WSI – ćwiczenie 4.
Algorytm ID3

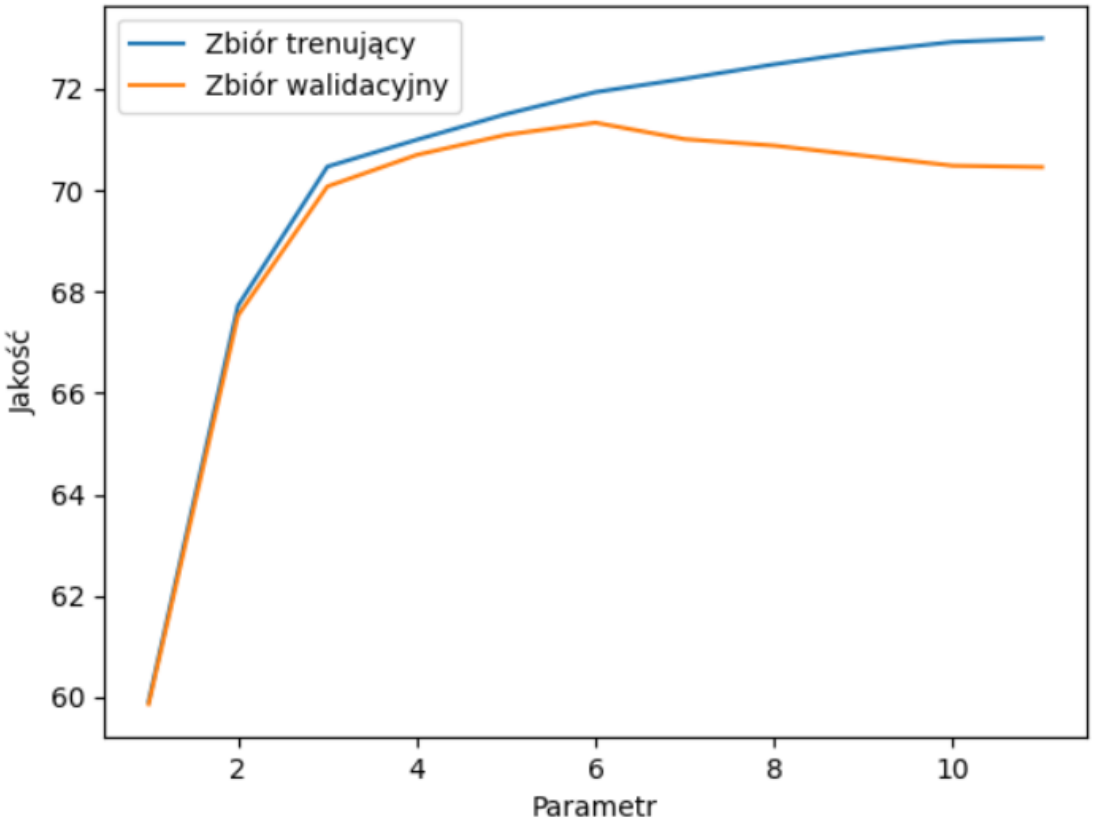
1. Cel ćwiczenia

Celem ćwiczenia jest implementacja drzew decyzyjnych tworzonych algorytmem ID3 z ograniczeniem maksymalnej głębokości drzewa. Następnie należy wykorzystać stworzony algorytm do stworzenia i zbadania jakości klasyfikatorów dla zbioru danych Cardio Vascular Disease Detection.

2. Metodyka badań

Dyskretyzacja niektórych parametrów (wiek , waga, wzrost, ciśnienie skurczowe, ciśnienie rozkurczowe) polega na dzieleniu całkowitym każdej wartości przez ustaloną stałą, dzięki temu zbiór unikalnych wartości atrybutów jest mniejszy.
Zbiór trenujący stanowi 70% danych. Zbiór walidacyjny 20 %, a zbiór testujący 10%.
Aby znaleźć najlepszą wartość parametru maksymalnej głębokości przeprowadzam badania na zbiorze walidacyjnym. Sprawdzam parametry głębokości [1;11], a następnie wybieram ten z najlepszym wynikiem.
Dla modelu z wybranym parametrem oceniam jakość modelu na zbiorze testującym.
Oceną danego modelu jest skuteczność wykrywania choroby podana w procentach.

3. Wybór maksymalnej głębokości



Na powyżej pokazanym wykresie widać, że parametrem maksymalnej głębokości o najlepszym wyniku jest parametr równy 6.

4. Ocena modelu

Oceny na podstawie 10 prób modelu o głębokości 6:

Minimalna ocena	Średnia ocena	Maksymalna ocena
70.319%	71.088%	71.449%

5. Wnioski

Wykres pokazuje, że przy maksymalnej głębokości przeszukiwania większej od 6 skuteczność algorytmu zaczyna zmniejszać się. Może wynikać to z przeuczenia, gdzie ostatnie atrybuty mają znikomy

wpływ na występowanie choroby i wręcz przeszkadzają w wykrywaniu jej.
Największy wzrost jakości można zauważyć podczas zmiany parametru głębokości z 1 do 3. Oznacza to, że największy wpływ na pojawienie się choroby mają trzy atrybuty.

6. Dodatkowe informacje

Zbiór danych jest dostępny pod linkiem (https://www.kaggle.com/datasets/bhadaneeraj/cardio-vascular-disease-detection).

W algorytmie wykorzystałem biblioteki:

- numpy – wykorzystywana w obliczeniach

- pandas – wykorzystywana do przechowywania i przetwarzania danych

- scikit-learn train-test split function – wykorzystywana do podziału danych na zbiory

- matplotlib – tworzenie wykresów

Łukasz Wójcicki