

# Sieci neuronowe – laboratorium nr 2

## Sprawozdanie

Oznaczenia:

$h$  – liczba neuronów w warstwie ukrytej

$b$  – rozmiar mini – batcha

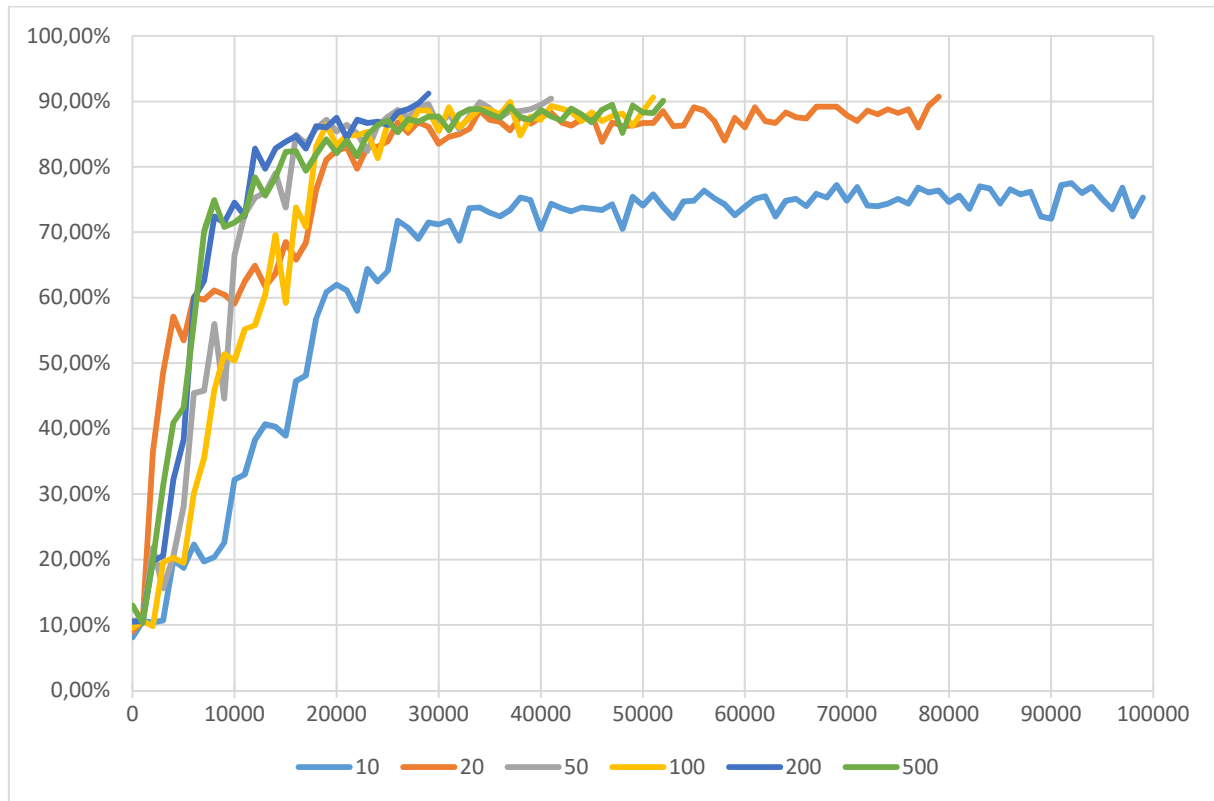
$w$  – zakres wag początkowych (od  $-w_i^0$  do  $+w_i^0$ )

$\alpha$  – współczynnik uczenia

$m$  – współczynnik momentum

Poniższe testy prowadzone są do osiągnięcia pożądanej dokładności – 90% lub do 100000 iteracji na danych treningowych

