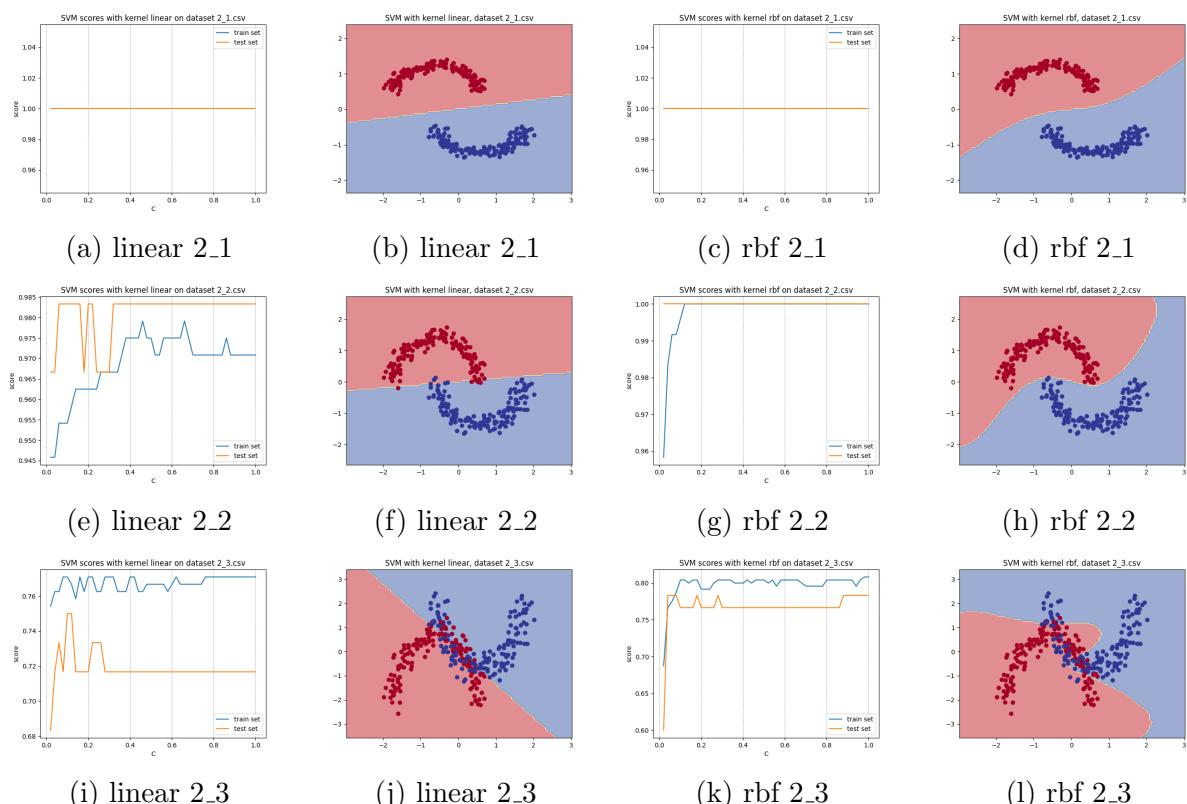
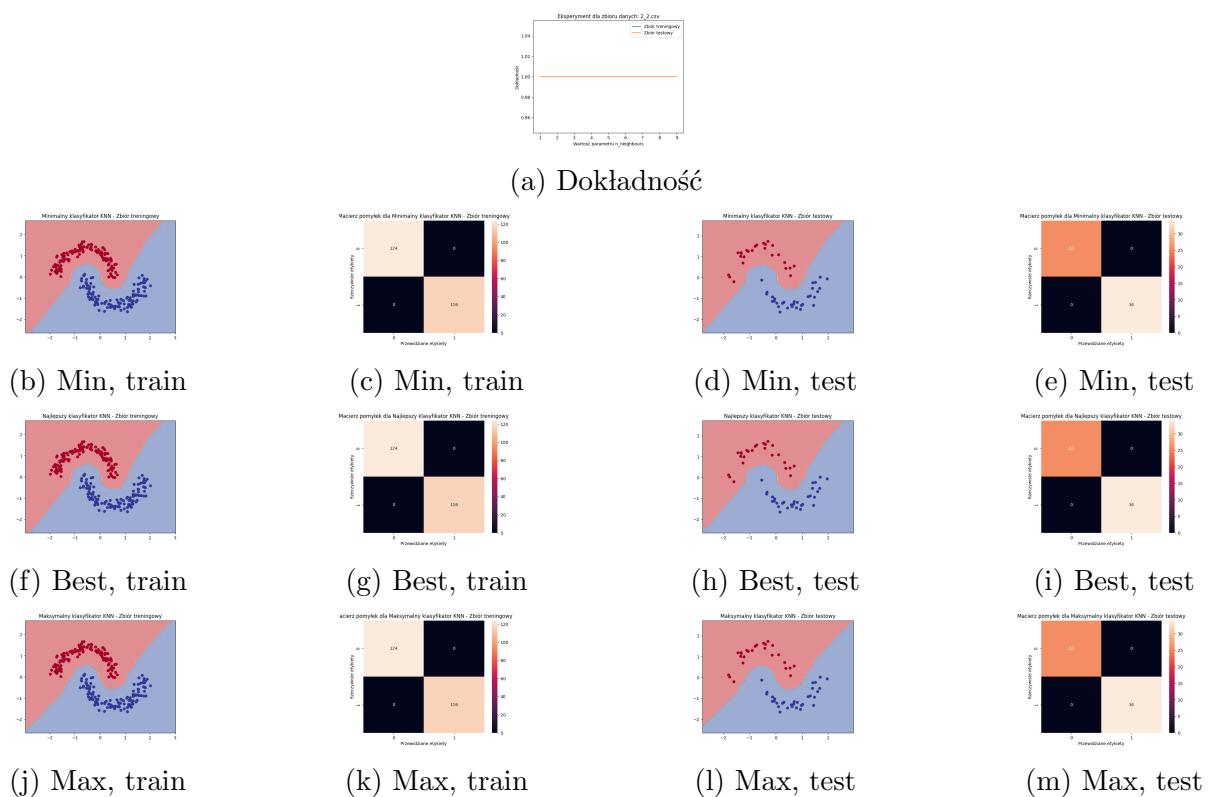


