

WSI Ćwiczenie 4

Algorytm ID3

Autor:

Jakub Kowieski

Przedmiot:

WSI, semestr: 21Z

Numer Albumu:

310765

Celem ćwiczenia jest implementacja drzew decyzyjnych tworzonych algorytmem ID3 z ograniczeniem maksymalnej głębokości drzewa.

Założenia:

1. **row** - jest to tuple ([atrybuty], klasa)
2. **dyskretyzacja cech** - każda cecha jest zamieniana z liczby ciągłej na stringa.
3. Podział 150 elementowego zbioru irisis:
 - Grupa ucząca - 100 elementów
 - Grupa walidacyjna - 25 elementów
 - Grupa testująca - 25 elementów
4. Podczas predykcji, jeżeli nie ma węzła o danej wartości cechy, wtedy jest wybierana klasa najczęściej występująca
5. Głębokość dla testów od 0 do 4

Posiadanie modułów:

- numpy
- sklearn
- matplotlib

Testowanie i symulacje:

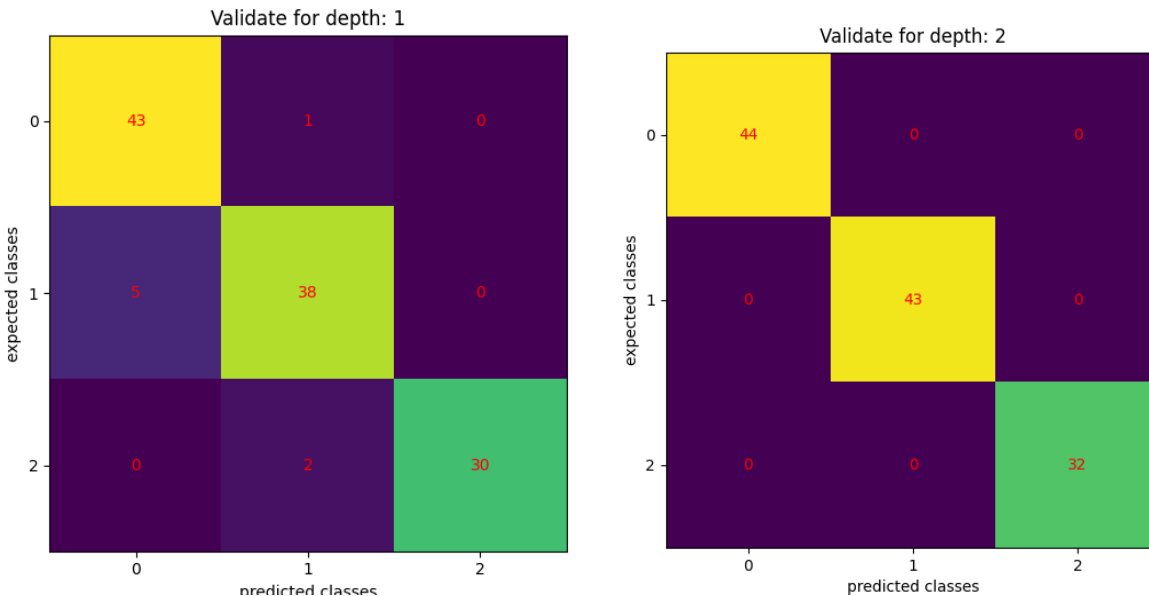
[python3 run.py](#) - uruchamia proces tworzenia i wybierania najlepszego drzewa oraz testowanie go

[python3 test.py](#) - uruchamia test czy predykcja zachowuje tak jak powinna dla drzew.

