## TREŚCI ZADAŃ

## CZĘŚĆ 1

Na tablicy decyzyjnej zaproponowanej przez prowadzącą/prowadzącego zajęcia wykonać zadania od 1 do 6.

- Określić zbiory elementarne (klasy nieodróżnialności) ze względu na cały zbiór atrybutów warunkowych.
- 2. Wypisać klasy decyzyjne i podać dolne i górne aproksymacje oraz obszary brzegowe tych klas ze względu na cały zbiór atrybutów warunkowych.
- 3. Wyznaczyć obszar pozytywny.
- 4. Wygenerować macierz rozróżnialności.
- 5. Wyznaczyć co najmniej jeden redukt i rdzeń.
- 6. Wygenerować reguły pewne, możliwe i przybliżone z wykorzystaniem algorytmu LEM2.

#### CZĘŚĆ 2

W wielu zagadnieniach data mining ważne jest określenie istotności poszczególnych atrybutów warunkowych.

Zaproponuj liczbowy współczynnik istotności atrybutu warunkowego na podstawie wpływu atrybutu na zwiększenie wielkości obszaru pozytywnego. Wyznacz ten współczynnik dla wszystkich atrybutów warunkowych z zaproponowanej przez siebie przykładowej tablicy decyzyjnej.

## Kod programu

#### **Zbiory elementarne**

```
Value
ind [
                                list [3] (S3: IndiscernibilityRelatio: List of length 3
  IND.relation
                              list [103]
                                                               List of length 103
      m 0 0 nie KIT nie nie 0... integer [1]
      m 0 0 nie KIT tak nie ... integer [1]
       m 0 0 nie KIT tak nie ... integer [1]
                                                                94
       m 0 0 nie KIT tak nie ... integer [1]
                                                                2
       m 0 0 nie KIT IIT tak n... integer [1]
                                                                 63
       m 0 0 tak KIT tak nie ... integer [1]
                                                                 80
       m 0 0 tak KIT_IIT tak n... integer [1]
                                                                49
      m 0 1 tak KIT nie nie ... integer [1]
                                                                77
       m 0 1 tak KIT tak nie ... integer [1]
```