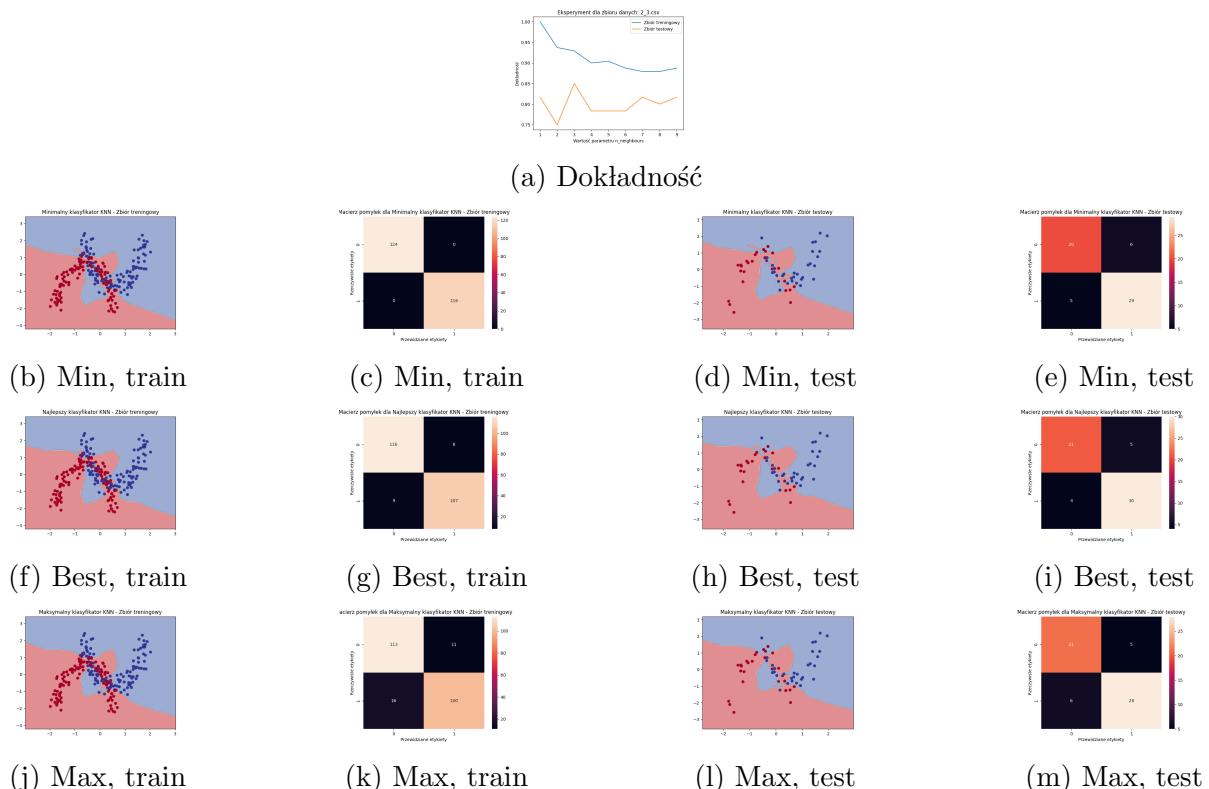
Rysunek 1: Eksperyment 1 — Perceptron MLP



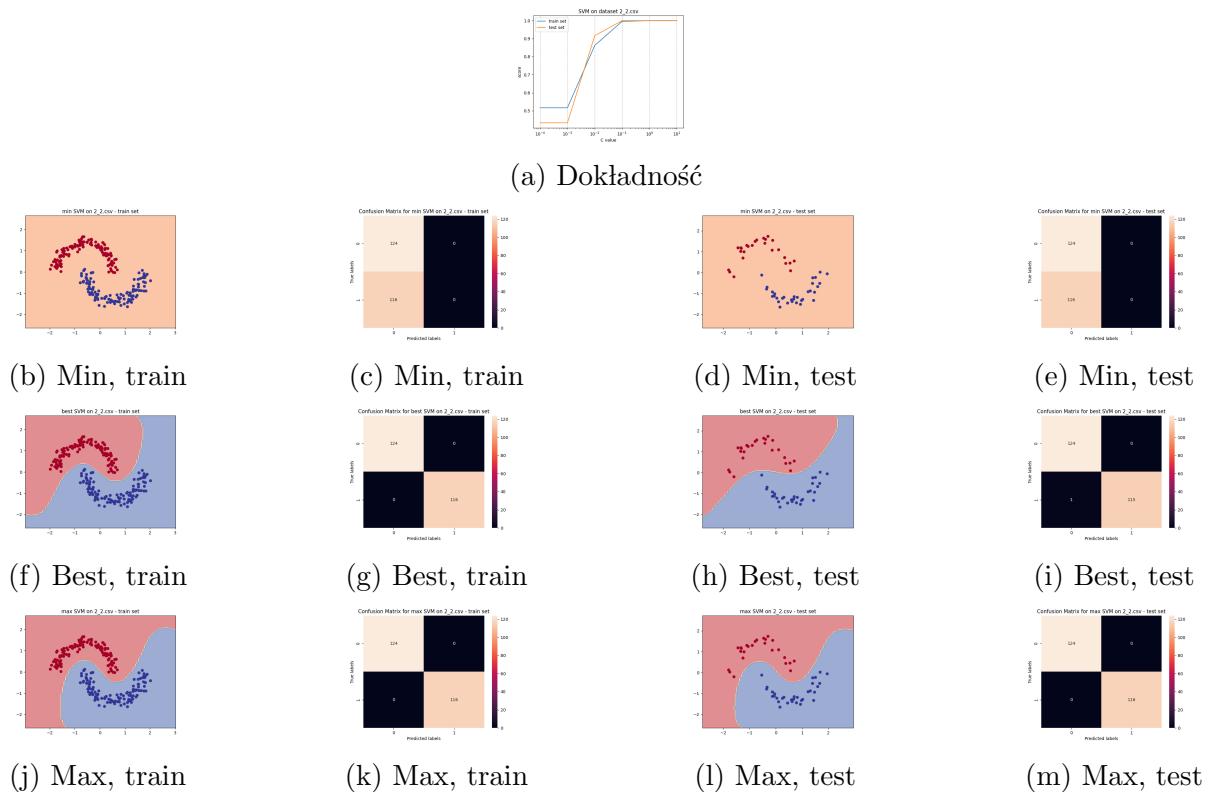
Rysunek 2: Eksperyment 1 — SVM



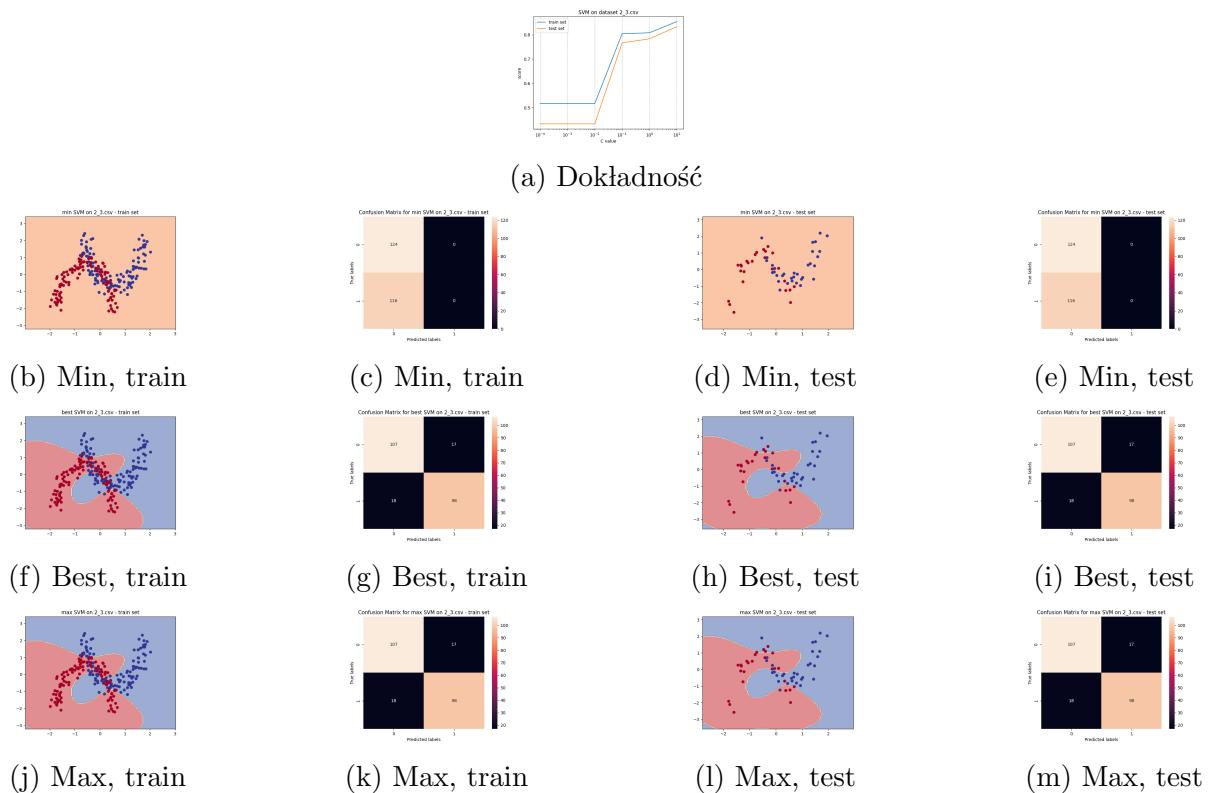
Rysunek 3: Eksperyment 2 — KNN na zbiorze 2.2