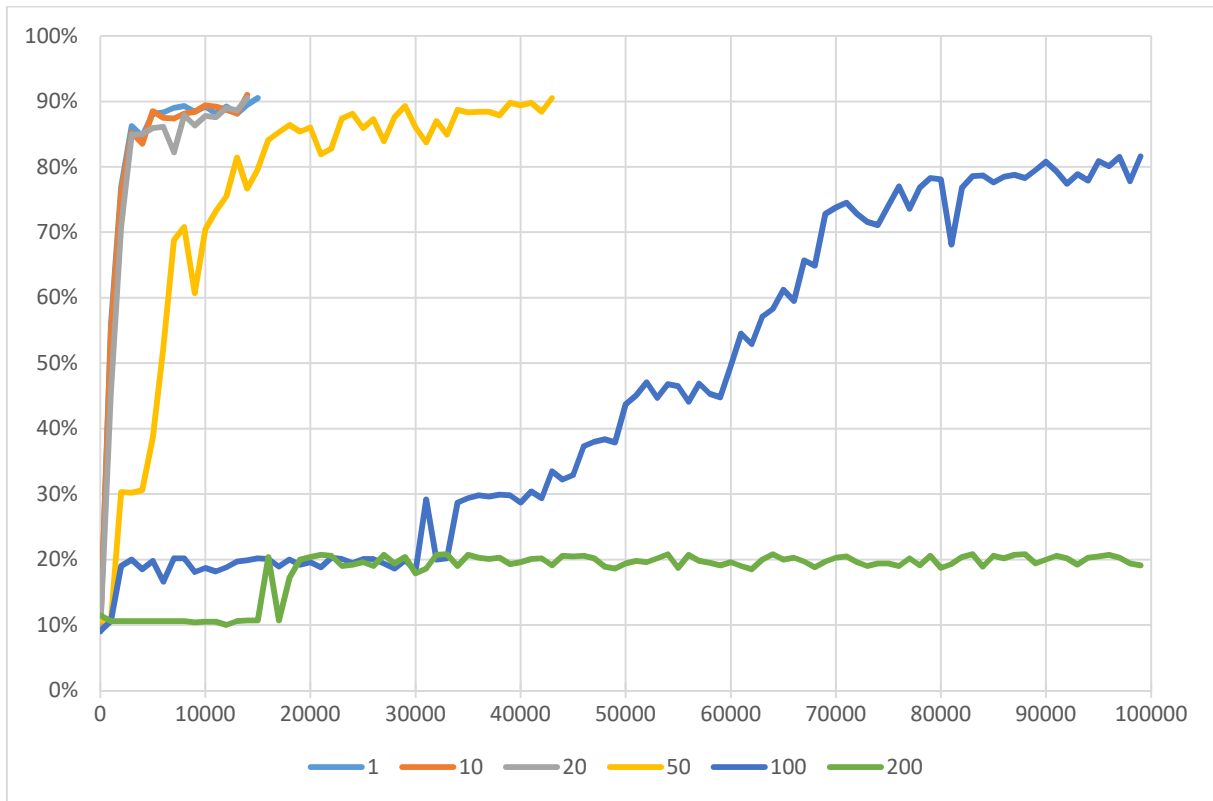
## Test 1. Wpływ liczby neuronów

 $h$  – zmienne $b = 50; w = 0,05; \alpha = 0,05; m = 0,05$ 

h	Dokładność	Czas	Dane treningowe
10	0,7682	26,954	100000
20	0,9121	43,524	79050
50	0,9114	45,72	41050
100	0,9175	103,246	51050
200	0,908	114,285	29050
500	0,9111	552,405	52050

- 10 neuronów nie wystarcza żeby osiągnąć pożądaną dokładność
- 20 to minimum, wymaga dużych danych testowych, ale zmniejsza czas uczenia
- 200 pozwala osiągnąć pożądaną dokładność przy najmniejszych danych testowych
- Zwiększenie liczby neuronów zwiększa czas uczenia