Test działania poprawności drzew:

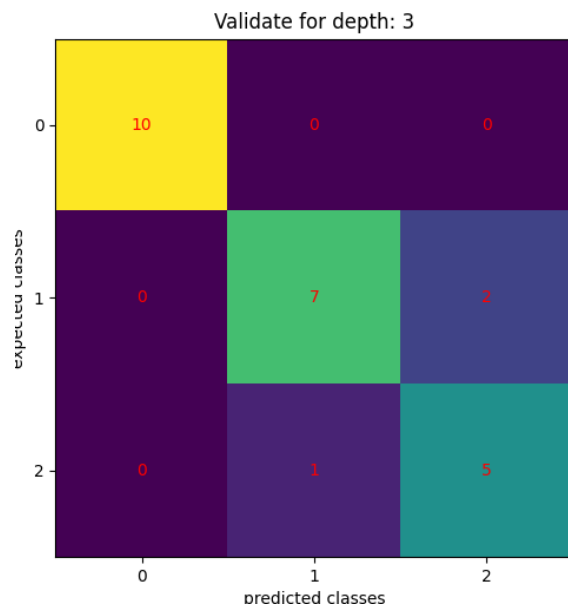
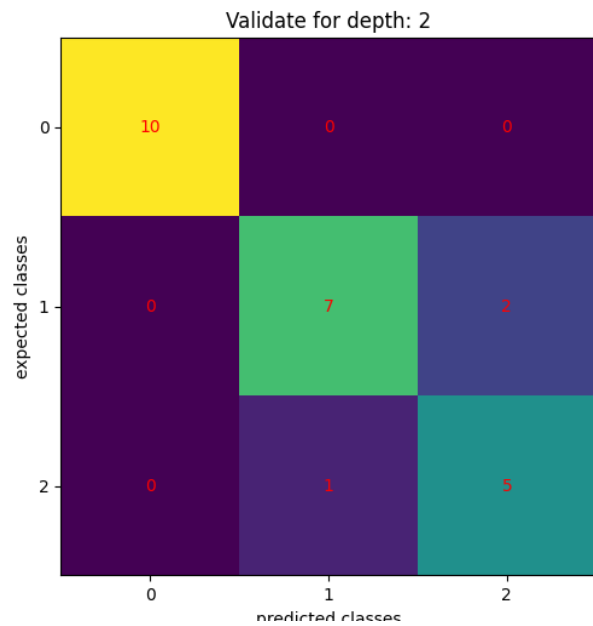
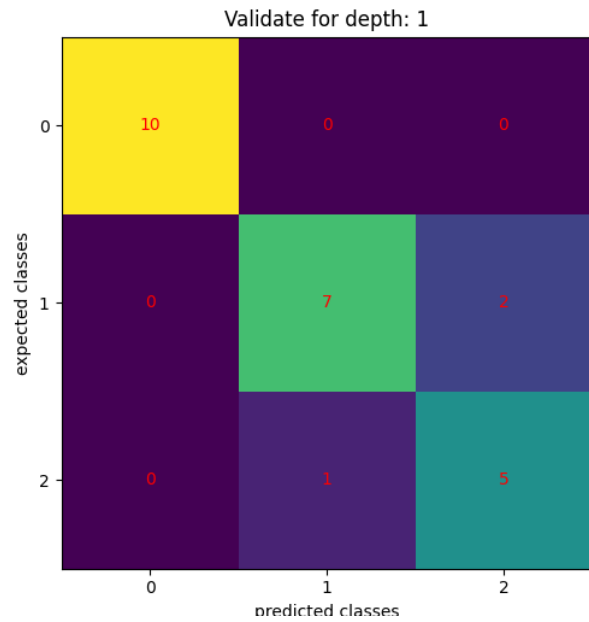
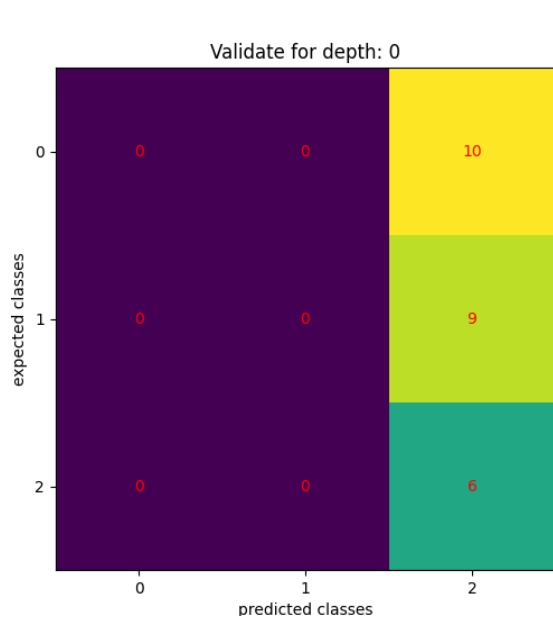
Jeżeli algorytm działa poprawnie to dla różnych głębokości algorytm dla odpowiednich danych powinien dawać różne wyniki. Dlatego w ramach testów przyjmujemy grupy walidacyjne i uczące za ten sam zbiór by być pewni, że dla odpowiedniej gęstości wszystkie predykcje powinny być poprawne.