m 0 1 tak KIT tak nie integer [1]	64	z 0 1 nie KIT tak nie 1 integer [1]	60
m 0 1 tak KIT tak nie integer [1]	18	z 0 1 nie KIT tak nie 1 integer [1] z 0 1 tak KIT tak nie 0 integer [1]	82 81
m 0 1 tak KIT tak nie integer [1]	103	z 0 2 nie KIT_IIT nie ni integer [1]	51
m 0 2 nie KIT_IIT nie n integer [1]	102	z 1 0 nie KIT nie nie 0 integer [1]	19
m 0 2 nie KIT_IIT nie n integer [1]	47	z 1 0 nie KIT tak nie 1 integer [1]	25
m 1 0 nie KIT nie nie 0 integer [1]	54	z 1 0 nie KIT_IIT nie ni integer [1] z 1 0 nie KIT_IIT tak ni integer [1]	62 1
m 1 0 nie KIT nie nie 1 integer [1]	13	z 1 0 nie KIT_IIT tak ni integer [1]	12
m 1 0 nie KIT nie nie 2 integer [1]	34	z 1 0 nie KIT_IIT tak ni integer [1]	76
m 1 0 nie KIT tak nie integer [2]	15 65	z 1 0 tak KIT tak nie 1 integer [1] z 1 0 tak KIT IIT tak ni integer [1]	5 58
m 1 0 nie KIT tak nie integer [1]	22	z 1 1 nie KIT nie nie 2 integer [1]	53
m 1 0 nie KIT tak tak integer [1]	57	z 1 1 nie KIT tak nie 0 integer [1]	98
m 1 0 nie KIT_IIT tak n integer [1]	33	z 1 1 nie KIT tak nie 0 integer [1] z 1 1 nie KIT tak nie 1 integer [2]	93 17 21
	75	z 1 1 nie KIT tak nie 1 Integer [2] z 1 1 nie KIT_IIT nie ta Integer [1]	24
m 1 0 nie KIT_IIT tak n integer [1]		z 1 1 nie KIT_IIT tak ni integer [1]	74
m 1 0 nie KIT_IIT tak n integer [1]	69	z 1 1 nie KIT_IIT tak ni integer [1]	30
m 1 0 nie KIT_IIT tak n integer [1]	86	z 1 1 nie KIT_IIT tak ni integer [1] z 1 1 nie KIT_IIT tak ta integer [1]	43 20
m 1 0 tak KIT_IIT nie n integer [1]	48	z 1 1 nie KIT_IIT tak ta integer [1]	70
m 1 0 tak KIT_IIT nie n integer [1]	46	z 1 1 tak KIT nie nie 0 integer [1]	68
m 1 0 tak KIT_IIT tak n integer [2]	37 73	z 1 1 tak KIT tak nie 2 integer [1]	89 36
m 1 1 nie KIT nie nie 0 integer [1]	83	z 1 1 tak KIT tak nie 2 integer [1] z 1 1 tak KIT_IIT tak ta integer [1]	28
m 1 1 nie KIT nie nie 2 integer [1]	16	z 1 2 nie KIT tak nie 2 integer [1]	88
m 1 1 nie KIT tak nie integer [1]	91	z 1 2 tak KIT_IIT tak ni integer [1]	11
m 1 1 nie KIT tak nie integer [1]	72	z 2 0 nie KIT tak nie 0 integer [1] z 2 0 nie KIT tak nie 1 integer [1]	100 23
m 1 1 nie KIT tak nie integer [1]	97	z 2 0 nie KIT tak tak 0 integer [1]	26
m 1 1 nie KIT tak nie integer [1]	105	z 2 0 nie KIT_IIT nie ni integer [1]	101
m 1 1 nie KIT tak tak integer [1]	79	z 2 0 nie KIT_IIT tak ni integer [1]	38 4
m 1 1 nie KIT tak tak integer [1]	32	z 2 0 nie KIT_IIT tak ni integer [1] z 2 0 tak KIT_IIT tak ni integer [1]	55
m 1 1 nie KIT_IIT nie n integer [1]	9	z 2 1 nie KIT tak nie 0 integer [1]	92
m 1 1 nie KIT_IIT tak t integer [1]	85	z 2 1 nie KIT tak nie 1 integer [1]	87
m 1 1 tak KIT tak nie integer [1]	52	z 2 1 nie KIT_IIT tak ni integer [1]	7
m 1 1 tak KIT_IIT nie n integer [1]	67	z 2 1 tak KIT tak nie 0 integer [1]	99
m 1 2 nie KIT tak nie integer [1]	41	z 3 0 nie KIT nie tak 0 integer [1]	40
m 2 0 nie KIT tak nie integer [1]	56	z 3 0 nie KIT_IIT nie ni integer [1]	39
m 2 0 nie KIT tak tak integer [1]	8	z 3 0 tak KIT_IIT nie ni integer [1]	14
m 2 0 nie KIT IIT nie n integer [1]	29	type.relation character [1]	'equivalence'
m 2 0 nie KIT_IIT tak n integer [1]	44	type.model character [1]	'RST'
m 2 0 nie KIT_IIT tak n integer [1]	61		
m 2 0 nie KIT_IIT tak n integer [1]	6		
m 2 0 tak KIT tak nie integer [2]	3 78		
m 2 1 nie KIT nie nie 1 integer [1]	84		
m 2 1 nie KIT tak tak integer [1]	35		
m 2 1 nie KIT_IIT tak n integer [1]	31		
m 2 1 nie KIT_IIT tak t integer [1]	104		
m 2 1 tak KIT nie nie integer [1]	66		
m 3 0 nie KIT tak nie integer [1]	10		
m 3 0 nie KIT IIT nie n integer [1]	27		
m 3 1 nie KIT tak nie integer [1]	96		
z 0 0 nie KIT nie nie 1 integer [1]	42		
z 0 0 nie KIT tak nie 0 integer [1]	45		
z 0 0 nie KIT tak nie 0 integer [1]	95		
z 0 0 nie KIT tak nie 0 integer [1]	50		
z 0 0 nie KIT tak nie 1 integer [1]	107		
z 0 1 nie KIT nie nie 0 integer [1]	71		
2			

