

# Algorytm grupowania oparty o łańcuch reguł dyskryminacyjnych

Dariusz Mazur

Silesian University of Technology, Faculty of Organisation and Management,  
ul. Roosevelta 26, 41-800 Zabrze, Poland  
e-mail: dmazur@polsl.gliwice.pl

---

## Streszczenie

Grupowanie jako technika eksploracji danych jest szeroko stosowana. Opiera się ona o algorytmy grupowania, których przydatność ściśle zależy od postaci i charakteru danych wejściowych. Niniejszy artykuł przedstawia metodę grupowania danych o postaci symbolicznej. Zaprezentowano nowy rodzaj prezentacji wyniku grupowania przy pomocy listy reguł dyskryminacyjnych.

---

## 1 Introduction

Zagadnienie grupowania jest przedmiotem zainteresowań badaczy z zakresu statystyki [10], systemów uczących [11, 7], baz danych [20]. Podstawowy podział metod grupowania omówiony został w [17], zastosowanie do eksploracji baz danych w [6]. Stosuje się również techniki wykorzystujące metody należące w różnych obszarów, przykładowo w [16]. W tym zakresie należy wspomnieć badania nad określeniem miar bliskości danych symbolicznych, tak często spotykanych w transakcyjnych bazach danych [1, 2, 5, 9]. Samo zagadnienie zastosowanie grupowania jako problem znajdowania interesujących struktur wśród zgromadzonych danych zaprezentowano w [22], w [4] zaprezentowano algorytm grupowania rozmytego jako szczególny przypadek algorytmu klasyfikacji opisanego w [3]. Innego rodzaju badania w celu uogólnienia idei grupowania przedstawiono w [19]. Dotyczyły one koncepcji identyfikacji i ekstrakcji pojedynczych klastrów zamiast dokonywania kompletnego rozdziału zbioru na ustaloną ilość grup. Rozszerzeniu podlega również dziedzina poddawana grupowaniu. Początkowo większość prac dotyczyła danych numerycznych, jednak z uwagi na to iż w bazach danych niezwykle często stosuje się dane postaci opisowej lub symbolicznej, w kręgu zainteresowań badaczy są również dane nienumeryczne [14, 15, 12].

Grupowanie powszechnie traktuje się jako uczenie bez nadzoru, w którym uczeń otrzymuje zbiór trenujący, zawierający przykłady bez określania ich kategorii (przykłady nieetykietowane). Właściwe kategorie uczeń musi zaproponować samodzielnie na podstawie pewnej zasady grupowania, wbudowanej w algorytm lub częściowo definiowanej przez użytkownika [7].

## 2 zakres pracy

Układ niniejszej pracy jest następujący: W sekcji 3 przedstawiono ujęcie zagadnienia grupowania jako zagadnienie uczenia maszynowego, w sekcji 4 przedstawiono podstawowy podział metod grupowania, ze szczególnym uwzględnieniem metod hierarchicznych. W sekcji 5 przedstawiono ideę list decyzyjnych, następnie zaprezentowano metodę reprezentacji wyniku grupowania w postaci list decyzyjnych, będących szczególnym przypadkiem metod hierarchicznych. Metoda reprezentacji wyników grupowania, a zwłaszcza przedstawiony

algorytm stanowią wkład własny autora. Sekcja 8 omawia poszczególne elementy związane z grupowaniem opartym o listy decyzyjne, są to kolejno zastosowanie entropii jako funkcji oceny grupowania oraz algorytm poszukiwania rozwiązania optymalnego. W sekcji 9 zostaną przedstawione rezultaty grupowania metodą list decyzyjnych w porównaniu do innych popularnych algorytmów. W sekcji 10 omówione są dotychczasowe wnioski i spostrzeżenia autora oraz kierunki dalszych prac związanych z zaprezentowaną metodą grupowania.