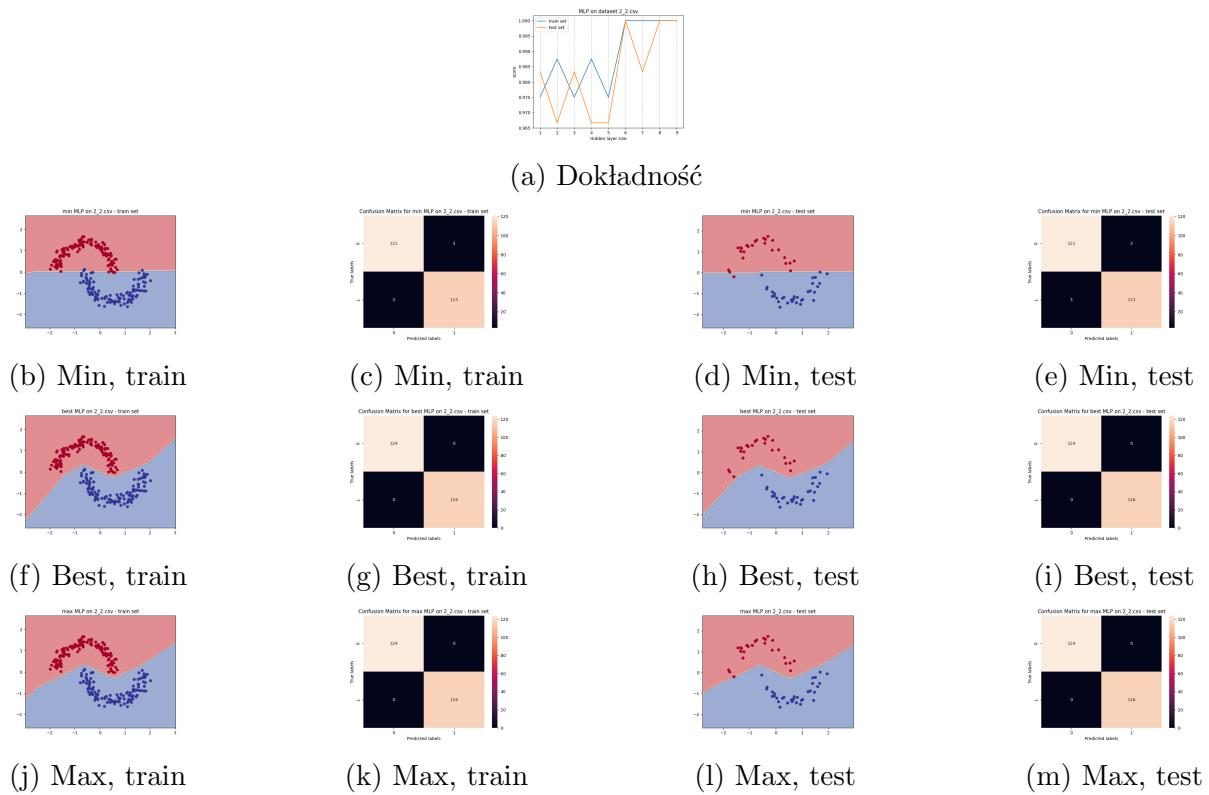
Rysunek 4: Eksperyment 2 — KNN na zbiorze 2.3



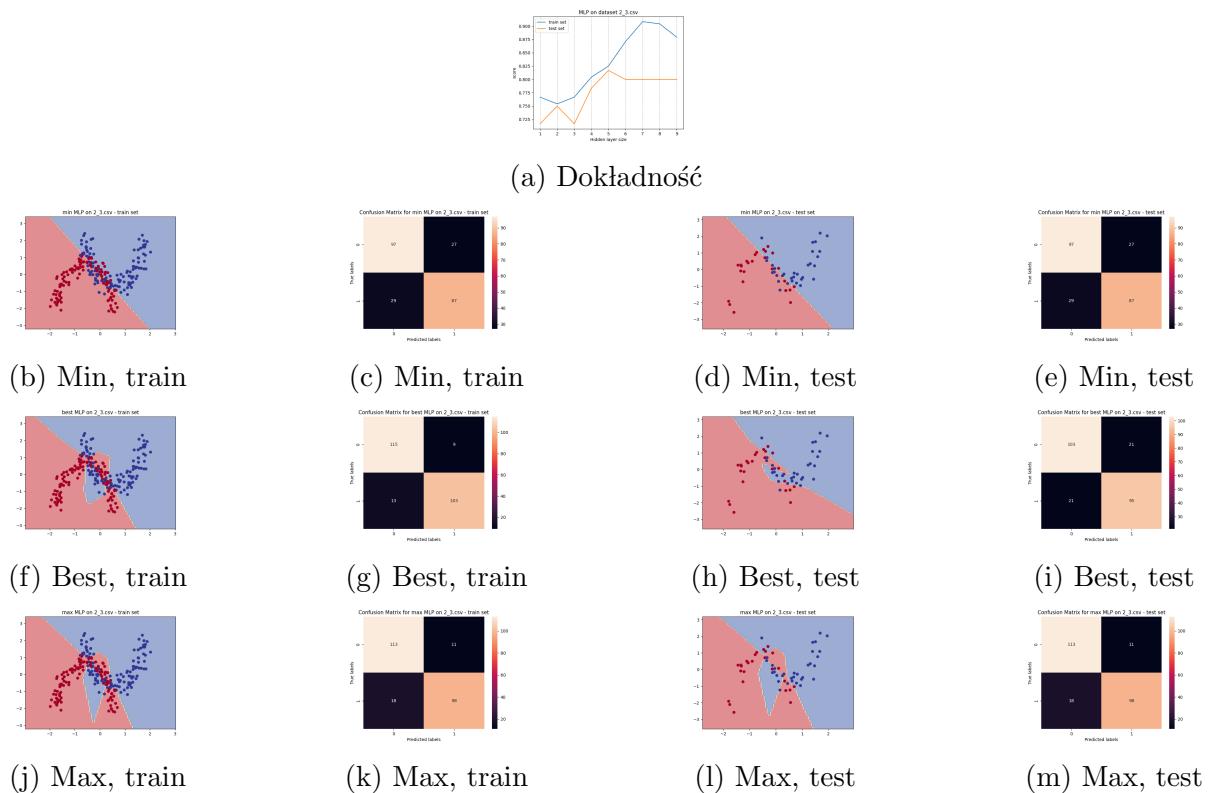
Rysunek 5: Eksperyment 2 — SVM na zbiorze 2_2



