

1 Teoria do ćwiczeń z przykładami

System Informacyjny: (U, A)

U - zbiór obiektów;

A - zbiór atrybutów warunkowych;

Przykład: (U, A) , $U = \{ob_1, ob_2, ob_3\}$, $A = \{a_1, a_2, a_3\}$

	a_1	a_2	a_3
ob_1	1	2	3
ob_2	3	2	5
ob_3	10	2	17

System decyzyjny: (U, A, d)

U - zbiór obiektów;

A - zbiór atrybutów warunkowych;

d - atrybut decyzyjny

$d \notin A$

Przykład: System decyzyjny zapisujemy jako (U, A, d) , przyjmijmy,

$U = \{ob_1, ob_2, ob_3\}$

$A = \{a_1, a_2, a_3\}$

$d \in D = 1, 2$

Przykładowy system decyzyjny zgodny z opisem powyżej, może wyglądać następująco,

	a_1	a_2	a_3	d
ob_1	7	2	3	1
ob_2	3	3	5	2
ob_3	10	45	4	1

Zdefiniujmy reguły decyzyjne wzajemnie niesprzeczne,

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

$(a_1 = 2) \wedge (a_2 = 7) \Rightarrow (d = 1)$

$(pogoda = soneczna) \wedge (zona = wpracy) \wedge (czas = wolny) \Rightarrow (decyzja = park)$

$(pogoda = soneczna) \wedge (zona = wdomu) \wedge (czas = wolny) \Rightarrow (decyzja = dom)$

Reguły decyzyjne wzajemnie sprzeczne

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

$(a_1 = 1) \Rightarrow (d = 2)$

$(pogoda = soneczna) \wedge (zona = wpracy) \wedge (czas = wolny) \Rightarrow (decyzja = park)$

$(pogoda = soneczna) \wedge (zona = wpracy) \wedge (czas = wolny) \Rightarrow (decyzja = dom)$

Reguła z alternatywna decyzja

$(pogoda = soneczna) \wedge (zona = wpracy) \wedge (czas = wolny) \Rightarrow (decyzja = park) \vee (decyzja = dom)$

2 Algorytm z rodziny sekwencyjnie pokrywających (sequential covering)

Idea algorytmu pokrywającego obiekty

Szukamy w obiektach systemu decyzyjnego, poczynając od pierwszego, a skończywszy na ostatnim regułą długości jeden, które są niesprzeczne. Po znalezieniu reguły niesprzecznej, dany obiekt wyrzucamy z rozważań, pamiętając o tym, że dalej bierze udział w sprawdzaniu sprzeczności i może wspierać inne reguły. Jeżeli po przeszukaniu wszystkich obiektów, pozostają obiekty nie wyrzucone z rozważań, szukamy w nich kombinacji niesprzecznej długości dwa i postępujemy analogicznie jak w przypadku reguły pierwszego rzędu. Wyszukiwanie reguły niesprzecznych jest kontynuowane do momentu wyeliminowania wszystkich obiektów niesprzecznych. Jeżeli w systemie pojawia się obiekty, które są sprzeczne na wszystkich deskryptorach, nie kreujemy z nich reguły.

Przykładowe wyliczanie reguły pokrywających obiekty:

Dany mamy system decyzyjny (U, A, d) , gdzie $U = o_1, o_2, \dots, o_7, o_8$,
 $A = a_1, a_2, \dots, a_6, d$ – atrybut decyzyjny

	a_1	a_2	a_3	a_4	a_5	a_6	d
o_1	1	1	1	1	3	1	1
o_2	1	1	1	1	3	2	1
o_3	1	1	1	3	2	1	0
o_4	1	1	1	3	3	2	1
o_5	1	1	2	1	2	1	0
o_6	1	1	2	1	2	2	1
o_7	1	1	2	2	3	1	0
o_8	1	1	2	2	4	1	1

Rząd I:

z o_1 brak

z o_2 ($a_6 = 2$) $\Rightarrow (d = 1)[3]$, wyrzucamy z rozważań obiekty o_2, o_4, o_6

z o_3 brak

z o_5 brak

z o_7 brak

z o_8 ($a_5 = 4$) $\Rightarrow (d = 1)$, wyrzucamy z rozważań obiekt o_8

Rząd II:

z o_1 ($a_3 = 1$) \wedge ($a_4 = 1$) $\Rightarrow (d = 1)[2]$, wyrzucamy z rozważań obiekt o_1

z o_3 ($a_3 = 1$) \wedge ($a_5 = 2$) $\Rightarrow (d = 0)$, wyrzucamy z rozważań obiekt o_3

z o_5 $(a_5 = 2) \wedge (a_6 = 1) \Rightarrow (d = 0)[2]$, wyrzucamy z rozwazan obiekt o_5
z o_7 $(a_3 = 2) \wedge (a_5 = 3) \Rightarrow (d = 0)$, wyrzucamy z rozwazan obiekt o_7