W niniejszej pracy zostanie przedstawiona nowa metoda grupowania

## 3 grupowanie jako przeszukiwanie

W większości algorytmów grupowania można doszukać się analogii do pewnego rodzaju uczenia się opartego na przeszukiwaniu przestrzeni hipotez w celu znalezienia hipotezy spełniającej pewne kryteria. Z tej perspektywy można wyróżnić cztery aspekty procesu grupowania, które to charakteryzują poszczególne algorytmy grupowania:

1. reprezentacja grupowania, czyli postać wyniku,
2. operatory używane do poruszania się w przestrzeni rozwiązań, umożliwiające sprawdzenie innych, również lepszych rozwiązań,
3. heurystyczna funkcja oceny grupowania, używana do kierowania procesem przeszukiwania,
4. strategia przeszukiwania, czyli opis sposobu wykorzystania operatorów i funkcji heurystycznej do poszukiwania najlepszego rozwiązania

Grupowanie jest problemem, którego rozwiązanie sprowadza się do rozwiązania następująco sformułowanego zadania:

*identyfikacja naturalnych grup, w których obiekty podobne do siebie mają zostać umieszczone w jednej grupie — natomiast obiekty znacznie się różniące w różnych.*

## 4 grupowanie hierarchiczne

Stosowane procedury grupowania można zaliczyć w większości to jednej z poniższych technik [17]:

- *optymalizacyjno-iteracyjnych*, za pomocą których dokonuje się podziału zbioru na  $m$  podzbiorów, przy czym wartość parametru  $m$  jest zadawane, algorytmy te polegają na minimalizacji funkcji kryterium,

- *K-średnich* (ang. *K-means*) - grupy reprezentowane są przez „środek ciężkości”
- *K-mediana* (ang. *K-medoids*) - grupy reprezentowane są przez jeden z obiektów

- *hierarchicznych*, w ramach których skupienia tworzą drzewa, gdzie liście reprezentują poszczególne obiekty, a węzły — ich grupy. Skupienia wyższego poziomu zawierają skupienia niższego poziomu. W ramach metod hierarchicznych, w zależności od sposobu konstruowania hierarchii klas można wyodrębnić:

- *metody aglomeracyjne*,
- *metody podziałowe* (deglomeracyjne).

- metody oparte o *algorytmy genetyczne*

- metody oparte o *sieci neuronowe*,

Proces grupowania w ramach metod podziałowych przebiega w odwrotnym kierunku niż w metodach aglomeracyjnych. Rozpoczyna się od jednego zbioru obejmującego wszystkie obiekty a następnie dzieli się go w kolejnych krokach na podzbiory, aż do uzyskania zbioru klastrów zawierających pojedyncze elementy. Problem określania kryterium podziału jest znacznie trudniejszy niż przy łączeniu. Należy bowiem rozważyć wszystkie możliwe podziały, uwzględniając wszystkie atrybuty obiektów. Najczęściej jednak, z uwagi na znacznie łatwiejsze do rozwiązania, stosuje się metody oparte o podział względem tylko jednej zmiennej w danym kroku. Ogólną postać algorytmu podziałowego można zapisać następująco:

- 1) Utwórz jeden zbiór (klastr) zawierający wszystkie obiekty.
- 2) Wygeneruj wszystkie możliwe podziały klastra na dwa lub więcej podzbiory.
- 3) znajdź najlepszy podział zgodnie z funkcją kryterium i zrealizuj go; otrzymane podzbiory stają się nowymi klastrami w miejsce źródłowego.
- 4) Jeżeli każdy z obiektów znajduje się w oddzielnej klasie, zakończ; w przeciwnym wypadku wykonuj rekurencyjnie kroki 2-4 dla każdego klastra.

Algorytmy podziałowe lepiej nadają się do grupowania obiektów opisanych atrybutami symbolicznymi niż numerycznymi. Kolejnym ograniczeniem jest duża złożoność pamięciowa i obliczeniowa algorytmów tej klasy. A rezultaty osiągane przy ich pomocy zazwyczaj są gorsze od osiąganych metodami aglomeracyjnymi [13].

## 5 Listy decyzyjne

Listy decyzyjne stanowią istotny element maszynowego uczenia się. Zostały zaproponowane po raz pierwszy w [21]. Listy

decyzyjne są formą zbioru reguł decyzyjnych w którym ustalono porządek, czyli kolejność w jakiej reguły mają być wykorzystane do klasyfikowania przykładów. Hipoteza reprezentowana przez taki zbiór reguł przyporządkowuje klasyfikowanemu przykładowi kategorię związaną z pierwszą w kolejności regułą, która ten przykład pokrywa. Listy decyzyjne można również traktować jak zdegenerowane drzewo decyzyjne, dla którego z każdego węzła wychodzą dwie gałęzie, z których przynajmniej jedna dochodzi do liścia a ewentualnie pozostała do innego węzła.

