Wprowadzenie do sztucznej inteligencji Ćwiczenie 4

Dominika Boguszewska

Semestr 23Z

1 Polecenie

Zaimplementować klasyfikator *ID3* (drzewo decyzyjne). Atrybuty nominalne, testy tożsamościowe. Podać dokładność i macierz pomyłek na zbiorach: *Breast cancer* i *mushroom*. Dlaczego na jednym zbiorze jest znacznie lepszy wynik niż na drugim? Do potwierdzenia lub odrzucenia postawionych hipotez konieczne może być przeprowadzenie dodatkowych eksperymentów ze zmodyfikowanymi zbiorami danych. Sformułować i spisać wnioski.

1.1 Poniżej kilka wskazówek ogólnych do tego ćwiczenia

- Atrybuty nominalne każdy atrybut może przyjmować jedną z kilku dozwolonych wartości, zakładamy, że wartość atrybutu to napis, np. "kot", "a", "20-34", "¿40".
- Testy tożsamościowe jeżeli atrybut testowany w danym węźle ma np. 3 dozwolone wartości, np. a, b, c, to z węzła tego wychodzą 3 krawędzie oznaczone: a, b, c.
- Na tym ćwiczeniu klasyfikator trenuje się na zbiorze trenującym, a ocenia jego jakość na zbiorze testującym. Należy losowo podzielić zbiór danych na trenujący i testujący w stosunku 3:2.
- Jeżeli zbiór danych zawiera numery lub identyfikatory wierszy to należy je wyrzucić nie chcemy uczyć się identyfikatorów wierszy.
- Brakujące wartości atrybutów taktujemy jako wartość, np. jeżeli symbol '?' oznacza brakującą wartość, a symbole 'a', 'b' wartości normalne, to z naszego punktu widzenia mamy 3 wartości normalne (fachowo: 3 wartości atrybutu): 'a', 'b', '?'.
- Tak naprawdę to nie musimy rozumieć dziedziny problemu na wejściu mamy napisy, na wyjściu napisy, nie ważne czy klasyfikujemy sekwencje DNA, grzyby, czy samochody.
- Nazwa pliku ze zbiorem danych jest parametrem algorytmu klasyfikacji, kod klasyfikatora powinien być w stanie obsłużyć inny zbiór danych o tym samym rozkładzie kolumn (czyli nie należy wpisywać wartości atrybutów "na sztywno" w kodzie).
- W repozytorium ze zbiorami danych zwykle w plikach "names" jest napisane, który atrybut to klasa (czyli wartości której kolumny mamy się nauczyć przewidywać).

2 Testy przeprowadzone na zbiorze breast cancer

2.1 Dla niepotasowanego zbioru

Dla niepotasowanego zbioru wystarczył jeden przeprowadzony test, ponieważ dane zbioru nie są zmieniane.



Rysunek 1: Wyniki przeprowadzonych testów



Rysunek 2: Rozkład wyników oczekiwanych i przewidzianych

Dla niepotasowanego zbioru dokładność przewidzeń jest bardzo niska, ponieważ w zbiorze breast cancer najpierw są podane wszystkie dane z klasy no-recurrence-events, a dopiero potem z klasy recurrence-events. Z tego powodu wszystkie dane ze zbioru uczącego należą do pierwszej klasy, a większość danych ze zbioru testującego do drugiej klasy, której drzewo nie nauczyło się przewidywać.

2.2 Dla potasowanego zbioru

Przeprowadziłam 20 testów na zbiorze *breast cancer*, który zaraz po odczytaniu z pliku został przetasowany.

