

WSI - ćwiczenie 3.

Regresja i klasyfikacja

12 listopada 2025

1 Sprawy organizacyjne

1. Ćwiczenie realizowane jest samodzielnie.
2. Ćwiczenie wykonywane jest w języku Python.
3. Ćwiczenie powinno zostać oddane najpóźniej do 24.11.2025r. W ramach oddawania ćwiczenia należy zademonstrować prowadzącemu działanie kodu oraz utworzyć merge/pull request (z kodem oraz raportem), który prowadzący będzie mógł komentować.
4. Rozwiązanie ćwiczenia powinno być zgodne z szablonem z repozytorium <https://gitlab-stud.elka.pw.edu.pl/jlyskawa/wsi-template>.
5. Kod oraz raport powinien być umieszczony na repozytorium `gitlab-stud.elka.pw.edu.pl`, a pull request utworzony do konta prowadzącego: `@djagodzi` (dla nazwisk A-N) i `@pzawist2` (dla nazwisk O-Z)
6. Raport powinien być w postaci pliku `.pdf`, `.html` albo być częścią notebooka jupyterowego. Powinien zawierać opis eksperymentów, uzyskane wyniki wraz z komentarzem oraz wnioski.
7. Na ocenę wpływa poprawność oraz jakość kodu i raportu.
8. Można korzystać z pakietów do obliczeń numerycznych, takich jak *numpy*.
9. Implementacja algorytmu powinna być ogólna. W szczególności powinna być możliwa do zastosowania dla dowolnego zbioru danych o dyskretnych wartościach atrybutów.
10. Można skorzystać z pakietów *pandas* i *scikit-learn* w celu załadowania zbioru danych oraz jego podziału na części.

2 Ćwiczenie

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

Następnie należy wykorzystać przygotowane rozwiązanie do stworzenia i zbadania jakości klasyfikatorów dla zbioru danych Cardio Vascular Disease Detection. Zbiór danych znajduje się pod linkiem (<https://www.kaggle.com/datasets/bhadaneeraj/cardio-vascular-disease-detection>).

Kolumna *cardio* oznacza klasę do klasyfikacji. Należy znaleźć taką wartość parametru maksymalnej głębokości, która da najlepszy wynik.

Część atrybutów w zbiorze danych nie jest dyskretna - wiek (*age*), waga (*weight*), wzrost (*height*), ciśnienie skurczowe (*ap_hi*), ciśnienie rozkurczowe (*ap_lo*). Należy je zdyskretyzować poprzez wybrane przez siebie podzielenie wartości na zakresy.

Implementacja powinna obsługiwać sytuację, w której w zbiorze trenującym nie ma wszystkich wartości jakiegoś atrybutu.

Należy pamiętać o podziale danych na zbiory trenujący, walidacyjny i testowy. Można użyć w tym celu gotowych funkcji.

Zadanie dodatkowe. Na podstawie przygotowanego rozwiązania należy zaimplementować las losowy. Następnie należy wykorzystać go do przygotowania klasyfikatorów na tym samym zbiorze, porównując ich jakość do pojedynczego drzewa decyzyjnego dla różnej liczby drzew w lesie.