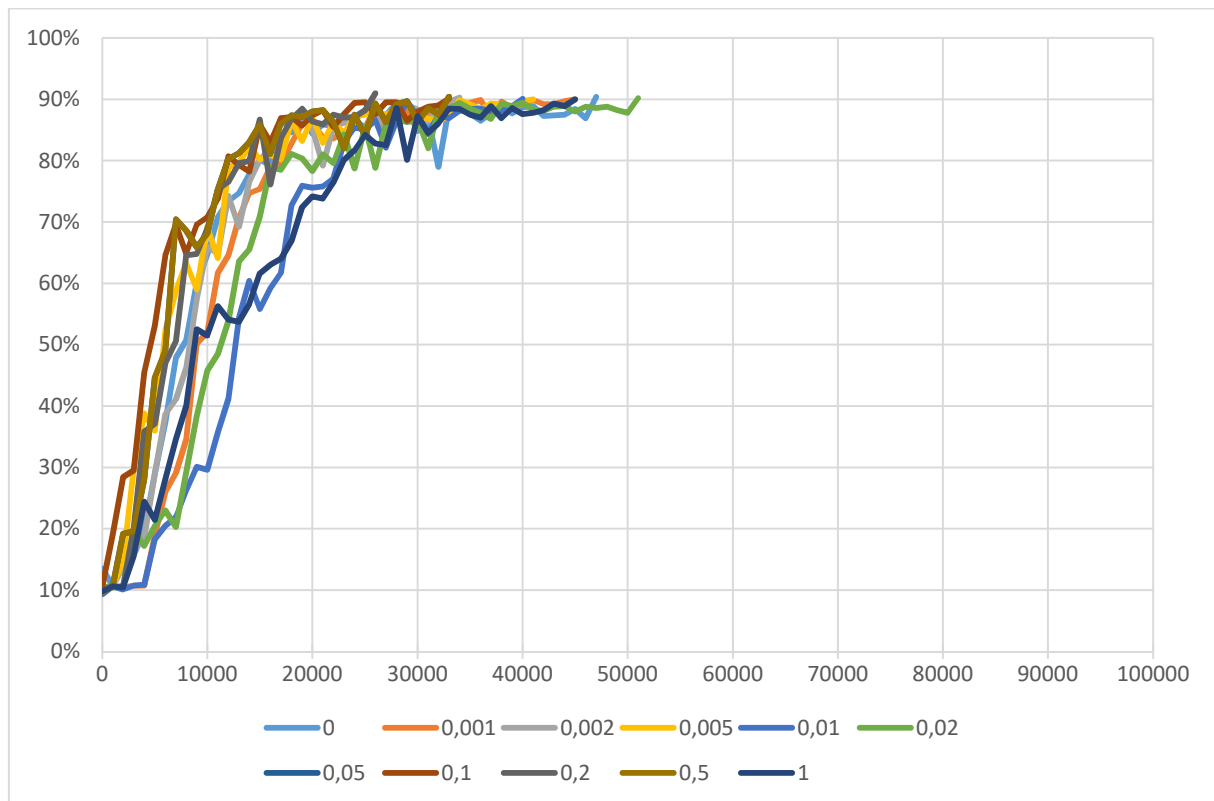
## Test 2. Wpływ rozmiaru mini – batcha

 $b$  – zmienne $h = 100; w = 0,05; \alpha = 0,05; m = 0,05$ 

b	Dokładność	Czas	Dane treningowe
1	0,9229	69,562	15001
10	0,9122	32,617	14010
20	0,9028	30,795	14020
50	0,9078	86,436	43050
100	0,8179	191,033	100000
200	0,195	188,924	100000
500	0,1135	190,296	100000

- Batch o rozmiarze powyżej 20 traci sens
- Batche o rozmiarze 10 i 20 uzyskują tą samą dokładność przy zbliżonych danych treningowych, w znacznie przyspieszonym czasie

