

Metody indukcji reguł



Indukcja reguł

- Grupa metod charakteryzująca się wydobywaniem reguł ostrych na podstawie analizy przypadków.

Dane doświadczalne składają się z dwóch części:

1) wejściowych \mathbf{X} , gdzie

$\mathbf{X} = [\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_3]^T$, gdzie \mathbf{a}_1, \dots to atrybut (zmienna) w naszym przypadku zwykle zmienna symboliczna

2) wyjściowych \mathbf{y}

Celem indukcji reguł, jest znalezienie takiego zbioru reguł, który realizuje funkcję $\mathbf{y} = f(\mathbf{X})$

Przykład

▣ Zbioru danych

Zawiera – różne przypadki decyzji czy grać/nie grać w golfa w zależności od warunków atmosferycznych

X				Y
Outlook	Temp	Humidity	Windy	Play
sunny	hot	high	FALSE	no
sunny	hot	high	TRUE	no
overcast	hot	high	FALSE	yes
rainy	mild	high	FALSE	yes
rainy	cool	normal	FALSE	yes
rainy	cool	normal	TRUE	no
overcast	cool	normal	TRUE	yes
sunny	mild	high	FALSE	no
sunny	cool	normal	FALSE	yes
rainy	mild	normal	FALSE	yes
sunny	mild	normal	TRUE	yes
overcast	mild	high	TRUE	yes
overcast	hot	normal	FALSE	yes
rainy	mild	high	TRUE	no

Szukamy zbioru reguł, który odwzorowuje informacje o pogodzie na wiedzę o tym, czy grać czy nie grać w golfa

Metody oceny reguł

Po co nam ocena reguł

- Bo mówimy o indukcji reguł (reguły są niepewne)
- Potrzebujemy prawdopodobieństwa mówiącego o jakości reguły i stopniu w jakim możemy jej ufać

Przykłady współczynników oceny reguł

- Wsparcie reguły
- Antywsparcie reguły
- Ufność
- Funkcja atrakcyjności reguły

Metody oceny reguł

- Wsparcie reguły ang. Rule support
 - Wsparcie przesłanki – ilość obiektów w zbiorze spełniająca warunki przesłanki (część reguły: *jeżeli*) – oznaczane jako $\text{sup}(p)$, gdzie p to przesłanka
 - Wsparcie konkluzji – ilość obiektów w zbiorze spełniająca konkluzję reguły (część reguły: *to*), oznaczane jako $\text{sup}(q)$, gdzie q to konkluzja
 - Wsparcie reguły $\text{sup}(p \rightarrow q)$ ilość obiektów w zbiorze spełniająca zarówno przesłankę jak i konkluzję.
 - Czasem wsparcie reguły wrażane jest względnie jako stosunek wsparcia reguły (liczby obiektów) w stosunku do całkowitej liczebności zbioru obiektów

Przykład

▣ Zbioru danych

Outlook	Temp	Humidity	Windy	Play
sunny	hot	high	FALSE	no
sunny	hot	high	TRUE	no
overcast	hot	high	FALSE	yes
rainy	mild	high	FALSE	yes
rainy	cool	normal	FALSE	yes
rainy	cool	normal	TRUE	no
overcast	cool	normal	TRUE	yes
sunny	mild	high	FALSE	no
sunny	cool	normal	FALSE	yes
rainy	mild	normal	FALSE	yes
sunny	mild	normal	TRUE	yes
overcast	mild	high	TRUE	yes
overcast	hot	normal	FALSE	yes
rainy	mild	high	TRUE	no

Metody oceny reguł

- Wsparcie reguły ang. Rule support

Mówi jak pewna jest decyzja reguły, jeśli przesłanka reguły zostanie spełniona. Czyli ile przypadków w zbiorze danych spełniło przesłanki reguły i reguła dała poprawną odpowiedź, lub jakie jest prawdopodobieństwo, w skali całego zbioru danych, że dana reguła jest poprawna.