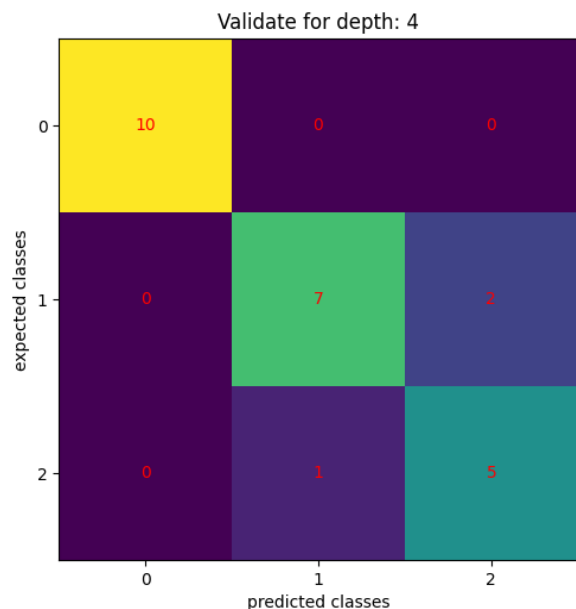
(wykresy z *python3 test.py*)



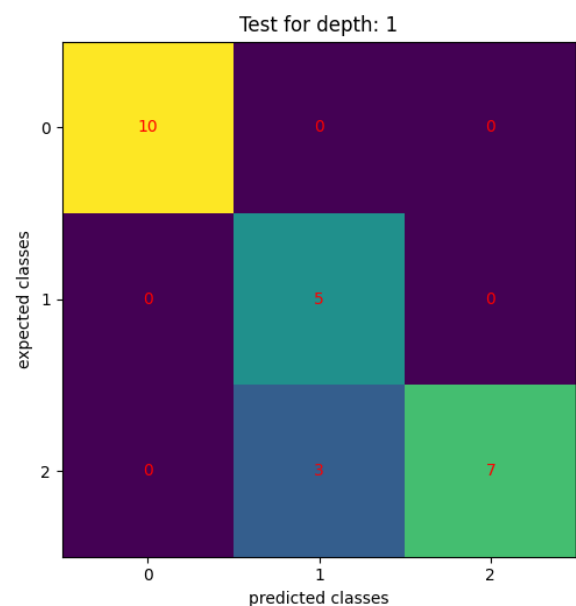
Możemy zaobserwować, że już dla głębokości 2 wszystkie predykcje są poprawne. Co znaczy że drzewo dla głębokości 1 jest zbyt niskie, aby otrzymać wartość klasy i musi wybrać klasę najczęściej występującą.

Walidacja i testowanie drzew:





Testowanie drzewa:



Wnioski:

Algorytm, działa sensownie z poprawnością 88% dla grupy testowej. Można zauważyć że głębokość ma znaczenie tylko na skoku z gęstości 0 do 1, a dla innych wykresy są takie same. Można to jednak uzasadnić tym, że próbek danych nie jest dużo oraz mają dużą ilość unikatowych wartości cech.