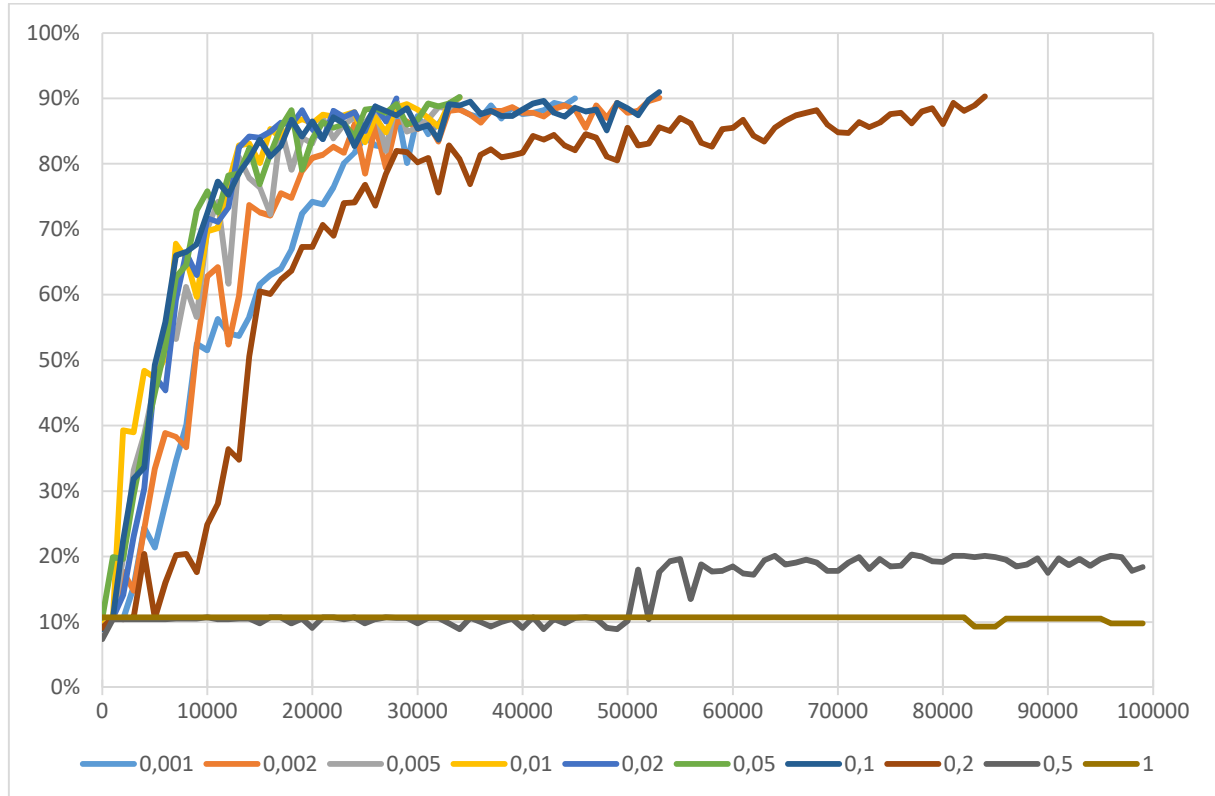
## Test 3. Wpływ wag początkowych

 $w$  – zmienne $h = 100; b = 50; \alpha = 0,05; m = 0,05$ 

w	Dokładność	Czas	Dane treningowe
0	0,9154	94,782	47050
0,001	0,8887	90,488	45050
0,002	0,9022	68,388	34050
0,005	0,9132	82,418	41050
0,01	0,8944	80,288	40050
0,02	0,9153	104,4	52050
0,05	0,9083	66,586	33050
0,1	0,9122	68,185	34050
0,2	0,9037	52,371	26050
0,5	0,8952	60,307	30050
1	0,9112	92,412	46050