### 5.1 definicja listy decyzyjnej

Reguły stosowane są do reprezentacji wiedzy o pojęciach klasyfikujących przykłady. Reguły składają się dwóch elementów: części warunkowej i części decyzyjnej w zapisie:

$$\text{JEŚLI warunek TO kategoria} \quad (1)$$

lub bardziej matematycznie:

$$\text{warunek} \longrightarrow \text{kategoria} \quad (2)$$

Składnik *warunki* jest formułą logiczną nałożoną na atrybuty opisujące zbiór obiektów, natomiast *kategoria* stanowi proste przypisanie obiektu do określonej kategorii. Można to przedstawić następująco:

$$(\forall x \in \mathbf{X})(\alpha[a_1(x), a_2(x), \dots, a_r(x)] \rightarrow h(x) = d). \quad (3)$$

gdzie:

- $a_i(x)$  poszczególne wartości atrybutów obiektu  $x$
- $\alpha(x)$  formuła logiczna odwołująca się do wartości atrybutów obiektu
- $d \in \mathbb{D}$  etykieta konkretnej kategorii

Jeżeli dany jest zbiór formuł logicznych  $\alpha_i \rightarrow \alpha_i \in \mathcal{A}$ , w oparciu o które tworzy się reguły  $r_i = \alpha_i(x) \rightarrow d_{\alpha_i}$  to mając uporządkowany zbiór takich reguł  $r_1, r_2, \dots, r_m$  tworzymy listę decyzyjną. Wyznaczanie przypisania obiektu do kategorii dokonuje się następująco. Dla danego obiektu  $x$  sprawdza czy spełnia on warunek  $\alpha_1$ , jeżeli tak to obiekt  $x$  zostaje przypisany do kategorii  $d_{\alpha_1}$  w przeciwnym przypadku dokonuje się analogicznej analizy przy pomocy następnej reguły. Zapis algorytmiczny można przedstawić przy pomocy zagnieżdżonej instrukcji warunkowej:

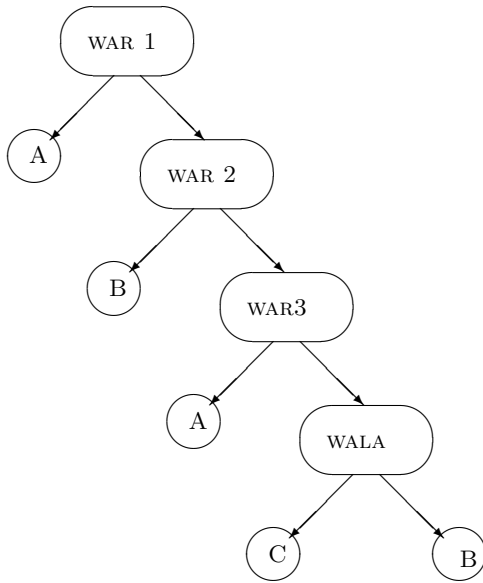
$$\begin{aligned} \text{IF } \alpha_1(x) \text{ THEN } & h(x) = d_{\alpha_1} \\ \text{ELSE IF } \alpha_2(x) \text{ THEN } & h(x) = d_{\alpha_2} \\ & \dots \\ & \dots \\ \text{ELSE IF } \alpha_m(x) \text{ THEN } & h(x) = d_{\alpha_m} \end{aligned}$$

Lista decyzyjna jest listą reguł na rys 5.1, ale często zapisuje się ją w postaci listy par *jwarunek, kategoria* czyli:

$$DL = (\alpha_1, d_{\alpha_1}), (\alpha_2, d_{\alpha_2}), \dots, (\alpha_m, d_{\alpha_m}). \quad (4)$$

Formuły warunku można zapisywać przy pomocy kompleksów. Kompleks jest zbiorem selektorów  $\mathbf{k} = \langle k_1, k_2, \dots, k_m \rangle$ , poszczególne selektory odpowiadają kolejnym atrybutom opisującym obiekt. Każdy selektor określa zbiór wartości dozwolonych.

**Definicja 5.1** Selektor odpowiadający atrybutowi  $a : X \rightarrow A$  pokrywa przykład  $x \in X$ , jeśli  $\alpha(x) \in V_s$ , przy czym  $V_s$  oznacza zbiór wartości dozwolonych dla selektora  $s$ . Piszemy wówczas  $s \triangleright x$  [7].



Rysunek 1. Lista reguł.

