

1 LEM2 - Idea algorytmu

Dany mamy system decyzyjny (U, A, d) , gdzie $U = o_1, o_2, \dots, o_7$, $A = a_1, a_2, \dots, a_5$, d – atrybut decyzyjny

	a_1	a_2	a_3	a_4	a_5	d
o_1	2	6	1	2	3	1
o_2	1	1	1	3	2	1
o_3	2	1	1	2	3	1
o_4	4	1	3	1	2	1
o_5	3	5	2	1	3	2
o_6	3	1	3	1	1	2
o_7	1	1	1	3	1	2

Algorytm polega na tworzeniu pierwszej reguły przez sekwencyjny wybór najlepszego elementarnego warunku, przy zachowaniu ustalonych kryteriów. Przykłady treningowe pokryte przez regułę są usuwane z rozważań. Proces tworzenia reguł jest powtarzany iteracyjnie do momentu, gdy pozostają jakieś niepokryte obiekty w systemie treningowym. Wszelkie konflikty rozwiązywane są hierarchicznie (wybierana jest wartość pierwsza od góry z lewej strony).

W praktyce wygląda to tak:

Patrzymy na koncept 1 (koncept jest synonimem klasy decyzyjnej), szukając deskryptora, który występuje najczesciej:

W wybranym konceptie najczesciej występuje deskryptor

$(a_2 = 1) \rightarrow$ powstaje z obiektów o_2, o_3, o_4

Nie tworzy jednak reguły ponieważ w konceptie 2 mamy sprzeczność. Skupiając uwagę na obiektach do których pasuje $(a_2 = 1)$ czyli o_2, o_3, o_4 , szukam kolejnego najlepszego deskryptora, z największym pokryciem w klasie 1. Tym deskryptorem jest $(a_3 = 1) \rightarrow$ powstaje z obiektów (o_2, o_3) , dokładam go do pierwszego deskryptora i tworzę koniunkcję:

$(a_2 = 1) \wedge (a_3 = 1)$, jednak powstała koniunkcja dalej jest sprzeczna,

Z faktu, że powyższa reguła powstała z obiektów o_2, o_3 , szukam w nich kolejnego najbardziej linczego deskryptora, tym razem jest nim $(a_1 = 1) \rightarrow$ powstaje z obiektu o_2 , dokładamy znaleziony deskryptor do budowanej reguły:

$(a_2 = 1) \wedge (a_3 = 1) \wedge (a_1 = 1)$, sprzeczność nie została usunięta, stąd wybieramy kolejny deskryptor z obiektu o_2 , dostajemy $(a_4 = 3)$, dołączymy do naszej reguły:

$(a_2 = 1) \wedge (a_3 = 1) \wedge (a_1 = 1) \wedge (a_4 = 3)$, koniunkcja jest wciąż sprzeczna, dodajemy do niej kolejny deskryptor postaci $(a_5 = 2)$, dostajemy:

$(a_2 = 1) \wedge (a_3 = 1) \wedge (a_1 = 1) \wedge (a_4 = 3) \wedge (a_5 = 2)$, ta kombinacja jest niesprzeczna, tworzymy z niej regułę postaci:

$$(a_2 = 1) \wedge (a_3 = 1) \wedge (a_1 = 1) \wedge (a_4 = 3) \wedge (a_5 = 2) \Rightarrow (d = 1)$$

W koncepcie 1 powstała już reguła z obiektu o_2 , stąd przy szukaniu kolejnej skupiamy uwagę na o_1, o_3, o_4 , najczęstszym deskryptorem jest $(a_1 = 2)$, który pasuje do obiektów o_1, o_3 , ten deskryptor nie jest sprzeczny, stąd powstaje reguła:

$(a_1 = 2) \Rightarrow (d = 1)[2]$, reguła ma support 2, ponieważ pasuje do dwóch obiektów, o_1, o_3 .

W koncepcie 1, został nam do rozważenia tylko obiekt o_4 , z którego powstaje nie sprzeczna reguła:

$$(a_1 = 4) \Rightarrow (d = 1)$$

Następnie tworzymy reguły z konceptu 2.

Czyli rozważamy obiekty o_5, o_6, o_7 , najbardziej licznym deskryptorem i pierwszym z brzegu jest $(a_1 = 3)$, do tego jest niesprzeczny, stąd tworzymy regułę:

$$(a_1 = 3) \Rightarrow (d = 2)[2], \text{ pokrywa obiekty } o_5, o_6.$$

Ostatecznie tworzymy regułę z obiektu o_7 : widzimy, że deskryptory:

$(a_1 = 1), (a_2 = 1), (a_3 = 1), (a_4 = 3)$ tworzą sprzeczność, dopiero dołożenie deskryptora $(a_5 = 1)$. likwiduje sprzeczność i powstaje reguła:

$$(a_1 = 1) \wedge (a_2 = 1) \wedge (a_3 = 1) \wedge (a_4 = 3) \wedge (a_5 = 1) \Rightarrow (d = 2)$$

W przypadku gdy sprzeczność występuje na wszystkich $card\{A\}$ deskryptorach warunkowych, tworzymy regułę, która ma alternatywną decyzję. Takie obiekty należą do brzegu systemu decyzyjnego.

Nasze reguły LEM2 ostatecznie mają postać:

rule1 $(a_2 = 1) \wedge (a_3 = 1) \wedge (a_1 = 1) \wedge (a_4 = 3) \wedge (a_5 = 2) \Rightarrow (d = 1)$

rule2 $(a_1 = 2) \Rightarrow (d = 1)[2]$

rule3 $(a_1 = 4) \Rightarrow (d = 1)$

rule4 $(a_1 = 3) \Rightarrow (d = 2)[2]$

rule5 $(a_1 = 1) \wedge (a_2 = 1) \wedge (a_3 = 1) \wedge (a_4 = 3) \wedge (a_5 = 1) \Rightarrow (d = 2)$