- Najlepsze efekty daje  $w=0,2$
- Różnice nie są duże

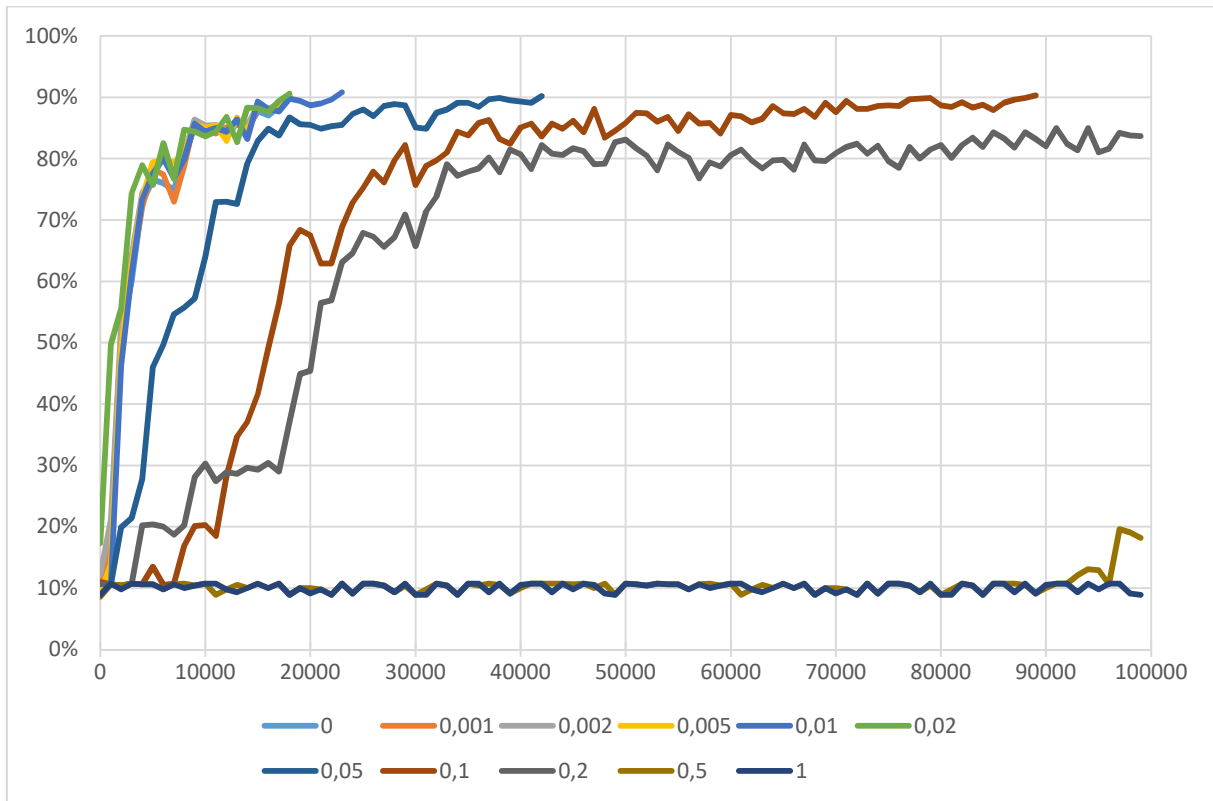
## Test 4. Wpływ współczynnika uczenia

 $\alpha$  – zmienne $h = 100; b = 50; w = 0,05; m = 0,05$ 

$\alpha$	Dokładność	Czas	Dane treningowe
0,001	0,9004	90,24	45050
0,002	0,9145	106,593	53050
0,005	0,9156	68,272	34050
0,01	0,914	68,368	34050
0,02	0,8968	56,23	28050
0,05	0,9106	70,24	35050
0,1	0,9132	106,411	53050
0,2	0,8919	168,433	84050
0,5	0,1837	201,497	100000
1	0,1009	200,037	100000

- Najlepszy efekt daje  $\alpha = 0,02$
- Powyżej 0,5 sieć przestaje się uczyć

## Test 5. Wpływ momentu

 $m$  – zmienne $h = 100; b = 50; w = 0,05; \alpha = 0,05$ 

m	Dokładność	Czas	Dane treningowe
0	0,9107	36,233	18050
0,001	0,9058	35,904	18050
0,002	0,9045	36,086	18050
0,005	0,9063	36,19	18050
0,01	0,8994	46,21	23050
0,02	0,9104	38,235	19050
0,05	0,915	84,292	42050
0,1	0,902	178,424	89050
0,2	0,8408	199,852	100000
0,5	0,2008	201,204	100000
1	0,1135	200,082	100000

