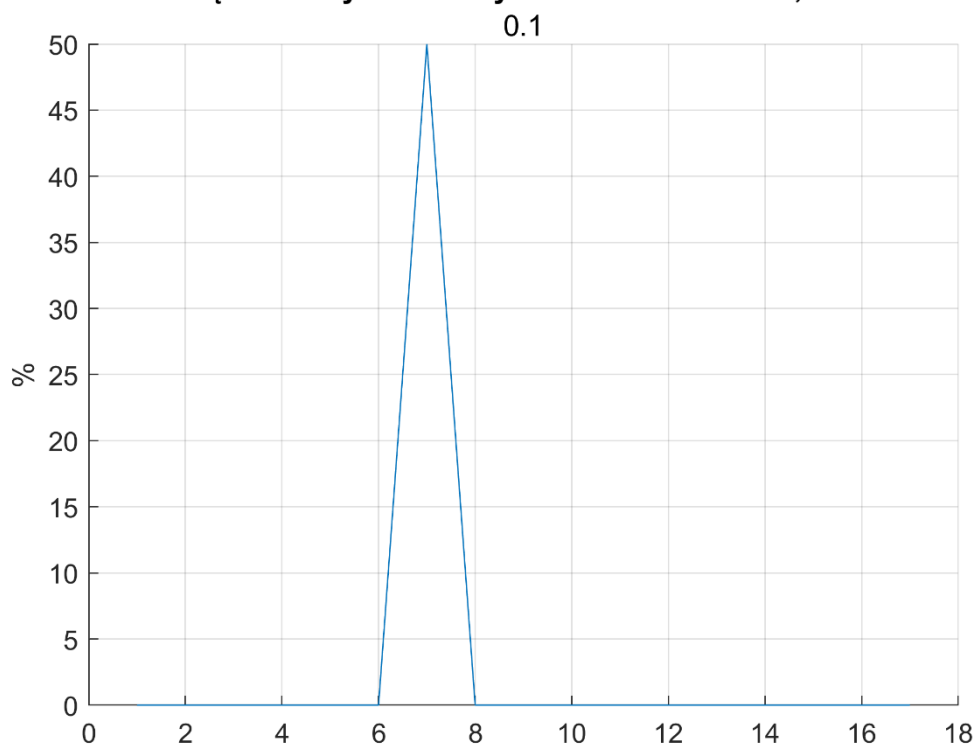
W przypadku, gdyby plik gif nie odtwarzał się w pliku pdf, w archiwum załączamy plik word i pliki gif

Dla $\eta = 0.1$:

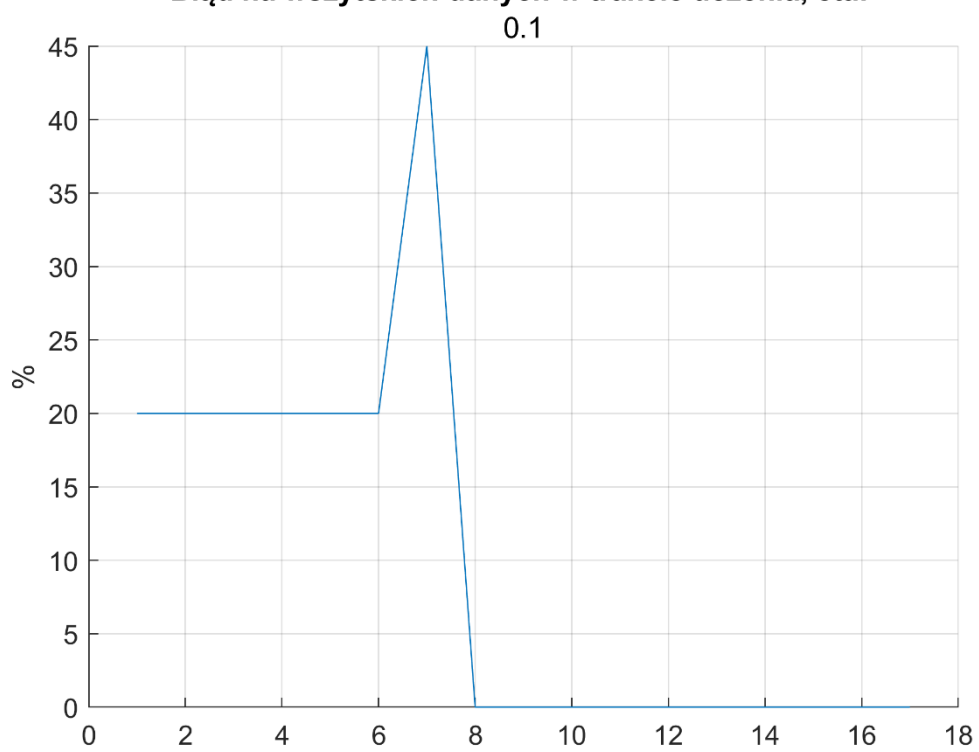




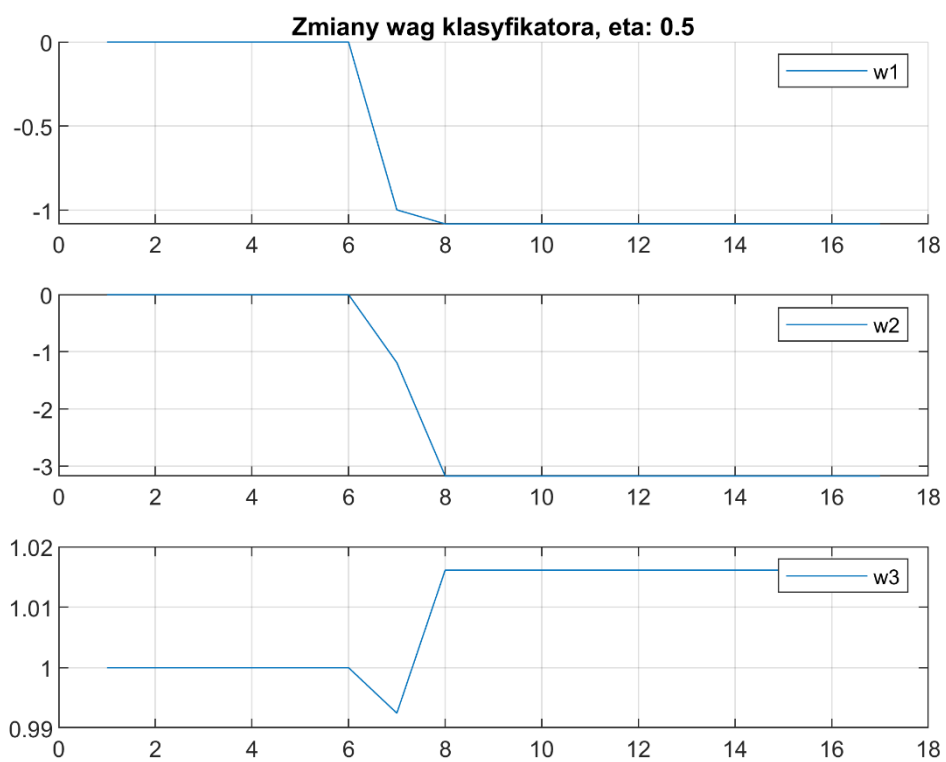
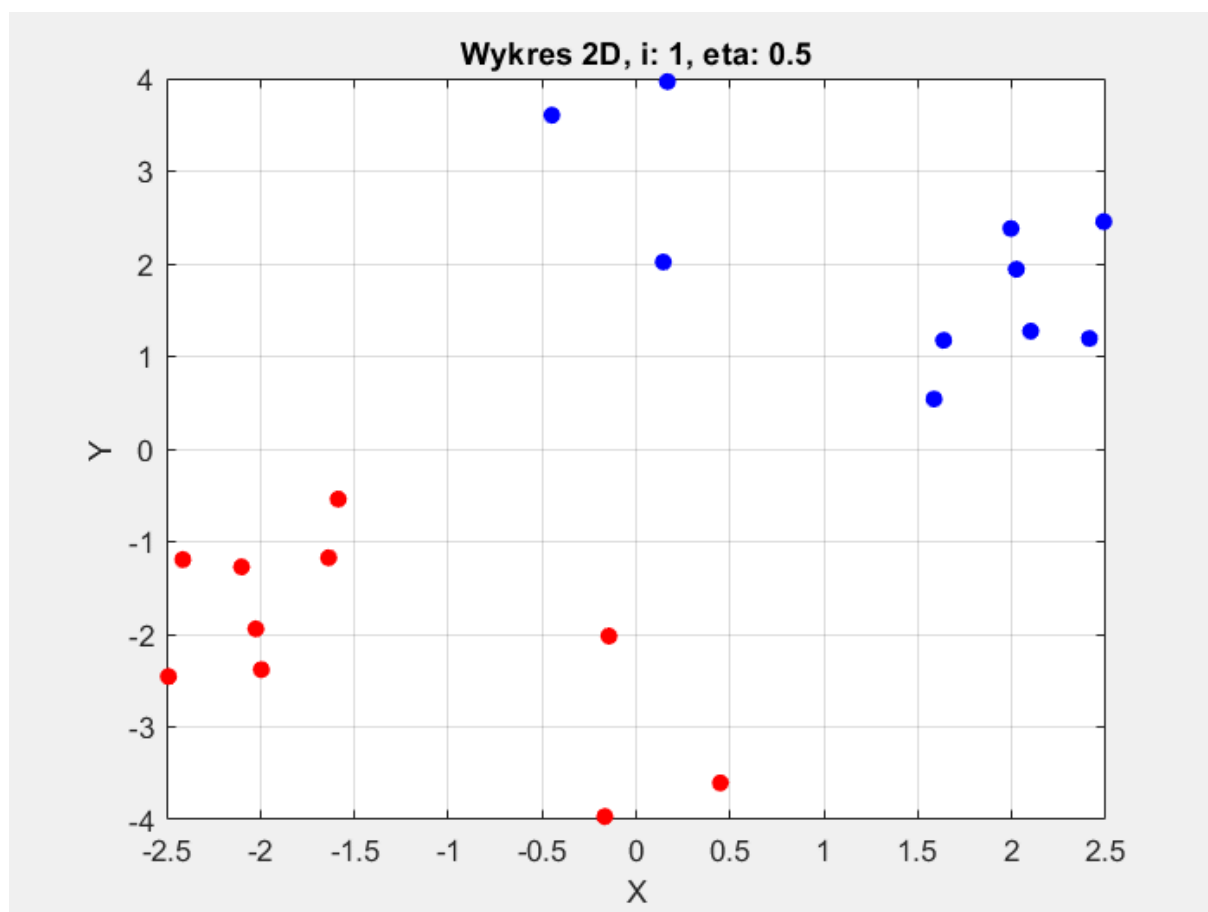
Błąd na danych testowych w trakcie uczenia, eta:

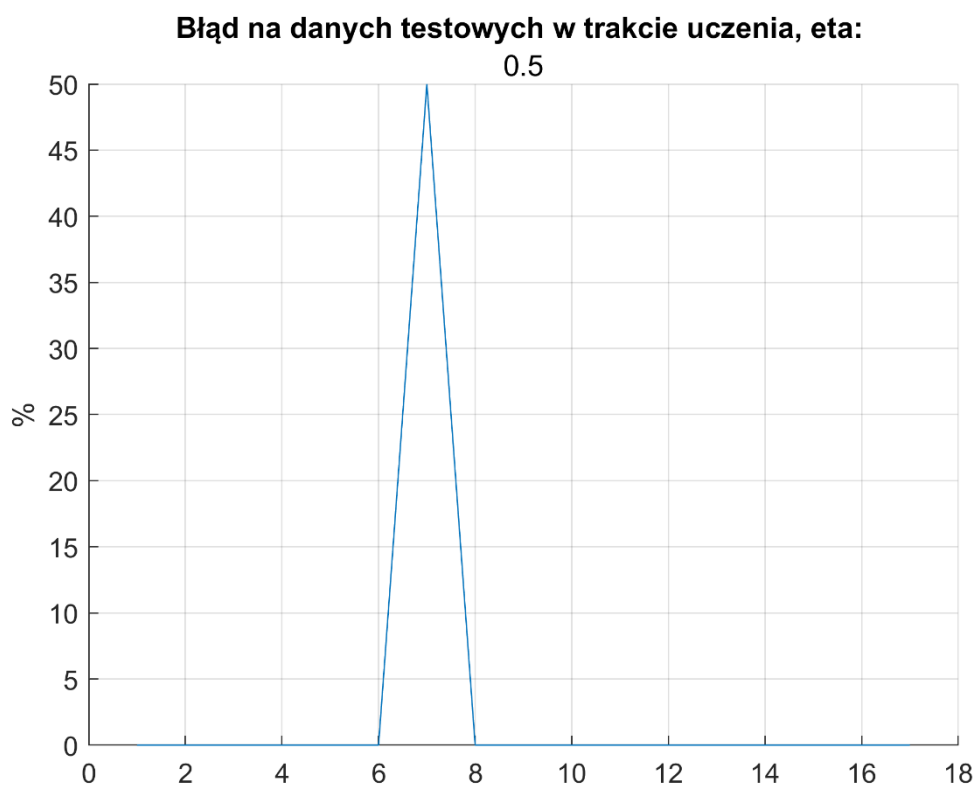
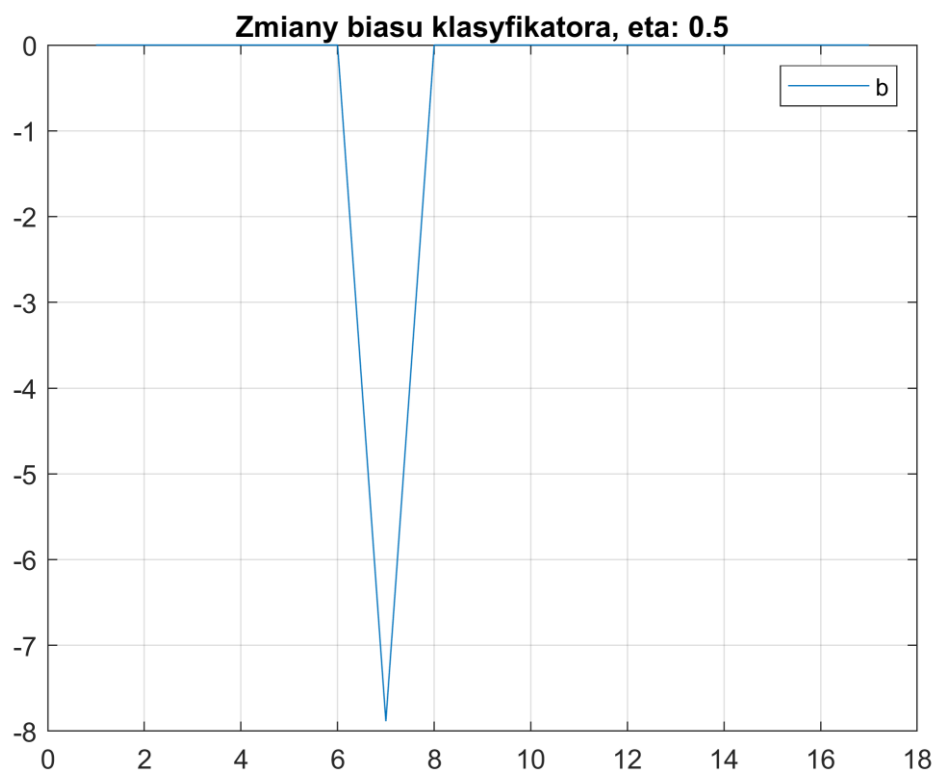