**Definition 5.2** *Kompleks  $k = \langle s_1, s_2, \dots, s_m \rangle$  pokrywa przykład  $x \in X$  jeśli każdy selektor  $s_i$ , gdzie  $1 \leq i \leq m$  pokrywa przykład  $x$ .*

W niniejszym opracowaniu, na potrzeby przedstawionego algorytmu, używany jest tylko jeden, specyficzny rodzaj kompleksów, zwanych kompleksami atomowymi. Formalna definicja jest następująca:

**Definition 5.3** *Kompleksem atomowym nazywa się każdy kompleks zawierający dokładnie jeden selektor pojedynczy lub dysjunkcyjny a pozostałe selektory są selektorami uniwersalnymi*

Inaczej rzecz ujmując, formuła logiczna składa się z prostego testu wartości dla pojedynczego atrybutu. Takie ograniczenie pozwala znacząco zmniejszyć ilość reguł poddawanych analizie.

## 6 reprezentacja grupowania

W algorytmach hierarchicznych rozwiązanie tworzył graf w postaci odwróconego drzewa (dendogramu). W takim drzewie liście reprezentują poszczególne obiekty a gałęzie wyznaczają poszczególne grupy. Ilość grup zależy od wysokości drzewa, grupy wyznaczają wszystkie gałęzie na danym poziomie. Liście należące do różnych gałęzi reprezentują obiekty należące do różnych grup.

W proponowanej metodzie zaproponowano istotną zmianę: graf zostanie utworzony na podstawie listy opisów dyskryminujących, analogicznie jak to zostało zaproponowane w algorytmie CN2 [8]. Zastosowano również taką samą miarę jakości grupowania opartą zaczerpniętą z teorii informacji funkcję entropii. Jak już wspomniano uzyskiwane rozwiązanie ma postać uporządkowanej listy reguł decyzyjnych. Pojęcie reguł zostało zaczerpnięte z dziedziny symbolicznej klasyfikacji wzorcowej. Reguły te mają formę:

$$D_j ::> K_i, \quad (5)$$

gdzie  $D_j$  to reguła opisująca,  $K_i$  oznacza klasę a  $::>$  operator przypisania. Reguły można interpretować następująco: *Jeżeli dany obiekt spełnia opis  $D_j$  to należy do klasy  $K_i$ .* W celu adaptacji pojęcia reguł do grupowania zbiorów tekstowych przyjęto następujące założenie. Reguła będzie miała postać występowania określonego słowa (termu) w tekście, tj. jeżeli słowo użyte w regule  $D_j$  występuje w tekście obiektu  $X_n$  to znaczy to, że obiekt  $X_n$  zostaje przypisany do klasy  $K_i$ . Reguły mają postać uporządkowanej listy. Proces przypisywania obiektów do grup przebiega następująco:

dla każdego obiektu:

jeżeli reguła umieszczona w głowie listy jest spełniona to przypisz obiekt do wskazanej grupy w przeciwnym wypadku powtórz obliczenie dla ogona listy

Lista przeglądana jest tak długo aż zostanie znaleziona odpowiadająca reguła, z tego też wynika że lista powinna zawierać odpowiednią ilość reguł tak aby każdy obiekt mógł zostać przypisany. Listę tą można traktować jako gen będący przedmiotem przetwarzania w algorytmach genetycznych.

Otrzymana postać grupowania to graf w postaci zdegenerowanego drzewa, w którym z każdego węzła wychodzi tylko jedna gałąź oraz jeden lub więcej liści. Liście reprezentują obiekty przypisane do danej grupy, przy czym grupa obejmuje obiekty obecne w jednym lub więcej węzle. Jest to istotny element odróżniający od metod hierarchicznych.

## 7 funkcja kryterium

Jako funkcje kryterium często wykorzystuje się wzory zaczerpnięte z teorii informacji. Zakłada się, że takie grupowanie, które daje największy przyrost informacji jest optymalne, gdyż odpowiada to małemu zróżnicowaniu kategorii w podzbiorach. Informację zawartą z zbiorze etykietowanych przykładów  $P$  można wyrazić następująco:

$$I(P) = \sum_{d \in \mathcal{D}} -\frac{|P^d|}{P} \log \frac{|P^d|}{P}, \quad (6)$$

przy czym logarytm może mieć dowolną podstawę, lecz konsekwentnie tę samą.