Metody oceny reguł

- Anty wsparcie reguły ang. Rule anti-support
 - Ilość obiektów w zbiorze spełniających warunki przesłanki reguły ale nie spełniających konkluzji reguły. Jeżeli przyjąć że przesłankę reguły spełnia k przypadków, a spośród nich konkluzję spełnia h przypadków to antywsparcie wynosi $k-h$
 - Czasem antywsparcie reguły wrażane jest względnie jako stosunek antywsparcia reguły (liczby obiektów) w stosunku do całkowitej liczebności zbioru obiektów

Przykład

▣ Zbioru danych

Outlook	Temp	Humidity	Windy	Play
sunny	hot	high	FALSE	no
sunny	hot	high	TRUE	no
overcast	hot	high	FALSE	yes
rainy	mild	high	FALSE	yes
rainy	cool	normal	FALSE	yes
rainy	cool	normal	TRUE	no
overcast	cool	normal	TRUE	yes
sunny	mild	high	FALSE	no
sunny	cool	normal	FALSE	yes
rainy	mild	normal	FALSE	yes
sunny	mild	normal	TRUE	yes
overcast	mild	high	TRUE	yes
overcast	hot	normal	FALSE	yes
rainy	mild	high	TRUE	no

Metody oceny reguł

- Anty wsparcie reguły ang. Rule anti-support

Mówi nam na ile w skali całego zbioru dana reguła może być błędna, czyli wyraża informacje o tym, że jeśli zostanie spełniona przesłanka reguły, to ile jest przypadków z błędną odpowiedzią. Innymi słowy – jakie jest prawdopodobieństwo w skali całego zbioru popełnienia błędu przez tą regułę

Metody oceny reguł

□ Ufność (ang. Confidence)

- Dla reguły $(p \rightarrow q)$ ufność $\text{conf}(p \rightarrow q)$ określana jest jako:

$$\text{conf}(p \rightarrow q) = \text{sup}(p \rightarrow q) / \text{sup}(p)$$

- Czułość to ilość przypadków spełniająca regułę w stosunku do całkowitej liczby przypadków spełniających przesłankę

Przykład

▣ Zbioru danych

Outlook	Temp	Humidity	Windy	Play
sunny	hot	high	FALSE	no
sunny	hot	high	TRUE	no
overcast	hot	high	FALSE	yes
rainy	mild	high	FALSE	yes
rainy	cool	normal	FALSE	yes
rainy	cool	normal	TRUE	no
overcast	cool	normal	TRUE	yes
sunny	mild	high	FALSE	no
sunny	cool	normal	FALSE	yes
rainy	mild	normal	FALSE	yes
sunny	mild	normal	TRUE	yes
overcast	mild	high	TRUE	yes
overcast	hot	normal	FALSE	yes
rainy	mild	high	TRUE	no

Metody oceny reguł

▣ Ufność (ang. Confidence)

Mówi jakie jest prawdopodobieństwo poprawności danej reguły, tzn jest to ocena tylko pojedynczej reguły w oderwaniu od oceny jej na tle całego zbioru.

Czyli ocenia prawdopodobieństwo, mówiące o tym, że jeśli dana reguła zostanie wybrana to jak ona jest pewna

Metody oceny reguł

- Funkcja atrakcyjności reguły (Rule interest function)
 - Mierzy korelację pomiędzy przesłanką a konkluzją reguły

$$RI(p \rightarrow q) = \text{sup}(p \rightarrow q) - (\text{sup}(p)\text{sup}(q)/c)$$

Gdzie:

c – całkowita ilość przypadków

Przykład

▣ Zbioru danych

Outlook	Temp	Humidity	Windy	Play
sunny	hot	high	FALSE	no
sunny	hot	high	TRUE	no
overcast	hot	high	FALSE	yes
rainy	mild	high	FALSE	yes
rainy	cool	normal	FALSE	yes
rainy	cool	normal	TRUE	no
overcast	cool	normal	TRUE	yes
sunny	mild	high	FALSE	no
sunny	cool	normal	FALSE	yes
rainy	mild	normal	FALSE	yes
sunny	mild	normal	TRUE	yes
overcast	mild	high	TRUE	yes
overcast	hot	normal	FALSE	yes
rainy	mild	high	TRUE	no

Metody oceny reguł

- Funkcja atrakcyjności reguły (Rule interest function)

Innymi słowy jest to swego rodzaju kombinacja ufności i wsparcia reguły, czyli miara ta łączy ocenę jakości pojedynczej reguły z oceną tej reguły na tle całego zbioru.