Błąd na wszystkich danych w trakcie uczenia, eta:

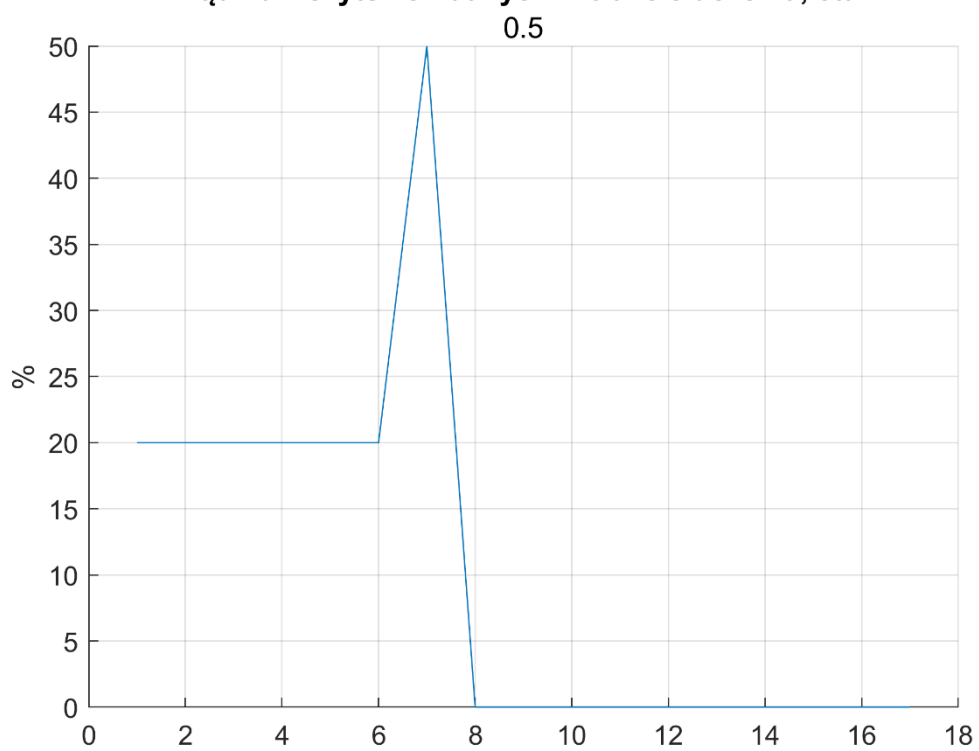


Dla $\eta = 0.5$:

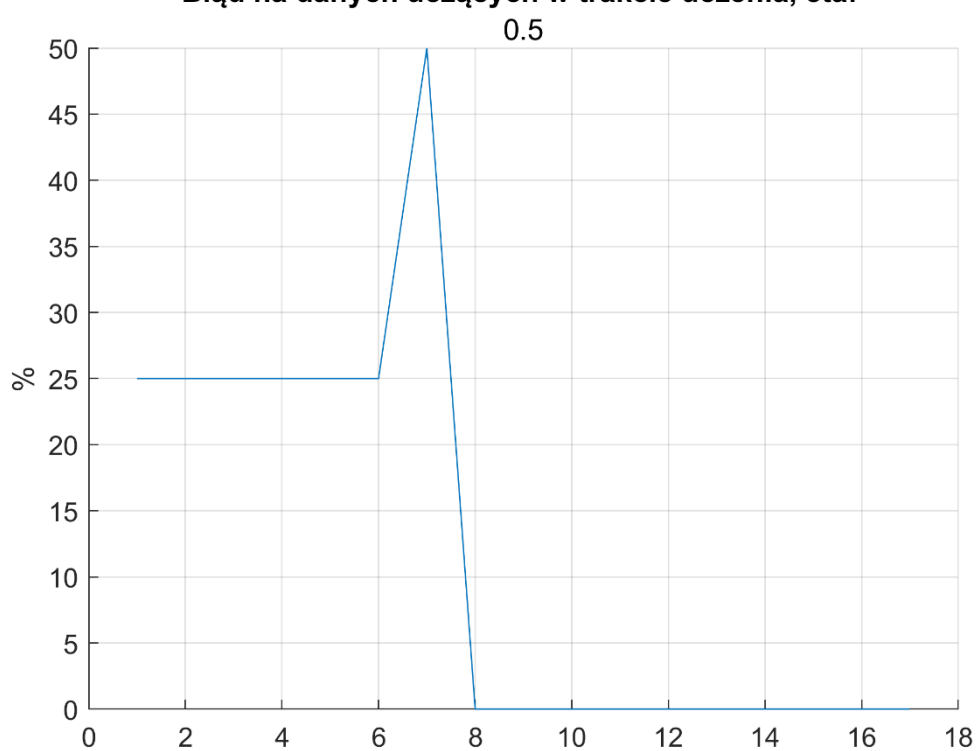




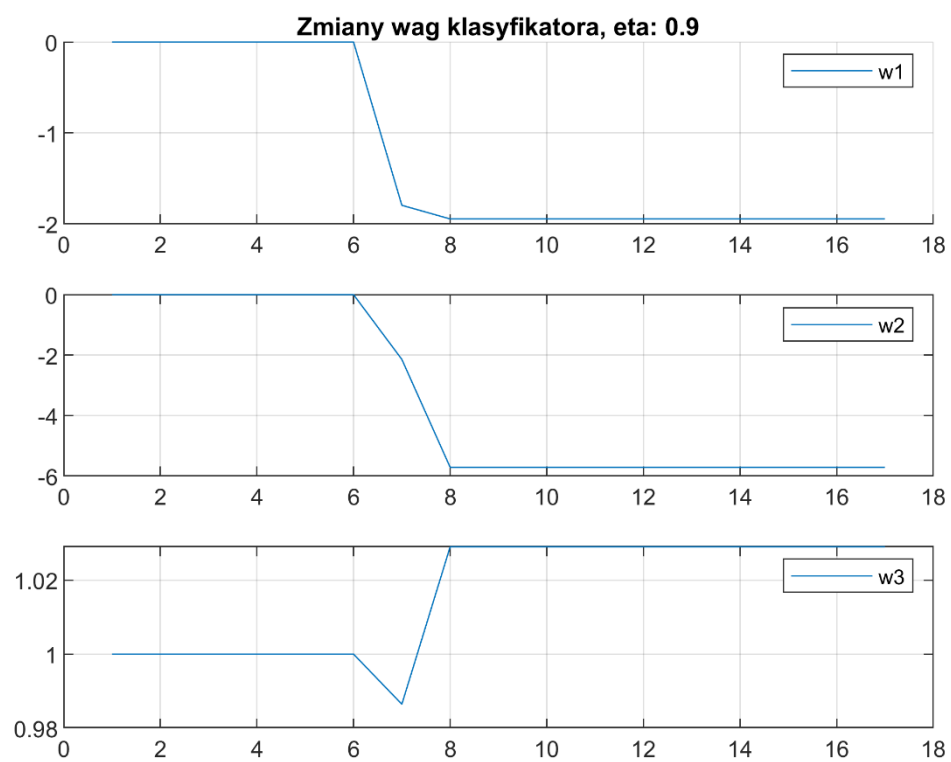