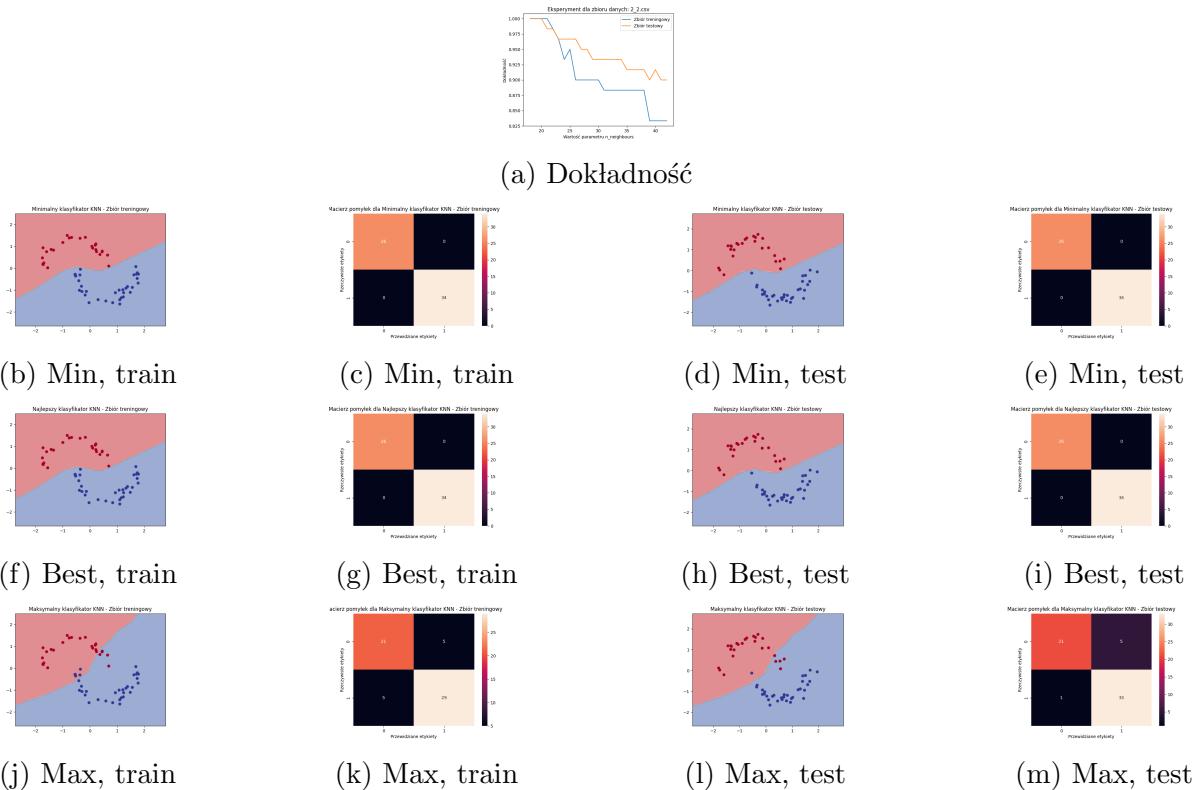
Rysunek 6: Eksperyment 2 — SVM na zbiorze 2_3



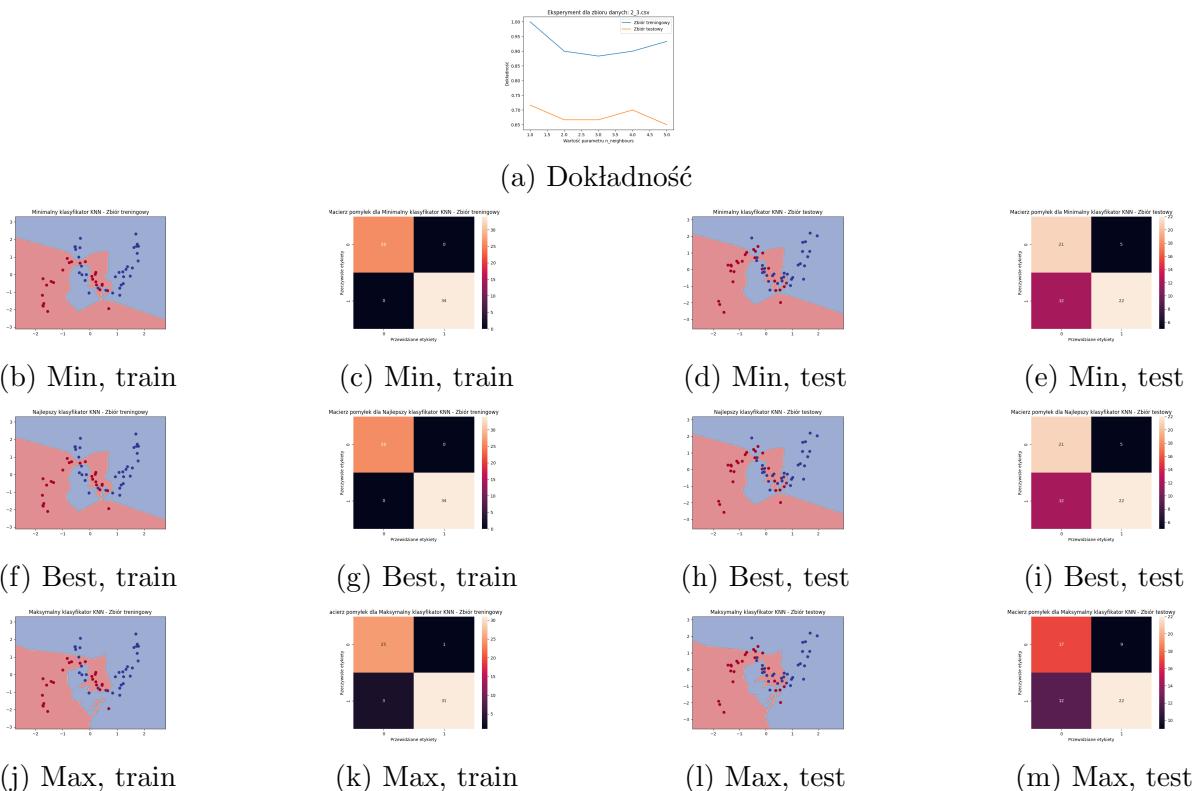
Rysunek 7: Eksperyment 2 — MLP na zbiorze 2_2