#### **Analiza**

Zbiór elementarny zawiera elementy nierozróżnialne ze względu na przyjęty zbiór atrybutów .Na podstawie powyżej tabeli można dostrzec, że około 4% klas nieodróżnialności składa się z więcej niż jednego obiektu.

```
> View(ind)
> ind[["IND.relation"]][["m 1 0 nie KIT tak nie 0 nie 3-97 nie nie"]]
```

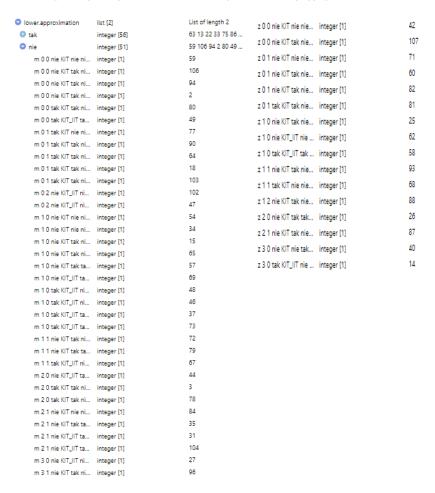
Np. 18-sta klasa nieodróżnialności od góry składa się z obiektów o numerach: 15, 65. Oznacza to, że obiekty mają te same wartości wszystkich atrybutów: płeć, wiek, czas, występ, infekcje, remisja i NT, itd.

## **Aproksymacje**

Dolna aproksymacja dla wartości atrybutów decyzyjnych "tak":

lower.approximation	list [2]	List of length 2	z 1 1 tak KIT_IIT tak integer [1]	28
tak	integer [56]	63 13 22 33 75 86	z 1 2 tak KIT_IIT tak integer [1]	11
m 0 0 nie KIT_IIT ta	integer [1]	63	z 2 0 nie KIT tak nie integer [1]	100
m 10 nie KIT nie ni	integer [1]	13	z 2 0 nie KIT tak nie integer [1]	23
m 10 nie KIT tak ni	integer [1]	22	z 2 0 nie KIT_IIT nie integer [1]	101
m 10 nie KIT_IIT ta	integer [1]	33	z 2 0 nie KIT_IIT tak integer [1]	38
m 10 nie KIT_IIT ta	-	75	z 2 0 nie KIT_IIT tak integer [1]	4
m 10 nie KIT_IIT ta	integer [1]	86	z 2 0 tak KIT_IIT tak integer [1]	55
m 11 nie KIT nie ni	• • • •	83		92
m 11 nie KIT nie ni	• • • •	16	z 2 1 nie KIT tak nie integer [1]	
m 11 nie KIT tak ni	•	91	z 2 1 nie KIT_IIT tak integer [1]	7
m 11 nie KIT tak ni	•	97	z 2 1 tak KIT tak nie integer [1]	99
m 11 nie KIT tak ni	•	105	z 3 0 nie KIT_IIT nie integer [1]	39
m 11 nie KIT tak ta	• • • • • • • • • • • • • • • • • • • •	32		
m 11 nie KIT_IIT ni	•	9		
m 11 nie KIT_IIT ta	-	85 52		
m 1 1 tak KIT tak ni	-	41		
m 12 nie KIT tak ni m 20 nie KIT tak ni	•	56		
m 20 nie KIT tak ni m 20 nie KIT tak ta	-	8		
m 2 0 nie KIT_IIT ni	• • • •	29		
m 2 0 nie KIT_IIT ta	• • • •	61		
m 2 0 nie KIT_IIT ta	•	6		
m 2 1 tak KIT nie ni	•	66		
m 3 0 nie KIT tak ni	-	10		
z 0 0 nie KIT tak nie	• • • • • • • • • • • • • • • • • • • •	45		
z 0 0 nie KIT tak nie	integer [1]	95		
z 0 0 nie KIT tak nie	integer [1]	50		
z 0 2 nie KIT_IIT nie	integer [1]	51		
z 1 0 nie KIT nie nie	integer [1]	19		
z 1 0 nie KIT_IIT tak	integer [1]	1		
z 1 0 nie KIT_IIT tak	integer [1]	12		
z 1 0 nie KIT_IIT tak	integer [1]	76		
z 1 0 tak KIT tak nie	integer [1]	5		
z 1 1 nie KIT nie nie	integer [1]	53		
z 1 1 nie KIT tak nie	integer [1]	98		
z 1 1 nie KIT tak nie	integer [1]	17		
z 1 1 nie KIT tak nie	integer [1]	21		
z 1 1 nie KIT_IIT nie	-	24		
z 1 1 nie KIT_IIT tak	•	74		
z 1 1 nie KIT_IIT tak	-	30		
z 1 1 nie KIT_IIT tak	-	43		
z 1 1 nie KIT_IIT tak	-	20		
z 1 1 nie KIT_IIT tak	•	70		
z 1 1 tak KIT tak nie	• • • •	89		
z 1 1 tak KIT tak nie	integer [1]	36		