		!	!
Liczba wykonanych przewidzeń	Liczba poprawnych przewidzeń	Liczba niepoprawnych przewidzeń	Dokładność przewidzeń
100		37	0.63
93	60	33	0.645161
101	70	31	0.693069
98	74	24	0.755102
93			0.645161
194			0.625
99		32	0.676768
95		36	0.621053
99	71	28	0.717172
103	73	30	0.708738
107		47	0.560748
101		36	0.643564
101	70	31	0.693069
104	65	39	0.625
100	64	36	0.64
95	65	38	0.684211
99		33	0.666667
101		29	0.712871
104		38	0.634615
110	76	34	0.690909

Rysunek 3: Wyniki przeprowadzonych testów



Rysunek 4: Rozkład wyników oczekiwanych i przewidzianych z ostatniego testu

Jak możemy zauważyć, poprawność przewidzeń poprawiła się po potasowaniu zbioru, dzięki któremu zarówno w zbiorze uczącym jak i testującym mogły znaleźć się dane należące do obu klas. Jednakże nadal nie zostało zbudowane drzewo idealne, które miałoby dokładność przewidywań bliską 100%.

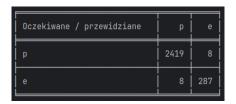
3 Testy przeprowadzone na zbiorze *mushroom*

3.1 Dla niepotasowanego zbioru

Dla niepotasowanego zbioru wystarczył jeden przeprowadzony test, ponieważ dane zbioru nie są zmieniane.



Rysunek 5: Wyniki przeprowadzonych testów



Rysunek 6: Rozkład wyników oczekiwanych i przewidzianych

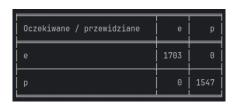
Jak możemy zauważyć, dla niepotasowanego zbioru *mushroom* nie pojawił się ten sam problem, który pojawił się w przypadku niepotasowanego zbioru *breast cancer*. Tutaj już na starcie obie klasy znajdowały się zarówno w zbiorze uczącym jak i testującym. Ponad to zostało zbudowane drzewo, które przewidziało prawie wszystkie dane poprawnie.

3.2 Dla potasowanego zbioru

Przeprowadziłam 20 testów na zbiorze *mushroom*, który zaraz po odczytaniu z pliku został przetasowany.

Liczba wykonanych przewidzeń	Liczba poprawnych przewidzeń	Liczba niepoprawnych przewidzeń	 Dokładność przewidzeń
3250	3250	0	1
3250	3250	9	1
3250	3250	9	1
3250	3246	4	0.998769
3250	3250	9	1
3250	3250	9	1
3250	3250	9	1
3250	3250	9	1
3250	3250	9	1
3250	3250	0	1
3250	3250	9	1
3250	3250	9	1
3250	3250	9	1
3250	3250	9	1
3250	3250	9	1
3250	3250	9	1
3250	3250	θ	1
3250	3250	0	1
3250	3250	0	1
3250	3250	0	1

Rysunek 7: Wyniki przeprowadzonych testów



Rysunek 8: Rozkład wyników oczekiwanych i przewidzianych z ostatniego testu

Po potasowaniu zbioru *mushroom* mogliśmy otrzymać drzewo idealne, któremu udało się przewidzieć poprawnie wszystkie dane w zbiorze testującym. Jedynie w jednym teście pomylił się 4 razy, jednakże nadal jest to zadowalająca skuteczność.

4 Dlaczego na jednym zbiorze jest znacznie lepszy wynik niż na drugim?

Na zbiorze mushroom otrzymaliśmy znacznie lepsze wyniki niż na zbiorze $breast\ cancer$, ponieważ zawiera on więcej informacji, na podstawie których możemy stworzyć dokładniejsze drzewo. Zbiór mushroom posiada 8124 instancji, gdzie każda z nich ma 22 równoważnych atrybutów, zaś zbiór $breast\ cancer$ posiada 286 instancji, gdzie każda z nich ma 9 atrybutów. Dodatkowo nie możemy zapomnieć, że drzewa były tworzone na $\frac{3}{5}$ instancji zbioru, co zmniejszyło ilość informacji, na podstawie których były one tworzone. Ponad to rozpoznawanie czy grzyb jest jadalny, czy trujący jest o wiele prostsze niż przewidywanie występowania raka piersi. Jest to bardziej złożony problem i wymaga więcej informacji na temat konkretnego przypadku, aby nauczyć algorytm rozpoznawania go.