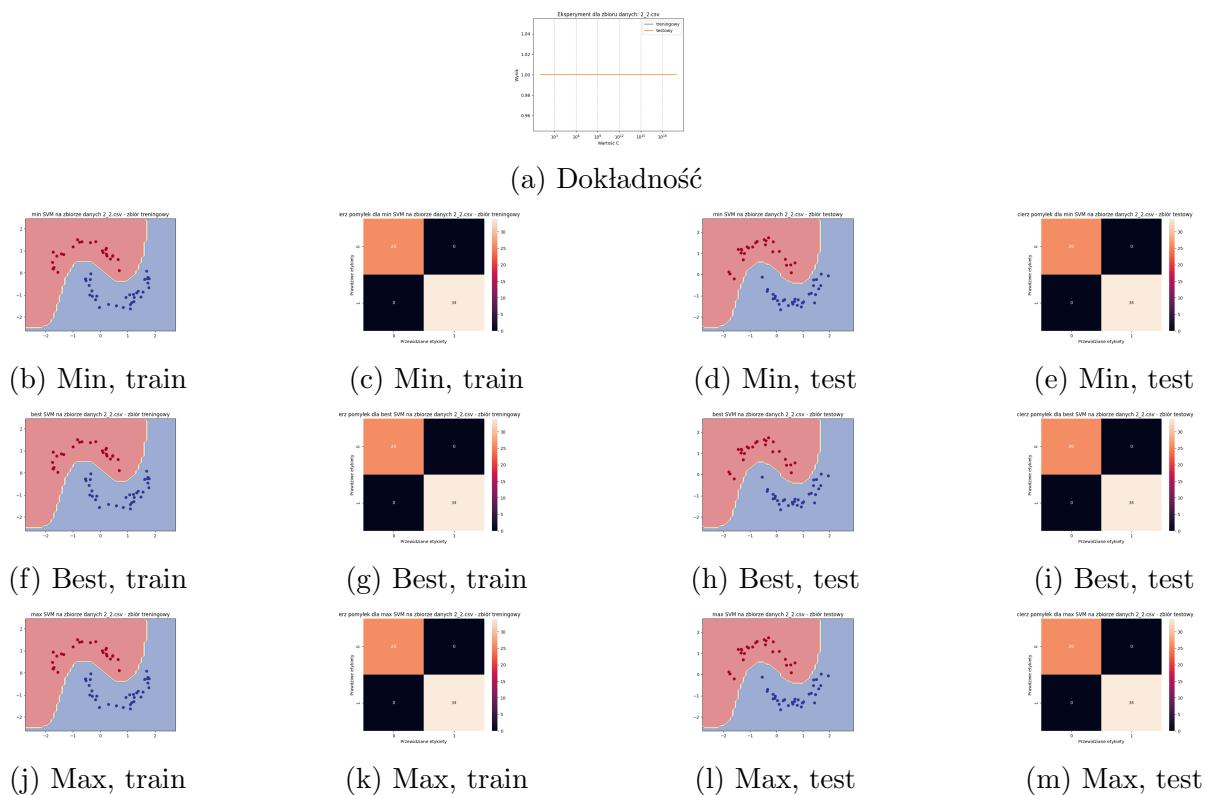
Rysunek 8: Eksperyment 2 — MLP na zbiorze 2_3



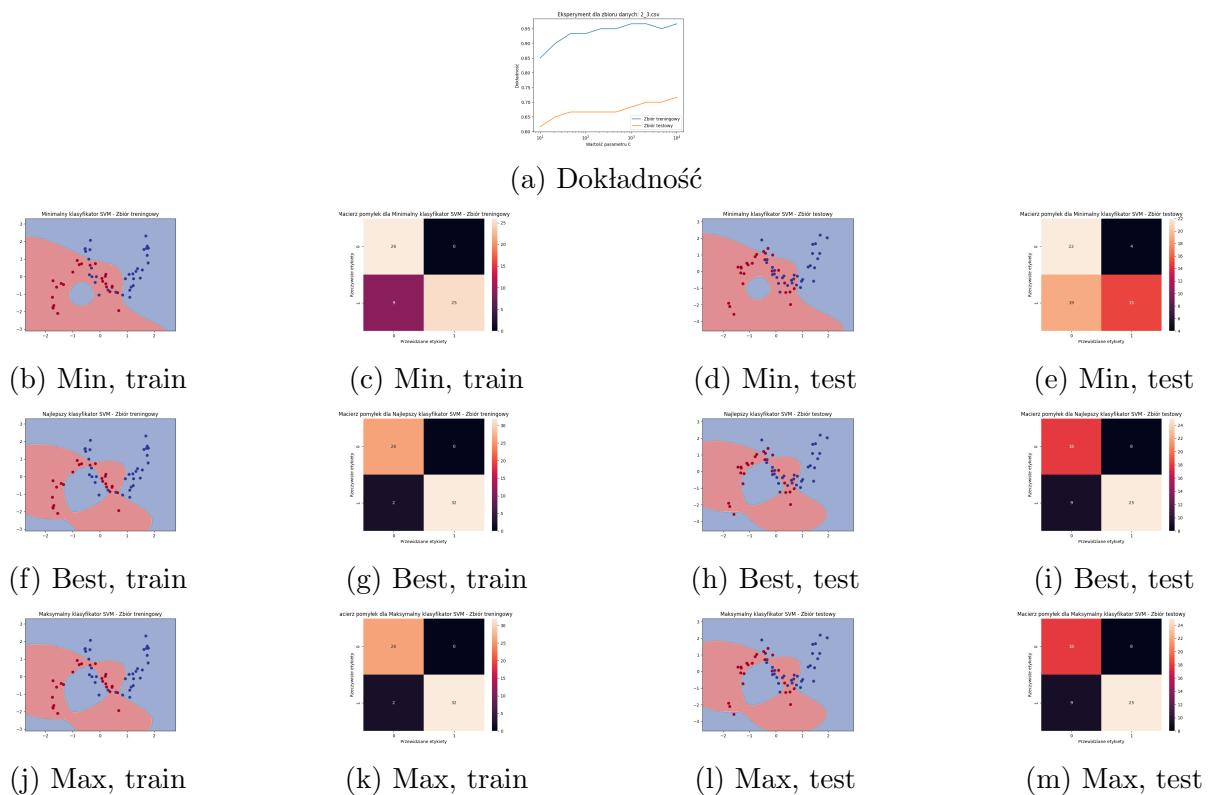
Rysunek 9: Eksperyment 3 — KNN na zbiorze 2_2