Dolna aproksymacja dla wartości atrybutów decyzyjnych "nie":



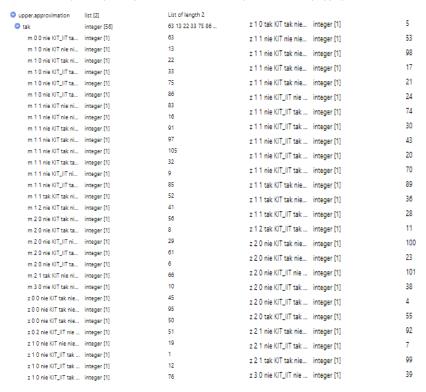
#### **Analiza**

Dolna aproksymacja to zbiór będący sumą zbiorów elementarnych, które bez wątpienia są objęte danym nieostrym pojęciem, dlatego pewne jest, że program zdecyduje prawidłowo.

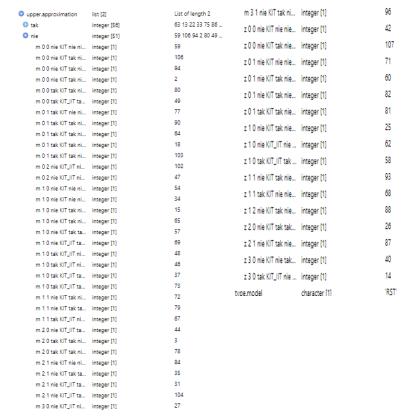
Analizując poszczególne obiekty z tabeli należące do dolnego przybliżenia nie mamy więc wątpliwości, że spełniają kryteria przynależności do danej klasy.

W dolnej aproksymacji zawierają się obiekty z jednoelementowych zbiorów elementarnych lub obiekty z wieloelementowych zbiorów elementarnych, w których wartość atrybutu decyzyjnego dla wszystkich obiektów jest taka sama.

Górna aproksymacja dla wartości atrybutów decyzyjnych "tak":



Górna aproksymacja dla wartości atrybutów decyzyjnych "nie":



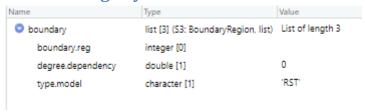
#### **Analiza**

Do górnego przybliżenia należą obiekty, których nie można wykluczyć, że są objęte danym nieostrym pojęciem, dlatego nie jest pewne, że program zdecyduje prawidłowo. Jest to spowodowane tym, że

należą do klas abstrakcji mających niepuste przecięcie z danym nieostrym pojęciem. Są zatem nierozróżnialne z pewnymi obiektami należącymi do tego pojęcia.