Metody indukcji reguł

Podział metod indukcji reguł

- Bezpośrednia indukcja reguł
 - indukcja reguł na podstawie przypadków

- Indukcja reguł z nauczonych modeli np. sieci neuronowych, modeli statystycznych itp.
 - Transformacja wiedzy zawartej w różnych modelach nie regułowych (w których zgromadzona wiedza dla człowieka nieczytelna)

Metody bezpośredniej indukcji reguł

- Najczęściej stosowana strategia
 - metody przeszukiwania połączone z sekwencyjnym pokrywaniem przestrzeni wejściowej.
 - Tworzenie reguł - Od najbardziej ogólnej do najbardziej szczegółowej

Gdzie:

- Najbardziej ogólna reguła - taka która pokrywa maksymalną liczbę przypadków przestrzeni wejściowej X , dla których odpowiedzią systemu jest ta sama wartość Y ,

oraz

jest to reguła o prostszej budowie swojej części warunkowej (mniej warunków)

- Reguła szczegółowa - pokrywająca małą liczbę wektorów przestrzeni X

oraz

ma bardziej złożoną część warunkową

Terminologia

- Atrybut – przykładowa zmienna opisująca dane zjawisko
- Typy atrybutów
 - Ciągły – jeżeli elementy przyjmują wartości ze zbioru liczb rzeczywistych
 - Symboliczny/nominalny – jeżeli elementy atrybutu przyjmują wartości ze skończonej liczby symboli np. zbiór A np.
atrybut: pogoda {słoneczna,pochmurna,deszczowa}
 - Binarny – jeżeli elementy atrybutu przyjmują wartości {0,1}

Terminologia – cd.

- Selektor – podzbiór wartości symbolicznych. W regułach – zbiór wartości symbolicznych dla danego atrybutu użytych następnie w części warunkowej reguły
- 4 typy selektorów (V_A) – zbiór wartości selektora
 - $V_A = \{v\}$ – selektor pojedynczy
 - $V_A = \{v_1, v_2, \dots, v_m\}$ – selektor dysjunkcyjny
 - $V_A = \{?\}$ – selektor uniwersalny
 - $V_A = \emptyset$ – selektor pusty

Terminologia – cd.

□ Kompleks – zbiór selektorów

□ Przykład selektora:

{hot,cool} dla zmiennej Temp

□ Przykład kompleksu:

1. $\langle \text{sunny}, \{\text{hot}, \text{cool}\}, ?, ? \rangle$

2. $\langle ?, \text{mild}, ?, \text{true} \rangle$

Outlook	Temp	Humidity	Windy	Play
sunny	hot	high	FALSE	no
sunny	hot	high	TRUE	no
overcast	hot	high	FALSE	yes
rainy	mild	high	FALSE	yes
rainy	cool	normal	FALSE	yes
rainy	cool	normal	TRUE	no
overcast	cool	normal	TRUE	yes
sunny	mild	high	FALSE	no
sunny	cool	normal	FALSE	yes
rainy	mild	normal	FALSE	yes
sunny	mild	normal	TRUE	yes
overcast	mild	high	TRUE	yes
overcast	hot	normal	FALSE	yes
rainy	mild	high	TRUE	no

Zadanie: podej numery wektorów spełniających powyższe dwa kompleksy

Algorytm sekwencyjnego pokrywania

□ Algorytm sekwencyjnego pokrywania

function sekwencyjnePokrywanie(**T**)

