**WYŻSZA SZKOŁA INFORMATYKI**

**I ZARZĄDZANIA**

Z SIEDZIBĄ W RZESZOWIE

Sprawozdanie

Sztuczna inteligencja

**Algorytm generowanie reguł GTS**

Prowadzący: dr inż. Mariusz Wrzesień Wykonawca: Oleh Korsunskyi w58957

Rzeszów 2019

Opis problemu

Celem jest nauczyć się korzystać z algorytmu GTS i za pomocą tego algorytmu stworzyć zestaw reguł decyzyjnych, dla danych umiejscowionych w tablicy decyzji. Reguły składają się z warunku oraz decyzji. Ponad 93% istniejących na świecie sy­stemów ekspertowych zawiera wie­­dzę, strukturyzowaną w postaci reguł składniowych, dlatego jest bardzo ważne rozumienie jak tworzyć takie reguły.

Badane zbiory danych

Dane, które należy przetworzyć, dotyczą problemu doboru soczewek. Są one przedstawione za pomocą tablicy decyzji, która składa się z 22 wierszy(przypadków) i 5 kolumn(4 cechy i 1 decyzja). Pierwszy atrybut **Wiek** jest atrybutem porządkowym i zawiera 3 wartości. Inne atrybuty **Wada, Astygmatyzm, Lzawienie** są atrybutami binarnymi. Ostatnia kolumna jest kolumną decyzji. Składa się z trzech klas: **brak, miękkie, twarde.**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| LP. | Wiek | Wada | Astygmatyzm | Lzawienie | Soczewki |
| 1 | mlody | krotkowidz | nie | normalne | miekkie |
| 2 | mlody | dalekowidz | tak | normalne | twarde |
| 3 | mlody | dalekowidz | nie | zmniejszone | brak |
| 4 | mlody | krotkowidz | tak | zmniejszone | brak |
| 5 | prestarczy | krotkowidz | tak | zmniejszone | brak |
| 6 | prestarczy | krotkowidz | nie | normalne | miekkie |
| 7 | mlody | krotkowidz | tak | normalne | twarde |
| 8 | starczy | dalekowidz | tak | zmniejszone | brak |
| 9 | prestarczy | dalekowidz | nie | zmniejszone | brak |
| 10 | prestarczy | dalekowidz | tak | zmniejszone | brak |
| 11 | prestarczy | krotkowidz | nie | zmniejszone | brak |
| 12 | mlody | dalekowidz | tak | zmniejszone | brak |
| 13 | starczy | dalekowidz | tak | normalne | brak |
| 14 | starczy | krotkowidz | tak | zmniejszone | brak |
| 15 | starczy | krotkowidz | nie | normalne | brak |
| 16 | prestarczy | krotkowidz | tak | normalne | twarde |
| 17 | mlody | krotkowidz | nie | zmniejszone | brak |
| 18 | starczy | krotkowidz | nie | zmniejszone | brak |
| 19 | starczy | dalekowidz | nie | zmniejszone | brak |
| 20 | mlody | dalekowidz | nie | normalne | miekkie |
| 21 | starczy | dalekowidz | nie | normalne | miekkie |
| 22 | prestarczy | dalekowidz | tak | normalne | brak |

Obliczenia

Na przykładzie tablicy decyzji z problemem doboru soczewek można opisać działanie algorytmu GTS. Generujemy pierwszą regułę zgodnie z algorytmem, czyli bierzemy pierwszy, niepokryty regułą przypadek(1 wiersz). Wartość dla pierwszego atrybutu to **Wiek = młody** z odpowiadającą mu w kolumnie decyzji klasą **Soczewki = miękkie**. Dalej szukamy w naszej tablicy decyzji takie same kombinacje. Kolejnym krokiem jest obliczenie wartości parametrów, potrzebnych do stworzenia reguł, opisanych wzorami:

* ogólność: ,
* dokładność ,
* parametr H ,

gdzie

* Ep to liczba przypadków pozytywnych, czyli takich, które posiadają taką samą wartość atrybutu i decyzji w danym zestawieniu,
* Eb to liczba przypadków błędnych, czyli takich, które dla danej wartości atrybutu posiadają inną wartość decyzji,
* E to liczba wszystkich przypadków w tablicy decyzji.