Warto zwrócić uwagę na fakt, że dolna aproksymacja zawiera się w górnej. Na tej podstawie można stwierdzić, że górne przybliżenie jest zbiorem większym i patrzącym na obiekty w "szerszej perspektywie".

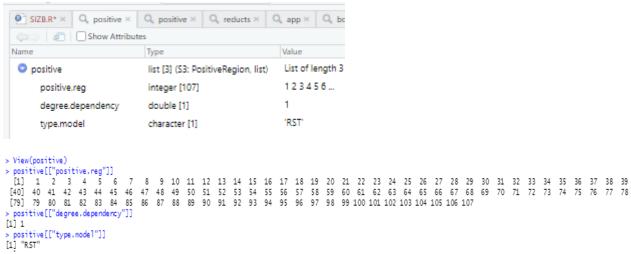
## **Obszar brzegowy**



#### **Analiza**

Obszarem brzegowym nazywamy te obiekty, co do których nie mamy pewności, czy reprezentują dane pojęcie. W danym przypadku różnica miedzy górnym a dolnym przybliżeniem wyszła równa 0, oznacza to, że w danym zbiorze nie występuje obszar niespójności.

# **Obszar pozytywny**



#### **Analiza**

Obszar pozytywny jest sumą dolnych przybliżeń przez co do tej grupy należą wszystkie obiekty dla których mamy pewność, że należą do pewnej wartości klasy decyzyjnej.

Dla powyższych obiektów obszar pozytywny dał nam wynik pokrywający się z całością zbioru startowego, co oznacza niesprzeczność i ostrość zbioru.

#### Wygenerowana macierz

```
> dm[["disc.mat"]]
                                          Plec, Wiek, Rodzaj, HbA1c,
Plec, Wiek, Wystep, Rodzaj,
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            Plec, Wiek, Rodraj, HbA
Plec, Wiek, Rodraj, HbALc,
Wiek, Remisja, HbA
kk, Czas, Rodraj, Infekcje, HbALc,
Zeas, Rodraj, Lange, Lange,
Zeas, Rodraj, HbALc,
Zeas, Rodraj, Ha
              V18 V19 V20 V21 V22 V23 V24 V
NA NA NA NA NA NA NA NA NA
NA NA NA NA NA NA NA
NA NA NA NA NA NA NA
NA NA NA NA NA NA NA
NA NA NA NA NA NA NA
NA NA NA NA NA NA NA
NA NA NA NA NA NA NA
NA NA NA NA NA NA
NA NA NA NA NA NA
NA NA NA NA NA NA
NA NA NA NA NA NA
NA NA NA NA NA NA
NA NA NA NA NA NA
NA NA NA NA NA NA
NA NA NA NA NA NA
NA NA NA NA NA NA
NA NA NA NA NA NA
NA NA NA NA NA NA
NA NA NA NA NA NA
NA NA NA NA NA NA
NA NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA NA NA
NA NA NA 
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   omitted 98 rows ]
```

Przy pomocy wygenerowanej macierzy jesteśmy w stanie wyznaczyć rdzeń i redukty.

#### Rdzeń i redukty

```
> reducts[["decision.reduct"]]
Sreduct1
A feature subset consisting of 9 attributes:
Plec, Wiek, Czas, Wystep, Rodzaj, Remisja, HbAlc, NT, TG
A feature subset consisting of 9 attributes:
Plec, Wiek, Czas, Wystep, Rodzaj, Remisja, HbA1c, Mc, TG
Sreduct3
A feature subset consisting of 9 attributes:
Plec, Wiek, Czas, Wystep, Rodzaj, Remisja, HbAlc, Ch, TG
A feature subset consisting of 10 attributes:
Plec, Wiek, Czas, Wystep, Rodzaj, Infekcje, HbAlc, NT, Mc, TG
A feature subset consisting of 10 attributes:
Plec, Wiek, Czas, Rodzaj, İnfekcje, HbAlc, NT, Mc, Ch, TG
Sreduct6
A feature subset consisting of 10 attributes:
Plec, Wiek, Czas, Wystep, Rodzaj, Infekcje, HbA1c, Mc, Ch, TG
```