T zbiór danych uczących **T = [X y]**

R = \emptyset ; **P** = **T**

while **P** $\neq \emptyset$

$k = \text{znajdźKompleks}(\mathbf{T}, \mathbf{P});$

$w = \text{kategoria}(k, \mathbf{T}, \mathbf{P});$

$\mathbf{R} = \mathbf{R} \cup \{k \rightarrow w\};$

$\mathbf{P} = \mathbf{P} - \mathbf{P}_k$ / \mathbf{P}_k – zbiór wektorów pokrytych przez kompleks k

end

return **R**

Przykład

□ Zbiór danych

Outlook	Temp	Humidity	Windy	Play
sunny	hot	high	FALSE	no
sunny	hot	high	TRUE	no
overcast	hot	high	FALSE	yes
rainy	mild	high	FALSE	yes
rainy	cool	normal	FALSE	yes
rainy	cool	normal	TRUE	no
overcast	cool	normal	TRUE	yes
sunny	mild	high	FALSE	no
sunny	cool	normal	FALSE	yes
rainy	mild	normal	FALSE	yes
sunny	mild	normal	TRUE	yes
overcast	mild	high	TRUE	yes
overcast	hot	normal	FALSE	yes
rainy	mild	high	TRUE	no

Przykład

□ Przykładowe kompleksy i ich selektory

Outlook	Temp	Humidity	Windy	Play
sunny	hot	high	FALSE	no
sunny	hot	high	TRUE	no
overcast	hot	high	FALSE	yes
rainy	mild	high	FALSE	yes
rainy	cool	normal	FALSE	yes
rainy	cool	normal	TRUE	no
overcast	cool	normal	TRUE	yes
sunny	mild	high	FALSE	no
sunny	cool	normal	FALSE	yes
rainy	mild	normal	FALSE	yes
sunny	mild	normal	TRUE	yes
overcast	mild	high	TRUE	yes
overcast	hot	normal	FALSE	yes
rainy	mild	high	TRUE	no

Szukamy selektorów i z nich kompleksów:

- 1) $\langle \text{overcast}, ?, ?, ? \rangle \Rightarrow \text{yes}$ (wsp.4)
- 2) $\langle ?, ?, \text{normal}, \text{False} \rangle \Rightarrow \text{yes}$ (wsp.4)
- 3) $\langle \text{Sunny}, ?, \text{normal}, ? \rangle \Rightarrow \text{yes}$ (wsp.2)
- 4) $\langle \text{Sunny}, ?, \text{high}, ? \rangle \Rightarrow \text{no}$ (wsp.3)

Przykład

- Wybieramy kompleks $\langle \text{overcast}, ?, ?, ? \rangle$
 \Rightarrow yes – bo najbardziej ogólna i usuwamy ze zbioru wektory dotychczas pokryte przez obecną regułę

Outlook	Temp	Humidity	Windy	Play
sunny	hot	high	FALSE	no
sunny	hot	high	TRUE	no
overcast	hot	high	FALSE	yes
rainy	mild	high	FALSE	yes
rainy	cool	normal	FALSE	yes
rainy	cool	normal	TRUE	no
overcast	cool	normal	TRUE	yes
sunny	mild	high	FALSE	no
sunny	cool	normal	FALSE	yes
rainy	mild	normal	FALSE	yes
sunny	mild	normal	TRUE	yes
overcast	mild	high	TRUE	yes
overcast	hot	normal	FALSE	yes
rainy	mild	high	TRUE	no



Outlook	Temp	Humidity	Windy	Play
sunny	hot	high	FALSE	no
sunny	hot	high	TRUE	no
rainy	mild	high	FALSE	yes
rainy	cool	normal	FALSE	yes
rainy	cool	normal	TRUE	no
sunny	mild	high	FALSE	no
sunny	cool	normal	FALSE	yes
rainy	mild	normal	FALSE	yes
sunny	mild	normal	TRUE	yes
rainy	mild	high	TRUE	no

Przykład

- Szukamy kolejnych kompleksów z obecnego podzbioru dotychczas nie pokrytego:

Outlook	Temp	Humidity	Windy	Play
sunny	hot	high	FALSE	no
sunny	hot	high	TRUE	no
rainy	mild	high	FALSE	yes
rainy	cool	normal	FALSE	yes
rainy	cool	normal	TRUE	no
sunny	mild	high	FALSE	no
sunny	cool	normal	FALSE	yes
rainy	mild	normal	FALSE	yes
sunny	mild	normal	TRUE	yes
rainy	mild	high	TRUE	no

Szukamy selektorów i z nich kompleksów:

- 1) $\langle ?, ?, normal, False \rangle \Rightarrow yes$ (wsp.3)
- 3) $\langle Sunny, ?, normal, ? \rangle \Rightarrow yes$ (wsp.2)
- 4) $\langle Sunny, ?, high, ? \rangle \Rightarrow no$ (wsp.3)

Przykład

Wybieramy kolejny kompleks
<?, ?, normal, False> => yes

| Outlook | Temp | Humidity | Windy | Play |
|---------|------|----------|-------|------|
| sunny | hot | high | FALSE | no |
| sunny | hot | high | TRUE | no |
| rainy | mild | high | FALSE | yes |
| rainy | cool | normal | FALSE | yes |
| rainy | cool | normal | TRUE | no |
| sunny | mild | high | FALSE | no |
| sunny | cool | normal | FALSE | yes |
| rainy | mild | normal | FALSE | yes |
| sunny | mild | normal | TRUE | yes |
| rainy | mild | high | TRUE | no |



| Outlook | Temp | Humidity | Windy | Play |
|---------|------|----------|-------|------|
| sunny | hot | high | FALSE | no |
| sunny | hot | high | TRUE | no |
| rainy | mild | high | FALSE | yes |
| rainy | cool | normal | TRUE | no |
| sunny | mild | high | FALSE | no |
| sunny | mild | normal | TRUE | yes |
| rainy | mild | high | TRUE | no |

Przykład

- Szukamy kolejnych kompleksów z obecnego podzbioru dotychczas nie pokrytego:

| Outlook | Temp | Humidity | Windy | Play |
|---------|------|----------|-------|------|
| sunny | hot | high | FALSE | no |
| sunny | hot | high | TRUE | no |
| rainy | mild | high | FALSE | yes |
| rainy | cool | normal | TRUE | no |
| sunny | mild | high | FALSE | no |
| sunny | mild | normal | TRUE | yes |
| rainy | mild | high | TRUE | no |

Szukamy selektorów i z nich kompleksów:

1) $\langle \text{Sunny}, ?, \text{high}, ? \rangle \Rightarrow \text{no}$ (wsp.3)

Przykład

- Wybieramy kolejny kompleks
<Sunny,?,high,?> => no

Outlook	Temp	Humidity	Windy	Play
sunny	hot	high	FALSE	no
sunny	hot	high	TRUE	no
rainy	mild	high	FALSE	yes
rainy	cool	normal	TRUE	no
sunny	mild	high	FALSE	no
sunny	mild	normal	TRUE	yes
rainy	mild	high	TRUE	no



Outlook	Temp	Humidity	Windy	Play
rainy	mild	high	FALSE	yes
rainy	cool	normal	TRUE	no
sunny	mild	normal	TRUE	yes
rainy	mild	high	TRUE	no

Przykład

- Szukamy kolejnych kompleksów z obecnego podzbioru dotychczas nie pokrytego:

Outlook	Temp	Humidity	Windy	Play
rainy	mild	high	FALSE	yes
rainy	cool	normal	TRUE	no
sunny	mild	normal	TRUE	yes
rainy	mild	high	TRUE	no

Szukamy selektorów i z nich kompleksów:

1) $\langle \text{rain}, ?, ?, \text{true} \rangle \Rightarrow \text{no}$

(wsp.3)

Przykład

- Wybieramy kolejny kompleks:
 $\langle \text{rain}, ?, ?, \text{true} \rangle \Rightarrow \text{no}$