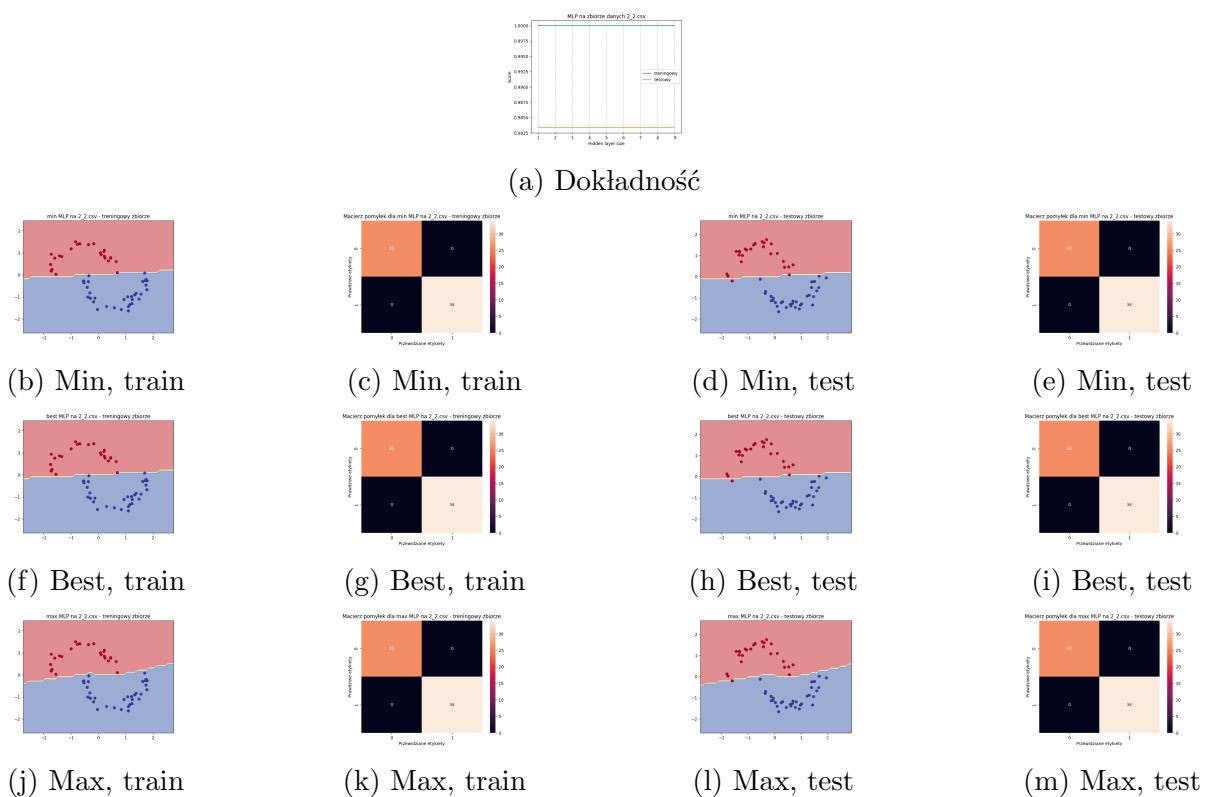
Rysunek 10: Eksperyment 3 — KNN na zbiorze 2_3



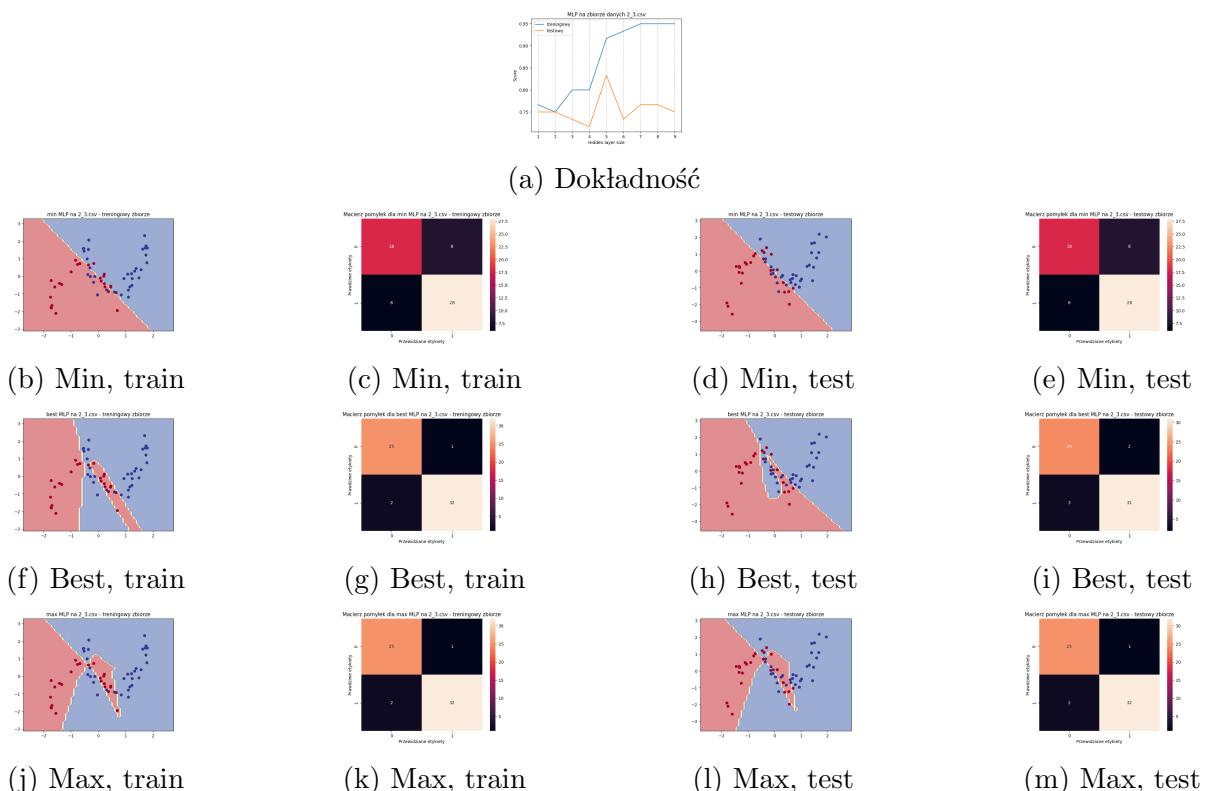
Rysunek 11: Eksperyment 3 — SVM na zbiorze 2_2



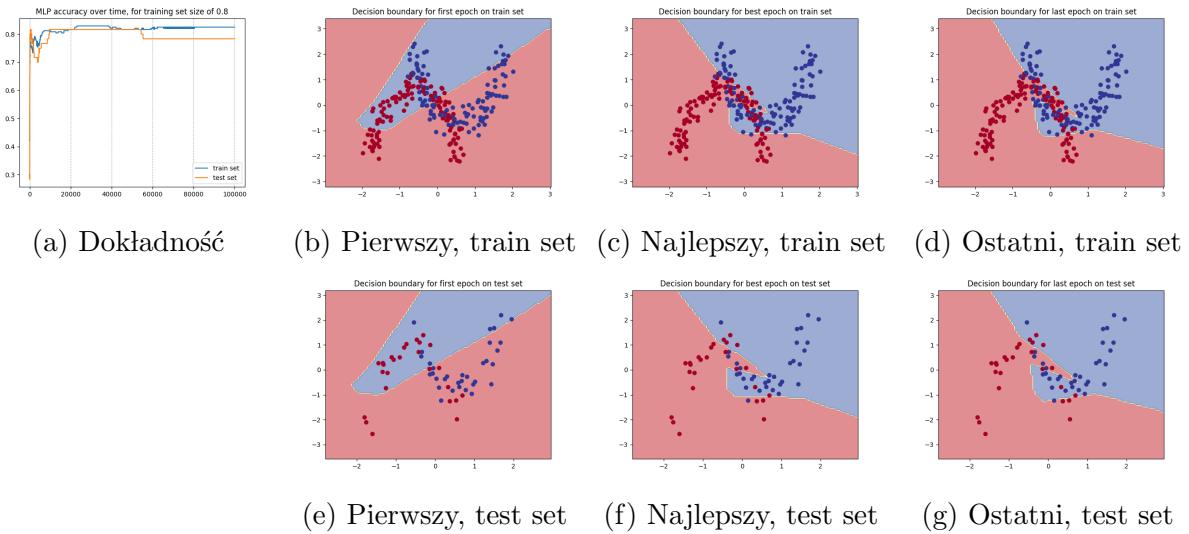
Rysunek 12: Eksperyment 3 — SVM na zbiorze 2_3