Żeby myśmy mogli stworzyć regułę wartość parametru A powinna być równa 1. A = 1

Wszystkich przypadków (*E*) jest 22, dla pary **Wiek = młody** oraz **Soczewki = miękkie** jest dwa przypadki pozytywne (*Ep*) oraz sześć przypadków błędnych (*Eb*). Podstawiając dane do wzoru dla kolejnych wartości atrybutów w pierwszym przypadku, otrzymano:

**Dla Wiek = młody**

Ponieważ parametr *A* nie jest równy 1, my zgodnie z algorytmem dalej przeszukujemy tablicę decyzji biorąc pod uwagę kolejną cechę w pierwszym przypadku:

**Dla *Wada = krotkowidz***

**Dla *Astygmatyzm = nie***

**Dla *Łzawienie = normalne***

W podanym przykładzie dla żadnych z badanych przypadków nie występuje sytuacja, gdzie kombinacja wartości pojedynczych atrybutów z daną klasą osiąga wartość *A=1, dlatego* wybieramy *tą regułę* która miała najwyższy parametr *H* i dobieramy do niej kolejne kombinacje, aż parametr *A* będzie miał odpowiednią wartość. Prawidłową kombinacją będzie **Wiek = młody, Astygmatyzm = nie, Łzawienie = normalne**. Z tej kombinacji możemy utworzyć regułę 1, która brzmi:

**JEŻELI** Wiek = młody **ORAR** Astygmatyzm = nie **ORAZ** Łzawienie = normalne

**TO** Soczewki = miękkie

Powyższa reguła pokrywa przypadki {1, 20}. Zatem kolejnym rozpatrywanym przypadkiem będzie następny nie pokryty przez żadną regułę przypadek, czyli przypadek 2, który należy badać w ten sam sposób. Jeżeli w którymś z badanych przypadków występuje sytuacja, gdzie kombinacja wartości pojedynczej atrybutu z daną klasą osiąga wartość *A=1* to należy stworzyć regułę z tego jednego atrybutu.

Wyniki

Wykonując wszystkie działania zgodnie z algorytmem GTS dostaliśmy 9 reguł:

1. **JEZELI** Wiek = mlody **ORAZ** Astygmatyzm = nie **ORAZ** Lzawienie = normalne **TO** Soczewki = miekkie
2. **JEZELI** Wiek = mlody **ORAZ** Astygmatyzm = tak **ORAZ** Lzawienie = normalne **TO** Soczewki = twarde
3. **JEZELI**  Lzawienie = zmniejszone **TO** Soczewki = brak
4. **JEZELI** Wiek = prestarczy **ORAZ** Astygmatyzm = nie **ORAZ** Lzawienie = normalne **TO** Soczewki = miękkie
5. **JEZELI** Wiek = starczy **ORAZ** Astygmatyzm = tak **ORAZ** Lzawienie = normalne **TO** Soczewki = brak
6. **JEZELI** Wiek = starczy **ORAZ** Wada = krotkowidz **ORAZ** Lzawienie = normalne **TO** Soczewki = brak
7. **JEZELI** Wiek = prestarczy **ORAZ** Wada = krotkowidz **ORAZ** Astygmatyzm = tak **ORAZ** Lzawienie = normalne **TO** Soczewki = twarde
8. **JEZELI** Wiek = starczy **ORAZ** Wada = dalekowidz **ORAZ** Astygmatyzm = nie **ORAZ** Lzawienie = normalne **TO** Soczewki = miękkie
9. **JEZELI** Wiek = prestarczy **ORAZ** Wada = dalekowidz **ORAZ** Lzawienie = normalne **TO** Soczewki = brak

Wnioski

Algorytm GTS działa na zasadzie sekwencyjnego pokrywania, czyli gdy wygenerujemy za pomocą tego algorytmu pierwszą regułę i znajdziemy dla niej przypadki, to w następnym cyklu pomijamy wszystkie przypadki pozytywne dla tej reguły. Następnie analizujemy pozostałe przypadki zamieszczone w tablicy decyzji. Wynikiem jest zbiór reguł, które pokrywają wszystkie przypadki ze zbioru uczącego. Zgodnie z algorytmem robimy działania, dopóki każdy przypadek nie jest pokryty regułą.

Z powyższej tablicy decyzji powstało dziewięć reguł. Niektóre reguły pokrywają więcej przypadków, niektóre mniej. Ilość przypadków pokrytych jedną regułą zależy od podobieństwa obiektów i od występowania rzadkich kombinacji cech i decyzji, im więcej takich kombinacji tym więcej będziemy potrzebować reguł.

Moim zdaniem, algorytm GTS za dosyć krótki czas generuje reguły dobrej jakości, oprócz tego ten algorytm jest łatwy w zrozumieniu dzięki prostym wzorom matematycznym i co z tego wypływa łatwy w implementacji. Dlatego uważam, że ten algorytm uczenia maszynowego może być wykorzystywany w różnego rodzaju projektach i badaniach.