Jan Mogielski 303899

Laboratorium 4
Regresja i klasyfikacja

4 maja 2023

Spis treści

1. Cel laboratorium

Celem laboratorium było napisanie kodu w języku Programowania Python i zaimplementowanie drzewa decyzyjnego za pomocą algorytmu ID3. Następnie za pomocą algorytmu ID3 miałem zbadać jakość klasyfikatorów dla zbioru danych Cardio Vascular Disease Detection i określić najlepszą głębokość drzewa, która da nam najlepszy wynik.

2. Przygotowanie danych

Wykorzystywany zestaw danych jest w formacie csv, którego separatorem jest ";". Część danych nie jest w postaci dyskretnej, czyli są w postaci ciągłej, a przetwarzanie takich danych jest problematyczne dla naszego algorytmu. W tym celu zamieniamy wartości ciągłe: age, weight, height, ap_hi oraz ap_lo na wartości dyskretne korzystając z funkcji "cut" wbudowanej w bibliotekę "Pandas". W ten sposób podzieliliśmy ciągłe dane na przedziały, którym przypisaliśmy dyskretne dane, np: 0, 1, 2.

3. Podział danych

Przy podziale danych wykorzystujemy gotową funkcję "train_test_split", która dzieli nasze dane na dane do trenowania, testowania i ewaluacji algorytmu ID3. Zazwyczaj podział danych w dziedzinie uczenia maszynowego to 80:20, ale w moich testach wykorzystałem różne wartości.

4. Implementacja

Główny kod z algorytmem ID3 składa się z kilku istotnych funkcji:

- _entropy, która oblicza entropię dla naszych danych,
- _information_gain, funkcja obliczająca zysk informacji dla danego atrybutu, zwraca różnicę pomiędzy obliczoną entropią, a entropią ważoną,
- _build_tree - rekursywnie buduje drzewo, wybierając atrybuty, które posiadają najwięcej informacji. Jeśli osiągniemy maksymalną głębokość przeszukiwania albo nie mamy już do sprawdzenia żadnych danych to zwracamy informację o tym, że dotarliśmy do liścia drzewa.
- fit - funkcja służąca do znalezienia idealnego balansu pomiędzy danymi wejściowymi, a docelowymi. Wszystko poza polem "cardio" jest danymi, które dopasowujemy do "wyjścia".
- _classify - następuje klasyfikacja "wyjścia" na podstawie danych, które nie znajdują się w polu "cardio", ponieważ wynik tego pola nas interesuje.
- predict - dochodzi do predykcji "wyjścia" przez algorytm. Na podstawie naszych atrybutów i danych do trenowania, stwierdzamy czy takie dane w atrybutach dadzą nam wyśjcie X lub Y, a może Z.

5. Testy działania algorytmu ID3

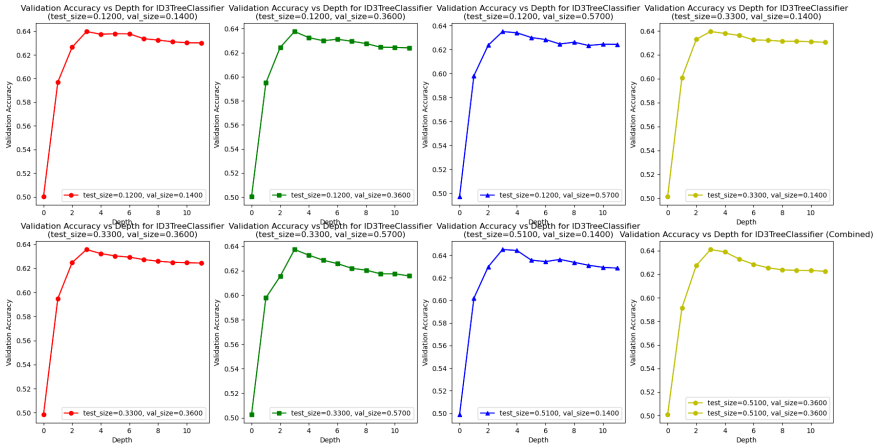
Działanie algorytmu w dużej mierze zależy od głębokości przeszukiwania. Dla małych głębokości, czyli max.5 działanie algorytmu jest bardzo szybkie i zajmuje niewiele czasu, im większa głębokość przeszukiwania tym więcej czasu potrzeba na zakończenie jego działania. Następnym istotnym czynnikiem jest wielkość data set'u, dla małych zestawów danych otrzymamy krótkie czasy działania, ale decyzje podjęte przez algorytm nie będą idealne ze względu na brak pełnej wiedzy na dany temat. Im większy zestaw danych tym dokładniej znajdziemy wzorce i lepiej przewidzimy wynik końcowy, ale potrzebujemy więcej czasu na jego przetworzenie.

W swoich testach sprawdziłem również wpływ podziału danych na zestaw trenujący, testowy i walidacyjny. Osobiście spodziewałem się większych różnic w końcowych wynikach, a różnice były bardzo małe. W przypadku uczenia maszynowego i trenowania modeli do rozpoznawania intencji wpływ podziału danych był dużo bardziej zauważalny na jego końcową skuteczność.

Działanie algorytmu przeprowadziłem na głębokościach od 0 do 12 wraz z różnymi podziałami danych. Działanie algorytmu ewidentnie jest najlepsze dla głębokości o wartości 3 i 4.

Poniżej prezentuję dokładne wyniki z przeprowadzonych testów:

test_size	val_size	max_depth	accuracy
0.1200	0.1400	3	0.6397
0.1200	0.3600	4	0.6376
0.1200	0.5700	3	0.6350
0.3300	0.1400	4	0.6395
0.3300	0.3600	3	0.6357
0.3300	0.5700	3	0.6373
0.5100	0.1400	4	0.6451
0.5100	0.3600	4	0.6410



Rys. 1. Działanie algorytmu ID3 dla różnych głębokości

6. Wnioski

Algorytm ID3 został poprawnie zaimplementowany nasz kod przeszukuje atrybuty w poszukiwaniu takich elementów, które zapewnią mu maksymalną ilość informacji. W ten sposób tworzymy drzewo, które przy korzeniu ma najważniejsze i najbardziej znaczące informacje w naszej klasyfikacji, a resztę atrybutów niżej. Prawdopodobnie można byłoby lepiej oczyścić zestaw danych w celu większej dokładności albo pogrupować na mniejsze podgrupy, aby mieć większe informacje na temat danej grupy wiekowej albo większą dokładność dla jednego z atrybutów.