Entropię przykładów  $P$  ze względu na wynik  $r$  testu  $t$  oblicza się następująco:

$$E_{tr}(P) = \sum_{d \in \mathcal{D}} -\frac{|P_{tr}^d|}{P_{tr}} \log \frac{|P_{tr}^d|}{P_{tr}}, \quad (7)$$

Entropia zbioru przykładów  $P$  ze względu na test  $t$  jest definiowana jako średnia ważona entropia dla poszczególnych testów:

$$E_t(P) = \sum_{r \in R_t} \frac{|P_{tr}|}{|P|} E_{tr}(P). \quad (8)$$

Przyrost informacji jest określony jako różnica:

$$g_t(P) = I(P) - E_t(P) \quad (9)$$

Ponieważ jednak informacja  $I(P)$  ma wartość niezależną od ocenianego testu i właściwą dla zbioru przykładów  $P$ , jako kryterium wystarczy zastosować minimalizację entropii  $E_{tr}(P)$ .

Dla zbioru przykładów przyrost informacji może być obliczony następująco:

$$\begin{aligned}
 g_t(P) = & - \sum_{d \in C} (c(x) = d) \cdot \log(c(x) = d) \\
 & + \sum_{d \in C} (c(x) = d | a_i(x) = v) \cdot \log(c(x) = d | a_i(x) = v) \\
 & + \sum_{d \in C} (c(x) = d | a_i(x) \neq v) \cdot \log(c(x) = d | a_i(x) \neq v).
 \end{aligned}
 \tag{10}$$

### 7.1 Własności testu opartego o przyrost informacji

Entropia przyjmuje maksymalne wartości wtedy gdy rozkład wartości atrybutów jest równomierny, natomiast osiąga minimum gdy obiekty posiadające atrybuty o danej wartości zgromadzone są w jednej grupie. Obiekty są do siebie podobne wtedy gdy posiadają równe wartości swych atrybutów. Tym samym stosowanie funkcji oceny przyrostu informacji jako poszukiwania podziału spełniającego warunek maksymalizacji wartości tej funkcji pokrywa się z celem grupowania.

$$\arg \max_t g_t(P) \tag{11}$$

## 8 algorytm

### 8.1 poruszanie się po przestrzeni rozwiązań

Najprostrzym i najpewniejszym algorytmem możliwym do zastosowania do grupowania jest procedura pełnego przeglądu. Ponieważ znana jest ilość reguł wchodzących w skład listy decyzyjnej wystarczy dokonać przeglądu wszystkich kombinacji ułożenia reguł w liście, a następnie wybrać takie ułożenie, które daje najlepsze dopasowanie funkcji kryterium. Największą wadą takiego podejścia jest wysoka złożoność obliczeniowa, już dla kilkunastu reguł czas obliczeń przestaje być akceptowalny. Natomiast w przypadku mniejszych zbiorów algorytm pełnego przeglądu może stanowić model wzorcowy, do weryfikacji innych algorytmów.

Lista reguł  $L$  zawiera w kolejnych węzłach wszystkie możliwe kombinacje reguł i przypisań do poszczególnych grup to ostatecznym rezultatem grupowania jest jedna (lub wiele równoważnych) permutacji takiej listy wynikających ze zmiany ułożenia kolejności węzłów. Jeżeli przez  $|D|$  oznaczymy całkowitą ilość reguł a  $|K|$  oczekiwana ilość grup to długość listy reguł wynosi:

$$|L| = |D| * |K| \tag{12}$$

gdzie ilość reguł dyskryminacyjnych można wyznaczyć następująco:

$$|D| = \sum_{i=1}^m |A_i| \tag{13}$$

gdzie:

- $A_i$  - atrybut opisujący obiekt
- $m$  - ilość atrybutów
- $|A_i|$  - liczebność dziedziny atrybutu  $A_i$

Ilość reguł dyskryminacyjnych w przypadku krytycznym może osiągnąć wartość iloczynu ilości obiektów i ilości atrybutów opisujących, natomiast ilość permutacji wynosi  $(|L|)!$

to przeszukiwany obszar rozwiązań może wydawać się bardzo duży. Wykorzystywanie takich właściwości pozwala na uniknięcie dokonywania niepotrzebnych obliczeń i zwiększenie wydajności algorytmu.