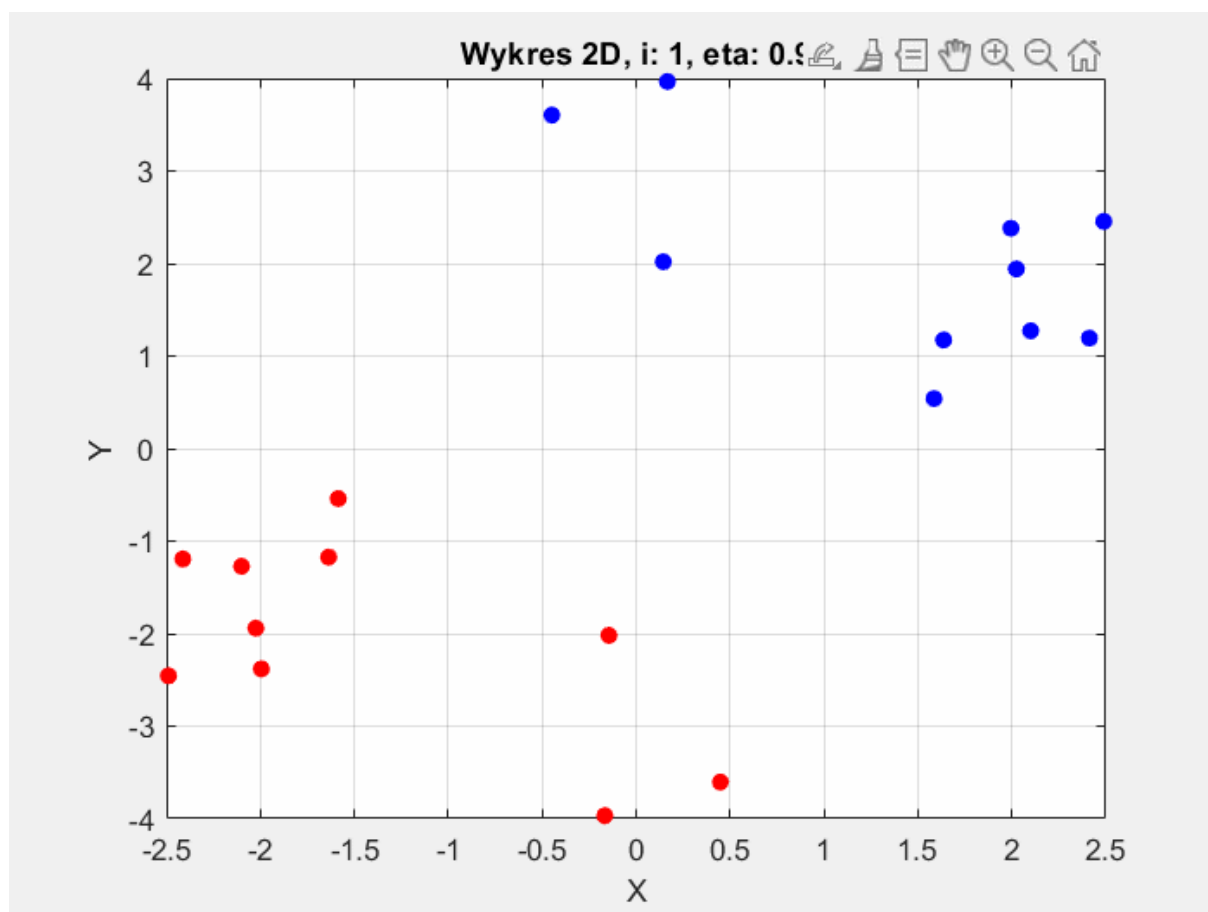
Błąd na wszystkich danych w trakcie uczenia, eta:

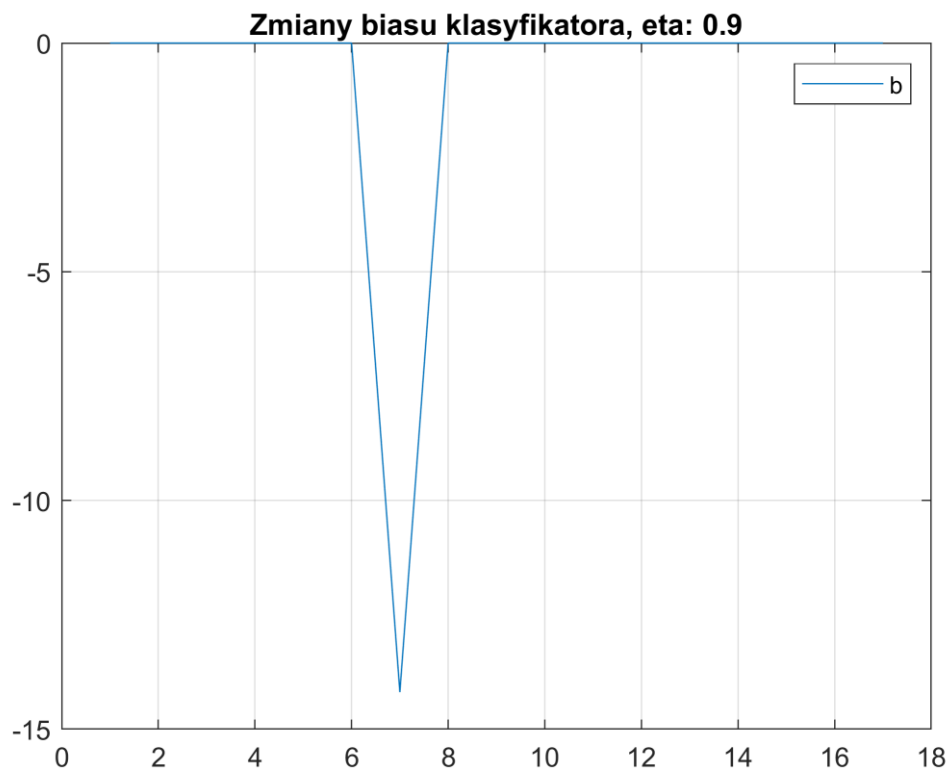


Błąd na danych uczących w trakcie uczenia, eta:

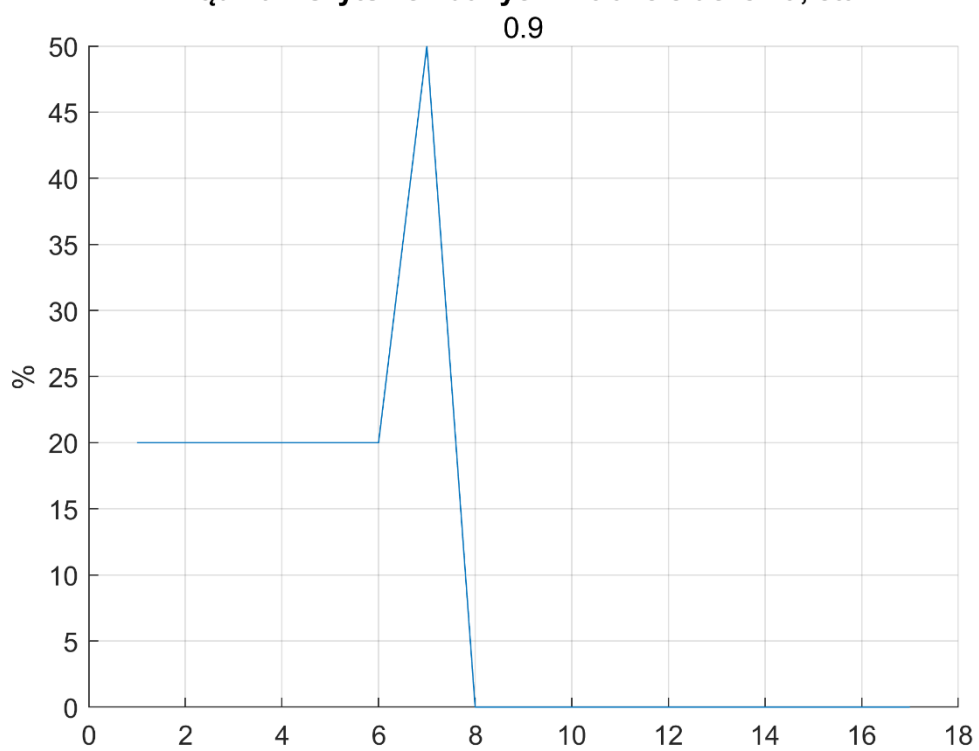


Dla $\eta = 0.9$:

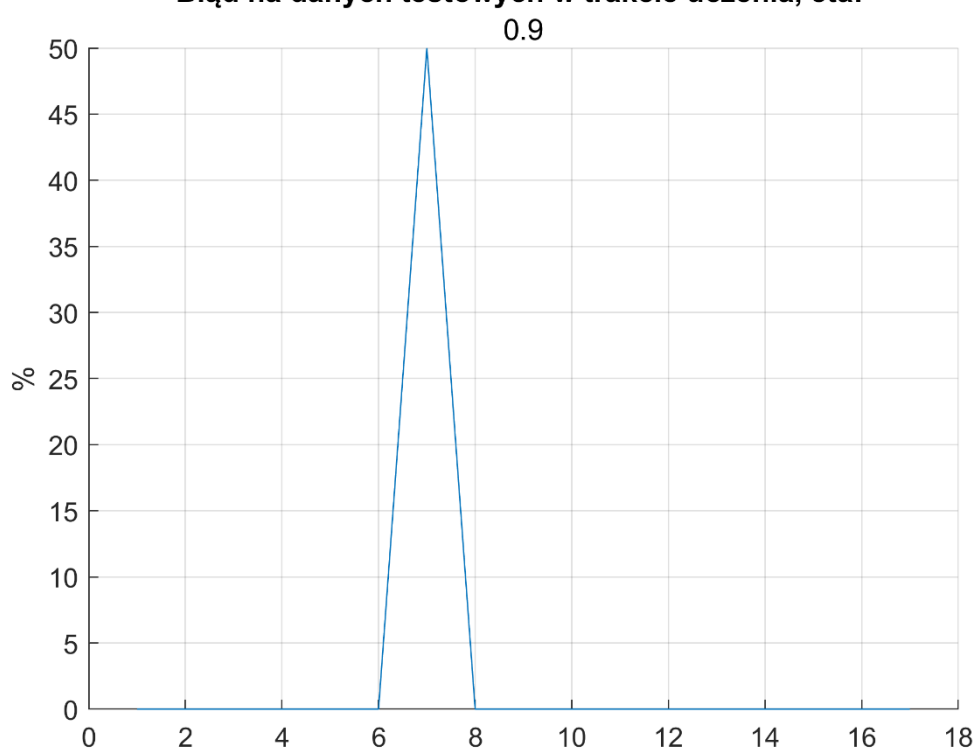




Błąd na wszystkich danych w trakcie uczenia, eta:



Błąd na danych testowych w trakcie uczenia, eta:



Zmiany parametru wagi i biasu dla klasyfikatora uczonego z parametrem $\eta=0.9$ znajdują się na poniższej tabeli i wykresach.

iteracja	w_1	w_2	w_3	b
1	0	0	1	0
2	0	0	1	0
3	0	0	1	0
4	0	0	1	0
5	0	0	1	0
6	-1,7955	-2.1429	0,9865	-14,1917
7	-1,9454	-5.7134	1,0290	0
8	-1,9454	-5.7134	1,0290	0
9	-1,9454	-5.7134	1,0290	0
10	-1,9454	-5.7134	1,0290	0
11	-1,9454	-5.7134	1,0290	0
12	-1,9454	-5.7134	1,0290	0
13	-1,9454	-5.7134	1,0290	0
14	-1,9454	-5.7134	1,0290	0
15	-1,9454	-5.7134	1,0290	0
16	-1,9454	-5.7134	1,0290	0