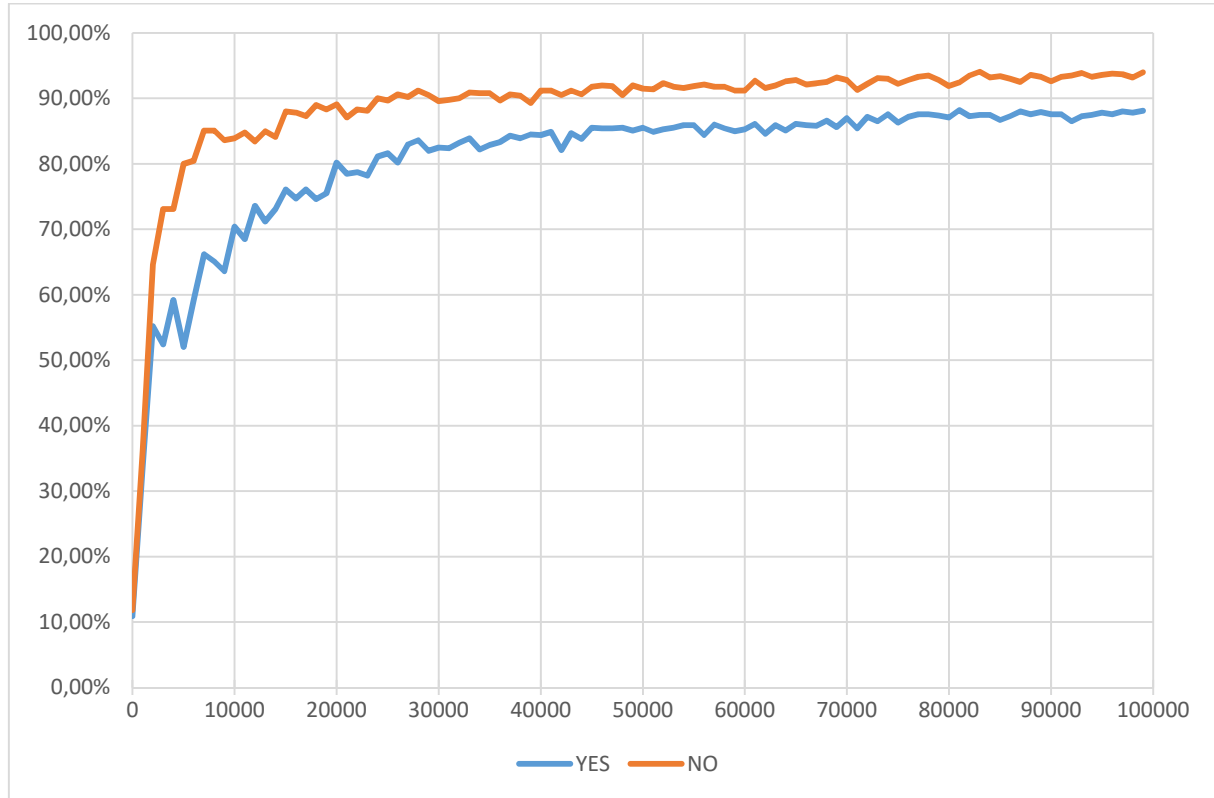
- Najlepszy efekt daje współczynnik momentu w przedziale  $[0; 0,005]$
- Powyżej 0,5 sieć przestaje się uczyć
- Współczynnik momentu musi być mniejszy od współczynnika uczenia, jeśli jest większy wydłuża czas uczenia