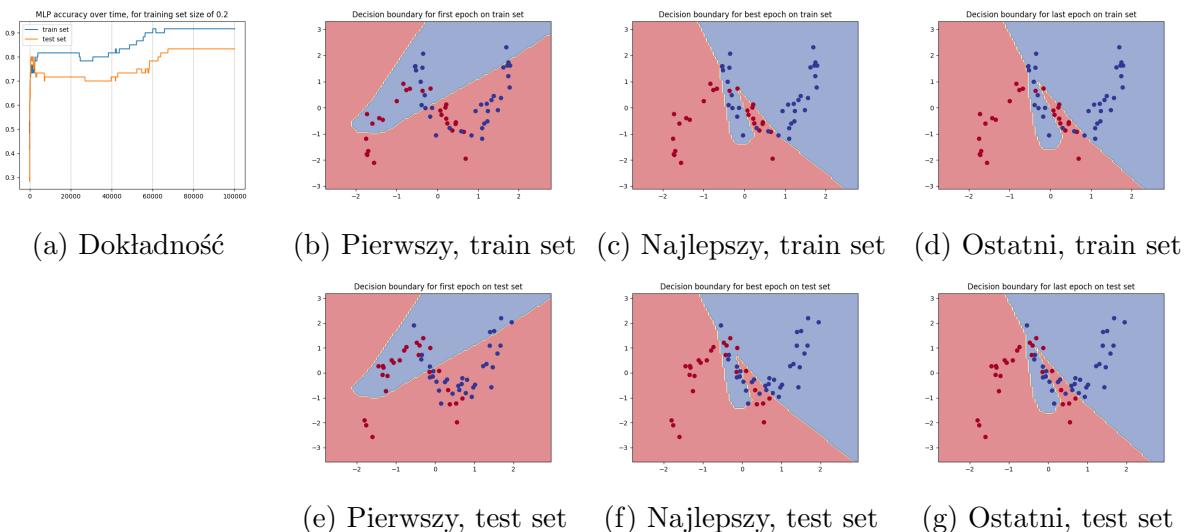
Rysunek 13: Eksperyment 3 — MLP na zbiorze 2_2



Rysunek 14: Eksperyment 3 — MLP na zbiorze 2_3



Rysunek 15: Trenowanie na 0.8 danych



Rysunek 16: Trenowanie na 0.2 danych

seed	pierwszy, train	najlepszy, train	najlepsza epoka	ostatni, train	pierwszy, test	najlepszy, test	najlepsza epoka	ostatni, test
12	0.587	0.895	38,320	0.891	0.633	0.800	44,930	0.800
31	0.516	0.820	32,325	0.812	0.433	0.733	722	0.716
16	0.516	0.841	44,575	0.829	0.433	0.783	11,244	0.750
2,353	0.545	0.795	77,067	0.795	0.766	0.850	74	0.783
314	0.308	0.791	24,176	0.783	0.400	0.783	60,806	0.783
6,523	0.541	0.779	25,312	0.775	0.416	0.766	404	0.716
634	0.620	0.837	9,400	0.825	0.450	0.733	329	0.700
23	0.491	0.783	275	0.779	0.566	0.766	38	0.700
802	0.516	0.920	75,135	0.920	0.433	0.800	39,816	0.800
134	0.379	0.808	94,967	0.808	0.333	0.733	1,268	0.700

Tabela 1: Trenowanie na 0.8 danych

seed	pierwszy, train	najlepszy, train	najlepsza epoka	ostatni, train	pierwszy, test	najlepszy, test	najlepsza epoka	ostatni, test
12	0.666	0.966	69,874	0.950	0.633	0.766	1,812	0.733
31	0.433	0.850	118	0.833	0.433	0.750	101	0.616
16	0.433	0.900	519	0.900	0.433	0.783	463	0.633
2,353	0.566	0.966	42,899	0.933	0.766	0.850	120	0.750
314	0.316	0.883	97,792	0.883	0.400	0.783	714	0.650
6,523	0.466	0.800	364	0.800	0.416	0.783	343	0.683
634	0.650	0.850	143	0.850	0.450	0.733	5,828	0.683
23	0.583	0.783	921	0.750	0.566	0.783	206	0.766
802	0.433	0.950	294	0.933	0.433	0.766	208	0.733
134	0.466	0.850	149	0.850	0.333	0.716	1,921	0.683

Tabela 2: Trenowanie na 0.2 danych

Eksperyment 1

Badania elastyczności w przypadku MLP wykazały lepsze rezultaty w przypadku funkcji ReLU. W przypadku funkcji identity możliwe było uzyskanie jedynie liniowej separacji, co, z racji na specyfike działania MLP, było oczekiwane. W przypadku zbioru 2_1 ograniczenie elastyczności klasyfikatora nie miało znaczenia, ponieważ dane były liniowo separowalne, natomiast przy pozostałych zbiorach danych, gdzie dane nie były liniowo separowalne, niewystarczajaca elastyczność klasyfikatora poskutkowała uzyskaniem gorszych wyników, niż w przypadku funkcji ReLU. Podobna sytuacja miała miejsce w przypadku klasyfikatora SVM, gdzie przy użyciu liniowego jadra, możliwe było uzyskanie jedynie liniowej separacji, co, ponownie, sprawdziło się tylko w przypadku zbioru 2_1. W toku eksperymentu ustalono, że w przypadku MLP, funkcja ReLU skutkuje wieksza elastycznością niż identity, a w przypadku SVM, jadro rbf jest bardziej elastyczne niż liniowe.

Eksperyment 2

Drugi eksperyment koncentruje się na wpływie elastyczności modelu na jego zdolności generalizacyjne. Zmiana hiperparametrów, takich jak liczba sasiadów dla klasyfikatora K-NN, parametr C dla klasyfikatora SVM oraz liczba neuronów w warstwie ukrytej dla sieci MLP, ma istotny wpływ na zdolności generalizacyjne modelu. Badanie tego wpływu na podziale danych na zbiór treningowy i testowy pozwala ocenić skuteczność klasyfikacji na nowych, nie widzianych wcześniej danych, co można zaobserować na przeprowadzonych testach. W eksperymencie KNN można zaobserwować zmiane wartości accuracy w stosunku do ilości `n_neighbours` na zbiorach 2_2 oraz 2_3, z czego wynika, że mała wartość `n_neighbours` daje szorstka granice decyzyjna, gdzie duża wartość może wskazywać na bardziej gładka, ale może również prowadzić do utraty szczegółów w danych. Wybór optymalnej wartości `n_neighbours` jest kluczowy dla osiągnięcia dobrych wyników z KNN. W eksperymencie SVM jadro RBF zapewnia lepsze uogólnienie niż jadro liniowe, zwłaszcza w przypadku złożonych danych. Zauważymy też, że parametr C kontroluje margines, a wyższe wartości C prowadzą do bardziej skomplikowanych granic decyzyjnych i potencjalnie lepszego uogólnienia. W eksperymencie MLP można zauważać, że liczba neuronów w warstwie ukrytej kontroluje złożoność modelu. Większa liczba neuronów może prowadzić do lepszego uogólnienia, ale może również powodować przeuczenie.