Outlook	Temp	Humidity	Windy	Play
rainy	mild	high	FALSE	yes
rainy	cool	normal	TRUE	no
sunny	mild	normal	TRUE	yes
rainy	mild	high	TRUE	no



Outlook	Temp	Humidity	Windy	Play
rainy	mild	high	FALSE	yes
sunny	mild	normal	TRUE	yes

Przykład

- Pozostałą resztę przypisujemy do klasy $\text{play} \Rightarrow \text{yes}$

Outlook	Temp	Humidity	Windy	Play
rainy	mild	high	FALSE	yes
sunny	mild	normal	TRUE	yes

Przykład

□ Zbiór danych

Outlook	Temp	Humidity	Windy	Play
sunny	hot	high	FALSE	no
sunny	hot	high	TRUE	no
overcast	hot	high	FALSE	yes
rainy	mild	high	FALSE	yes
rainy	cool	normal	FALSE	yes
rainy	cool	normal	TRUE	no
overcast	cool	normal	TRUE	yes
sunny	mild	high	FALSE	no
sunny	cool	normal	FALSE	yes
rainy	mild	normal	FALSE	yes
sunny	mild	normal	TRUE	yes
overcast	mild	high	TRUE	yes
overcast	hot	normal	FALSE	yes
rainy	mild	high	TRUE	no

Szukamy selektorów i z nich Kompleksów i odpowiadającym im etykiet klas:

1. $\langle \text{overcast}, ?, ?, ? \rangle \Rightarrow \text{yes}$
2. $\langle ?, ?, \text{normal}, \text{False} \rangle \Rightarrow \text{yes}$
3. $\langle \text{Sunny}, ?, \text{high}, ? \rangle \Rightarrow \text{no}$
4. $\langle \text{rain}, ?, ?, \text{true} \rangle \Rightarrow \text{no}$
5. $\langle ?, ?, ?, ? \rangle \Rightarrow \text{yes}$

Ostateczny zbiór reguł

- *If Outlook=Overcast then Play=true*
- *If Humidity=Normal & Windy=False
then Play = true*
- *If Outlook=Sunny & Humidity=high
then Play = no*
- *If Outlook=rain & Windy=true
then Play= no*
- *Else Play = yes*

Problem!

- Jak znaleźć selektory i kompleksy w jak najkrótszym czasie (przy minimalnej złożoności obliczeniowej)
- Rozwiązania:
 - Algorytm AQ
 - Algorytm CN2

Algorytm - AQ

- Algorytm zaproponowany przez Ryszarda Michalskiego
- W rzeczywistości rodzina różnych metod bazujących na tym samym algorytmie podstawowym,
 - modyfikacje poprzez dodanie różnych rozszerzeń lub funkcjonalności.
- Wersja podstawowa algorytmu AQ bazuje na wykorzystaniu metod przeszukiwania w sekwencyjnym pokrywaniu przestrzeni wejściowej regułami.
- Specyficzne słownictwo:
 - gwiazda - zbiór kompleksów (selektorów), spośród których wybierany jest poprzednik reguły.
- Podstawowa wersja algorytmu – reguła może pokrywać tylko przypadki pozytywne!!!, bardziej ogólna metoda – pokrywa więcej przypadków pozytywnych.
Gdzie:
 - Przypadki pozytywne - zbiór wektorów należących do klasy C^+ dla których reguła jest tworzona, zaś przypadki negatywne C^- to zbiór wektorów nie należących do klasy C^+ , czyli $C^- = C - C^+$
- Podstawowa wersja algorytmu była zdefiniowana dla atrybutów dyskretnych przy założeniu separacji klas. (Klasy separowalne tzn nie nakładają się na siebie)

Algorytm - AQ

Function znajdźKompleks(**T**,**P**)

arg. Wejściowe

T – zbiór trenujący

P – zbiór zawierający przykłady nie pokryte przez dotychczasowy zbiór reguł

\mathbf{x}_p = ziarnoPozytywne(**P**);