#### Analiza

W powyższym przypadku otrzymano sześć reduktów. Analizując otrzymane wyniki okazuje się, że nie wszystkie atrybuty są potrzebne w procesie podejmowania decyzji. W trzech pierwszych przypadkach potrzeba tylko dziewięć atrybutów do poprawnego przydzielania obiektów do poszczególnych klas, zaś w pozostałych przypadkach potrzeba ich aż 10.

Podsumowując, można więc powiedzieć, że redukty ograniczają liczbę atrybutów do minimum, niezbędnego do poprawnego przydzielania obiektów do poszczególnych klas.

		-
core	character [6]	'Czas' 'HbA1c' 'Plec' 'Rodzaj' 'TG' 'Wiek'
discernibility.type	character [1]	'RST'
type.task	character [1]	'computation of all reducts'
type.model	character [1]	'RST'

#### **Analiza**

Rdzeń jest częścią wspólną wszystkich reduktów systemu informacyjnego. W danym przypadku zbiorem atrybutów niezbędnych dla zachowania rozróżnialności obiektów w systemie są: Czas, HbA1c, Plec, Rodzaj, TG, Wiek.

## **Algorytm LEM**

Algorytm LEM2 generuje reguły decyzyjne, które pozwalają podejmować decyzje klasyfikacji dla całkiem nowych obiektów spoza tych w tabeli decyzyjnej.

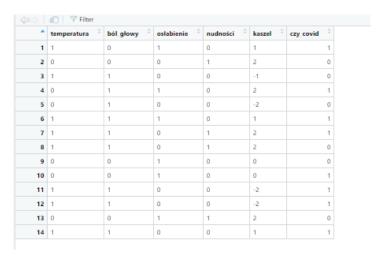
# Ogólna analiza

- Dane do zadania składały się z 12 atrybutów warunkowych i jednego decyzyjnego.
- Przybliżenia dolne zawierały obiekty, w których jesteśmy pewni decyzji.
- Do górnej aproksymacji trafiły obiekty, co do których nie było pewności.
- Obie aproksymacje posiadały takie same obiekty, dlatego różnica tych zbiorów tj. obszar brzegowy wynosi 0.
- Obszar pozytywny dał wynik pokrywający się z całością zbioru startowego co oznacza niesprzeczność i ostrość zbioru.
- Wyznaczono 6 reduktów i 6-cio elementowy rdzeń, który jest iloczynem reduktów.
- Rdzeń jest niezbędny do prawidłowego sklasyfikowania każdego obiektu z danych.
- Algorytm Lem2 wygenerował 35 reguł. Można zauważyć że po redukcji zbioru atrybutów warunkowych liczba reguł nie zmieniła się, gdyż do klasyfikacji obiektów używa się tylko atrybutów zawartych w reduktach.