Eksperyment 3

Trzeci eksperyment rozważa wpływ zmiany hiperparametrów modelu na jego zdolności generalizacyjne w przypadku niewielkiej liczby danych treningowych równej 0.2. Przy ograniczonym zbiorze treningowym istotne staje się dobranie optymalnych hiperparametrów, aby uniknąć przeuczenia, co można zaobserwować na zmianie wykresu accuracy, gdzie w przypadku zbioru 2_2 przez pierwsze ukryte warstwy nie ma znaczenia, a w przypadku 2_3 ma to ogromny wpływ, co wynika to ze specyfikacji danego zbioru oraz jego wielkości do trenowania. W związku z tym, możemy zauważać, że mniejsza liczba danych treningowych może prowadzić do większej zmienności wyników i gorszego uogólnienia. Dzięki czemu, wiemy że ważne jest, aby użyć wystarczającej ilości danych treningowych, aby nauczyć model prawidłowo.

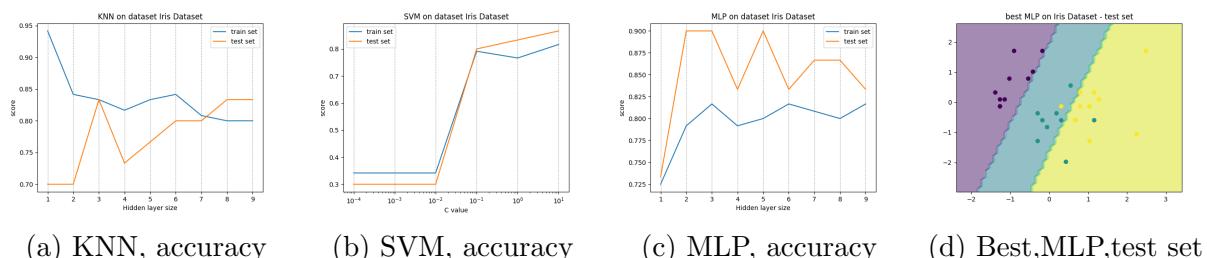
Eksperyment 4

Czwarty eksperyment badał postepy nauki MLP w czasie, jak również wpływ początkowej inicjalizacji na efekty uczenia. Eksperyment pokazał, że mniejsza ilość danych treningowych skutkuje w dostrzegalnie wolniejszej nauce. Jednakże, nie wpływa aż w tak dużym stopniu na maksymalna skuteczność klasyfikatora. Nawet klasyfikator trenowany na mniejszym zbiorze danych, był w stanie uzyskać wynik porównywalny z klasyfikatorem trenowanym na większym zbiorze, potrzebował jednak więcej epok, żeby osiągnąć taką skuteczność. Początkowa inicjalizacja klasyfikatora miała duży wpływ na skuteczność po 1 epoce, jak również na ilość epok potrzebną do osiągnięcia maksymalnej skuteczności. Nie miały jednak dużego wpływu na same wartości maksymalnej skuteczności. Można więc wyciągać wniosek, że, generalnie, dla większości stanów początkowych, klasyfikator predzej czy później będzie w stanie osiągnąć maksymalną skuteczność.

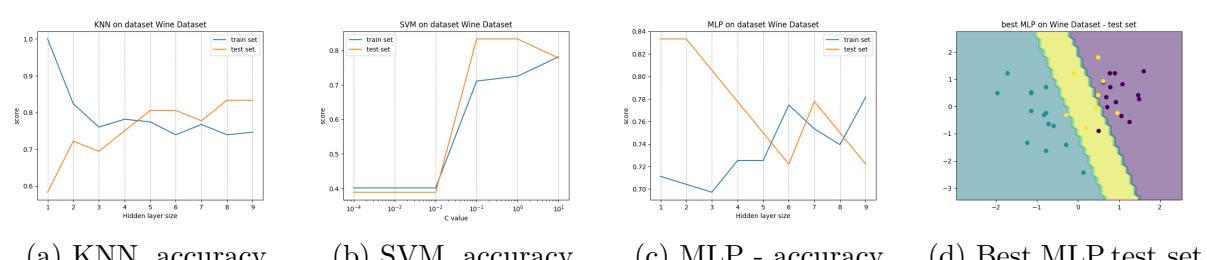
Wyniki przeprowadzonych eksperymentów na sztucznie wygenerowanych zbiorach danych dostarczają cennych informacji na temat działania metod klasyfikacji KNN, SVM i MLP w kontekście uczenia maszynowego. Należy jednak pamiętać, że wnioski te opierają się na specyfice sztucznie wygenerowanych danych i mogą nie w pełni odzwierciedlać zachowanie tych metod w przypadku rzeczywistych zbiorów danych.

Patrząc na wyniki, wiemy że w zbiorze Iris, w przypadku KNN, która jest prosta intuicyjna metoda, dobrze sprawdza się w przypadku małych zbiorów danych i prostych zależności między cechami a klasami. Należy jednak koniecznie zwrócić uwagę na wpływ wyboru parametru `n_neighbours`. Wiemy też, że KNN może być wrażliwy na outliersy i skale cech, dlatego przed zastosowaniem tej metody należy rozważyć zastosowanie odpowiednich technik preprocesowania danych. W stosunku do SVM - biorąc pod uwagę liniowa separowalność danych, SVM z jadrem liniowym może zapewnić wysoką dokładność klasyfikacji. Co więcej, jadro RBF może również być użyteczne, jeśli dane nie są idealnie liniowo separowalne. W eksperymencie MLP, może on modelować złożone zależności między cechami, co może być pomocne w przypadku, gdy dane nie są liniowo separowalne, jednak trzeba uważać na przeuczenie.

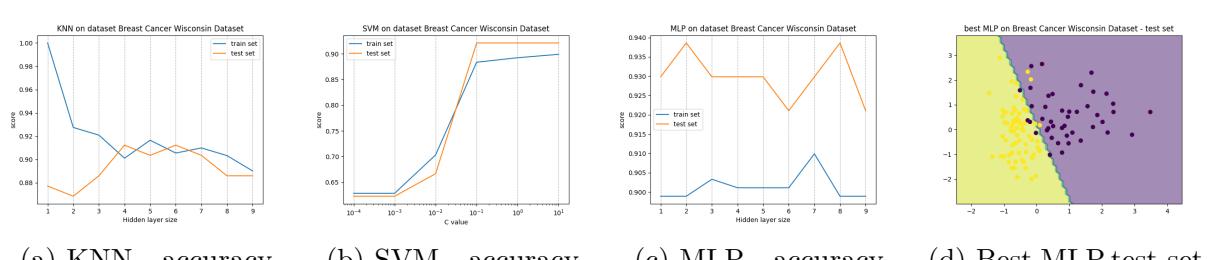
W zbiorze Breast i Wine wnioski z eksperymentu KNN są identyczne. W eksperymencie SVM dla setu Wine i Breast - SVM z jadrem RBF może być dobrym wyborem do klasyfikacji wina, ponieważ może modelować nieliniowe zależności między cechami, natomiast parametr C i wybór jadra mogą mieć istotny wpływ na dokładność klasyfikacji. W eksperymencie MLP - może być poteżna metoda klasyfikacji wina i klasyfikacji nowotworów piersi, ale wymaga starannego doboru hiperparametrów, aby uniknąć przeuczenia. Funkcja aktywacji ReLU może być korzystna w obu przypadkach.



Rysunek 17: Wyniki eksperymentów na zbiorze Iris



Rysunek 18: Wyniki eksperymentów na zbiorze Wine



Rysunek 19: Wyniki eksperymentów na zbiorze Breast Cancer Wisconsin