S = {⟨?⟩} //gwiazda

While (($\exists \mathbf{x} \in T$), takie że **S** pokrywa \mathbf{x} i $c(\mathbf{x}) \neq c(\mathbf{x}_p)$) //dopóki istnieją przypadki negatywnego pokryte przez **S**

\mathbf{x}_n = ziarnoNegatywne(**T**,**S**, \mathbf{x}_p);

S' = częściowaGwiazda(\mathbf{x}_p , \mathbf{x}_n);

if **S'** = \emptyset

return ⟨ \emptyset ⟩;

endif

S = **S** \cap **S'** //uwzględnienie w **S** częściowej gwiazdy **S'**

S = **S** - { $k \in \mathbf{S} \mid$ (istnieje $k' \in \mathbf{S}$) $k \leq k'$ } //Usunięcie nadmiarowych kompleksów

S = $\operatorname{argmax}_{k \in \mathbf{S}, m} V(\mathbf{x}_p, \mathbf{T}, \mathbf{P})$ //Pozostawienie tylko m najlepszych kompleksów wg. Kryterium $V()$

Endwhile

Return $\operatorname{argmax}_{k \in \mathbf{S}} V(\mathbf{x}_p, \mathbf{T}, \mathbf{P})$

Algorytm - AQ

Function częściowaGwiazda($\mathbf{x}_p, \mathbf{x}_n$);

arg. Wejściowe

\mathbf{x}_p – ziarno pozytywne

\mathbf{x}_n – ziarno negatywne

$\mathbf{S}' = \{\emptyset\}$ //zbiór maksymalnie ogólnych kompleksów pokrywających \mathbf{x}_p i nie pokrywających \mathbf{x}_n

Foreach $a_i \in \mathbf{P}$ //dla wszystkich atrybutów

$k = \langle ? \rangle$;

$V = A_i - \{a_i(\mathbf{x}_n)\}$ //ze zbioru wartości atrybutu a_i usuń wartości ziarna negatywnego

if $a_i(\mathbf{x}_p) \in V$

 umieść selektor w kompleksie k na pozycji i

$\mathbf{S}' = \mathbf{S} \cup k$

endif

Endforeach

Return \mathbf{S}'

Algorytm – CN2

- Autorzy Clark i Niblett
- Modyfikacja i rozszerzenie algorytmu AQ
 - Stawiany przez autorów problem - sytuację ekstrakcji reguł z danych przy założeniu istnienia szumu.
- Rozwiązanie1: indukcji reguł na podstawie miary entropii:

$$Q_E(q) = - \sum_{i=1}^c p(C_i|q) \log_2 p(C_i|q)$$

gdzie

$p(C_i|q)$ jest względną liczbą wektorów z klasy C_i pokrytych przez dany kompleks q .

- Efekt - możliwość pokrywania poprzez regułę niewielkiej ilości przypadków negatywnych, dzięki czemu poprawa generalizacji algorytmu.
- Rozwiązanie 2: wbudowany algorytm oczyszczania reguł, podobny do stosowanego w systemach drzew decyzji, efekt usunięcie nieużytecznych reguł, które są wynikiem przetrenowania systemu.
- Uwaga reguły z CN2 mogą być niespójne (mogą się nakładać – te same przesłanki mogą aktywować konkluzje przeciwstawnych reguł).
- Zasada czytania reguł: od najbardziej szczegółowej do najogólniejszej (czytanie reguł wg. zasad czytania stosu! Ostatnio dodana reguła sprawdzana jest jako pierwsza)
- Rozwiązanie 3: dodanie do systemu reguły typu *jeśli nie* (ang. else), stosowanej w przypadku nie spełnienia żadnej z reguł. \\

Algorytm – CN2

Function znajdźKompleks(**T**,**P**)

arg. Wejściowe

T – zbiór trenujący

P – zbiór zawierający przykłady nie pokryte przez dotychczasowy zbiór reguł

\mathbf{x}_p = ziarnoPozytywne(**P**);

S = {⟨?⟩}; //gwiazda

k_* = ⟨?⟩;