- Teoria zbiorów przybliżonych pozwala na identyfikowanie redundancji danych w tablicach informacyjnych oraz jej eliminację tworzenie reduktów.
- Klasą decyzyjna nazywamy zbiór obiektów o zbliżonej złożoności obliczeniowej potrzebnej do podjęcia decyzji.
- Aproksymacja zbioru określa nam dokładny sposób klasyfikowania obiektów naszego zbioru danych. Dolne przybliżenie zawiera obiekty pewne co do swojej decyzji, zaś górne przybliżenie niepewne – może, ale nie musi mieć zawsze jednoznacznej decyzji.
- W celu precyzyjnego i konkretnego opisania relacji pomiędzy obiektami występującymi w bazie wiedzy, stosuje się redukcję liczby atrybutów opisujących owe relacje.
- Algorytm Lem2 pozwala nam na wygenerowanie reguł decyzyjnych z aproksymacji tablicy decyzyjnej. Jeżeli są to dolne aproksymacje generuje on reguły pewne, jeśli górne - reguły niepewne (Zamiast aproksymacji można podać zbiór brzegowy, wtedy zostaną wygenerowane reguły przybliżone).
- Teoria zbiorów przybliżonych świetnie nadaje się do reprezentacji pojęć niespójnych.
- Pozwala redukować wiedzę do niezbędnego minimum.
- Pozwala generować reguły minimalne które skracają drogę wnioskowania.
- Dziedziny, w których teoria ta została zastosowana to nie tylko medycyna czy biznes (bankowość, badania rynku) ale również rozpoznawanie mowy, sieci neuronowe czy ogólnie mówiąc sztuczna inteligencja

# Część 2

Eksploracja danych (data mining) to jeden z etapów procesu uzyskiwania wiedzy z baz danych, przeznaczony do badania dużych zasobów danych w poszukiwaniu regularnych wzorców oraz systematycznych współzależności pomiędzy zmiennymi.



Rys. Tablica decyzyjna.

# Obszar pozytywny danych:

> View(positive)

```
> positive[["positive.reg"]]
[1] 1 2 3 4 5 6 13 14
> positive[["degree.dependency"]]
[1] 0.5714286
> positive[["type.model"]]
```

[1] "RST"

## Obliczanie wpływu atrybutów warunkowych na wyniki

```
$decision.reduct
$decision.reduct$reduct1
A feature subset consisting of 3 attributes:
temperatura, ból_głowy, kaszel
$decision.reduct$reduct2
A feature subset consisting of 3 attributes:
temperatura, nudności, kaszel
$core
[1] "kaszel" "temperatura"
$discernibility.type
[1] "RST"
$type.task
[1] "computation of all reducts"
$type.model
[1] "RST"
attr(,"class")
[1] "ReductSet" "list"
```

Na podstawie powyższych danych można zauważyć, że rdzeń zawiera takie atrybuty jak "kaszel" oraz "temperatura". Dane atrybuty mają więc znaczny wpływ na klasyfikację obiektów.

Propozycja wyliczania współczynnika istotności dla atrybutów warunkowych:

- Jeżeli zmiana wartości atrybutu na przeciwny, zmieni wartość atrybutu decyzyjnego , dodawane są do atrybutu "punkty istotności".
- Następnie po wykonaniu owej operacji na wszystkich obiektach, dzielę punkty istotności przez sumę obiektów.

Po wykonaniu tych operacji otrzymano:

- Współczynnik istotności dla atrybutu "temperatura": 0.43
- Współczynnik istotności dla atrybutu "ból głowy": 0.21
- Współczynnik istotności dla atrybutu "osłabienie": 0
- Współczynnik istotności dla atrybutu "nudności": 0.64
- Współczynnik istotności dla atrybutu "kaszel": 0.71

Okazuje się, że kaszel jest najważniejszym atrybutem warunkowym. Zmiana wartości na przeciwną przyczyniłaby się do zmiany wartości atrybutu decyzyjnego. Wartość osłabienie jest atrybutem neutralnym. Nie wpływa ono na wykrycie Covid'a. Jego usunięcie nie

zmieniłoby wyniku w klasyfikacji. Można więc powiedzieć, że nie występuje też ono w żadnym redukcie.

## Wnioski

Dzięki data eksploracji danych możliwe jest przewidywanie i klasyfikacja obiektów w przyszłości, dlatego też predykcyjny data mining jest bardzo popularny, natomiast aby go wykonywać prawidłowo potrzebne jest określenie istotności poszczególnych atrybutów warunkowych.

## Pliki:

- grypa.csv- zawiera dane do zadania 2 czy\_covid
- diabetes.csv
- Zad2\_SI.R
- **♣** SIZB.R