## Test 6. Zmienny współczynnik uczenia

$d$  – dokładność na ciągu walidacyjnym

$$\alpha = \frac{\alpha_0}{10 \cdot d}$$

$$h = 200; b = 20; w = 0,2; \alpha_0 = 0,02; m_0 = 0,005$$

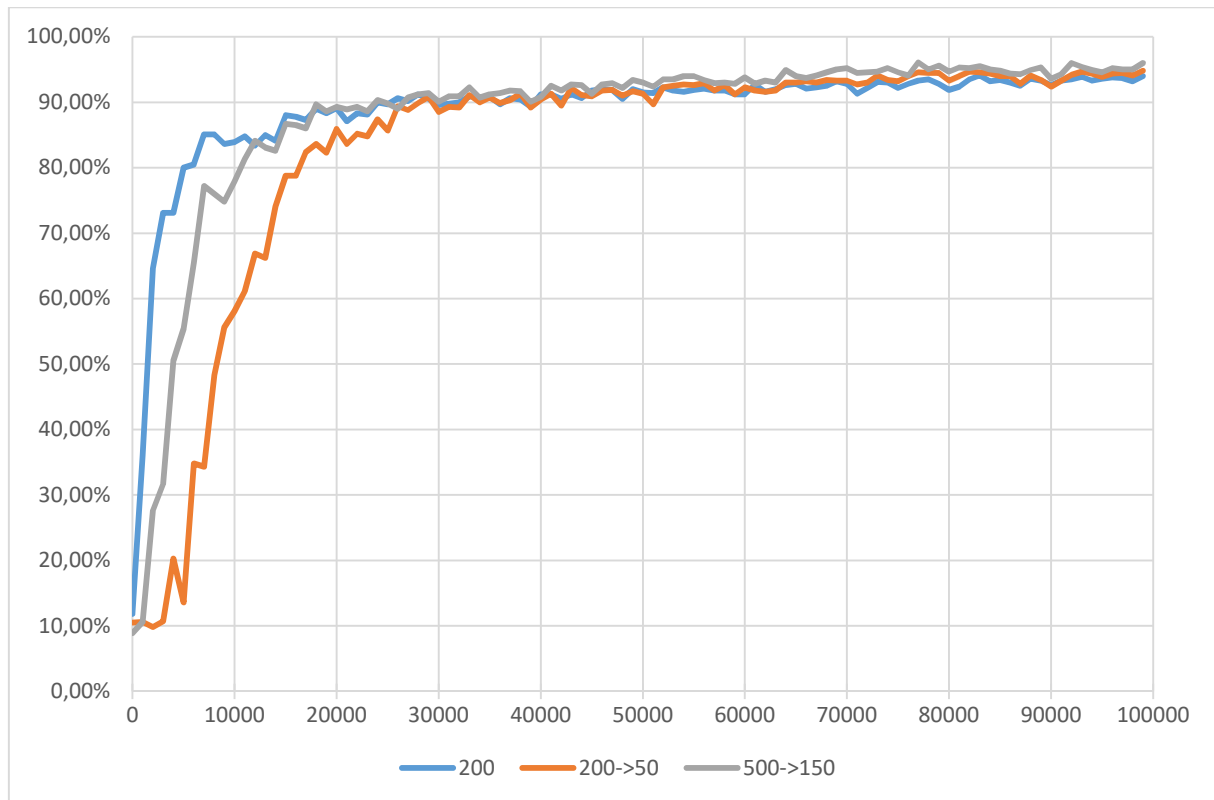


Adaptive?	Dokładność	Czas	Dane treningowe
YES	0,8982	445,463	100000
NO	0,9454	473,809	100000

- Ten sposób adaptacji współczynnika uczenia nie pomaga

## Test 7. Wpływ dodatkowej warstwy ukrytej

$$b = 20; w = 0,2; \alpha_0 = 0,02; m_0 = 0,005$$



warstwy	Dokładność	Czas	Dane treningowe
200	0,9454	473,809	100000
200->50	0,9526	497,274	100000
500->150	0,9574	1383,403	100000

- Dodatkowa warstwa pozwala osiągnąć minimalnie lepsze wyniki za cenę wolniejszego uczenia



## Wnioski

- Wyniki sieci neuronowej (szybkość uczenia i dokładność) zależą w dużym stopniu od dobranych parametrów
- Najwyższa osiągalna dokładność dla prostej sieci MLP w tym zadaniu to ok. 96%
- Dodatkowe warstwy sieci nie dają istotnej różnicy
- Optymalne parametry dla jednowarstwowej sieci:  $h = 200$ ;  $b = 20$ ;  $w = 0,2$ ;  $\alpha = 0,02$ ;  $m = 0,005$