While (**S** ≠ ∅)//dopóki **S** nie jest pusty

S' = **S** ∩ **S**

S' = **S'** - **S** - {⟨ ∅ ⟩}

Foreach $k \in \mathbf{S}$

if $\psi_k(\mathbf{P}) > \theta$ and $v_k(\mathbf{P}) > v_{k^*}(\mathbf{P})$

$k_* = k$

endif

return ⟨ ∅ ⟩;

endforeach

S = $\text{argmax}_{k \in \mathbf{S}', m} V_k(\mathbf{x}_p, \mathbf{T}, \mathbf{P})$ //Pozostawienie tylko m najlepszych kompleksów wg. kryterium $V()$

Endwhile

Return k_*

Gdzie $v_k(\mathbf{P}) = -E_k(\mathbf{P})$

S = {⟨deszczowa,?,?,?⟩
⟨deszczowa V słoneczna,?,?,?⟩
⟨deszczowa V pochmurna,?,?,?⟩
⟨pochmurna,?,?,?⟩
⟨pochmurna V słoneczna,?,?,?⟩
⟨słoneczna,?,?,?⟩
⟨?,ciepła,?,?,?⟩
⟨?,zimna,?,?,?⟩
⟨?,ciepła V zimna,?,?,?⟩
⟨?,ciepła V umiarkowana,?,?,?⟩
⟨?,umiarkowana,?,?,?⟩
⟨?,umiarkowana V zimna,?,?,?⟩
⟨?,?,duża,?⟩
⟨?,?,normalna,?⟩
⟨?,?,?,silny⟩
⟨?,?,?,słaby⟩}

Przykładowe algorytmy – MLP2LN

- Autorzy: Duch, Adamczak
- Dedykowany algorytm poszukiwania reguł bazujący na sieciach neuronowej MLP.
- Specjalna struktura sieci, w której na wejścia warstwy ukrytej podawane są odpowiednie interwały sygnałów wejściowych sieci, wstępnie przetworzone jako różnica dwóch funkcji sigmoidalnych.
- Wykorzystanie faktu możliwości zmiany nachylenia skosu funkcji sigmoidalnej,
- Efekt: wykorzystanie algorytmów gradientowych oraz propagacji wstecznej do optymalizacji wag sieci.
- Przy małym skosie funkcji sigmoidalnej interpretacja jako sieć MLP i normalne uczenie gradientowe, przy skosie dążącym do ∞ interpretację binarna – logiczna.
- Algorytm uczenia: wraz z kolejnymi epokami uczenia sieci zmianie ulega również współczynnik skosu s . specjalnej funkcji kosztu:

$$E(\mathbf{w}) = \frac{1}{2} \sum_p \sum_i (y(\mathbf{x}_i, \mathbf{W}) - d_i^p)^2 + \frac{\lambda_1}{2} \sum_{i>j} w_{ij}^2 + \frac{\lambda_2}{2} \sum_{i>j} w_{ij}^2 (w_{ij} + 1)^2 (w_{ij} - 1)^2$$

Gdzie:

wyjście sieci wyznaczone dla wektora \mathbf{x}_i oraz wag \mathbf{w} ,
 d - etykieta wektora \mathbf{x}_i ,
 $w_{i,j}$ to i,j -ty element macierz wag,
 λ_1, λ_2 - współczynniki definiowane przez użytkownika.

- Opis funkcji kosztu: 1czynnik – zwykły błąd średniokwadratowy, 2 czynnik wpływa na stopień skomplikowania modelu - wymuszający małe wartości wag \mathbf{w} (selekcję cech na wejściach neuronów) oraz trzeci wymusza duże wartości wag $[-1,0,1]$.
- Wagi poszczególnych czynników funkcji kosztu są zmieniane w procesie uczenia, tak by w końcowym etapie dominującą wartość miał ostatni czynnik odpowiedzialny za binaryzację wejścia sieci.

To tyle jeśli chodzi o



algorytmy indukcji reguł