Szczegółowa analiza pokazuje że można go znacząco ograniczyć z uwagi na następujące właściwości:

- Rzeczywista ilość reguł jest znacznie mniejsza od wspomnianego powyżej przypadku krytycznego, aby mówić o grupowaniu to jednak wiele wartości atrybutów musi się powtarzać (obiekty muszą mieć cechy wspólne, być do siebie 'podobne').
- Część listy reguł jest nieaktywna, gdyż podczas procesu przypisywania brane są pod uwagę tylko czołowe reguły i można określić, która z nich jako ostatnia dzieli listę na dwie części. Kolejność reguł w drugiej części, nieaktywnej nie ma znaczenia dla procesu przypisywania gdyż reguły te nigdy nie biorą udziału w procesie.
- Istnienie reguł nieistotnych w danym kontekście, tj. takich których możliwe do spełnienia przypisania są w pełni pokryte przez wcześniejsze reguły (czyli dany kontekst). Zjawisko to występuje wtedy gdy wszystkie obiekty objęte kompleksem z danej reguły zostały wcześniej (przez reguły umieszczone bliżej głowy listy) przypisane do grup. Tym samym ułożenie takich reguł pozostaje bez znaczenia na rezultat.

Wykorzystywanie takich właściwości pozwala na uniknięcie dokonywania niepotrzebnych obliczeń i zwiększenie wydajności algorytmu.

### 8.2 dane wejściowe i inicjalizacja

Danymi wejściowymi są oczekiwana ilość grup oraz lista reguł potrzebnych do skonstruowania listy decyzyjnej. Inicjalizacja polega na wygenerowaniu listy reguł. Długość listy równa jest iloczynowi wszystkich występujących słów i ilości oczekiwanej listy grup. Wynika to z postaci reguły. W wyniku tej generacji powinna zostać uzyskana lista, w oparciu o którą funkcja przypisywania podzieli zbiór na wymaganą ilość grup.

## 9 eksperyment

### 9.1 test na małej bazie

W tabeli 1 przedstawiono sztucznie wygenerowaną bazę, którą poddano testom. Baza zawiera 6 obiektów i 6 atrybutów binarnych (przyjmujących wartości 0 – brak i 1 – obecny). Celem jest znalezienie najlepszego podziału zbioru obiektów na dwie grupy. Zbiór został sztucznie spreparowany, posiada on specyficzny rozkład wartości atrybutów. W trakcie analizy podanego przykładu można zauważyć, że atrybut  $a$  występuje (ma wartość 1) w 3 obiektach, a atrybut  $b$  w pozostałych trzech. Są to najliczniej występujące atrybuty i do tego zupełnie rozdzielone więc w prosty sposób można na ich podstawie dokonać podziału na dwie grupy  $\{\{x_1, x_2, x_3\}, \{x_4, x_5, x_6\}\}$ . Dokładna analiza wskazuje, że pozostałe atrybuty też są ułożone wg określonego porządku. Dokonanie następującego podziału  $\{\{x_1, x_2, x_4\}, \{x_3, x_5, x_6\}\}$  powoduje że żaden z pozostałych atrybutów (atrybuty  $c, d, e, f$ ) nie występuje w obu grupach jednocześnie.

Tabela 1. Mała baza obiektów.

ob	opis	ab	cd	ef	gt	uv	wz
$x_1$	acdf	10	11	01	00	00	00
$x_2$	adec	10	11	10	00	00	00
$x_3$	awtv	10	00	00	10	01	10
$x_4$	befg	01	00	11	10	00	00
$x_5$	btwz	01	00	00	01	00	11
$x_6$	bvzu	01	00	00	00	11	01

Tabela 4. Lista reguł decyzyjnych

atrybut	grupa
C	2
A	1
V	1
F	2
W	1

Tabela 2. Jakość grupowania dla różnych algorytmów grupowania

funkcja odległości	entropia	czas obliczeń
deglomeracyjny	0,4815	00:00:00
aglomeracyjny	0,3312	00:00:00
proste szukanie	0,3312	00:00:00
proste Decision List	0,3312	00:04:46

## 9.2 otrzymane rezultaty

Wyniki przedstawiono w tabeli 2. Dokonano grupowania następującymi metodami:

1. hierarchiczna aglomeracyjna,
2. pełny przegląd dla każdej kombinacji podziału obliczana jest funkcja jakości i wybierany jest najlepszy podział,
3. hierarchiczna deglomeracyjna
4. oparta o listy decyzyjne

Z uwagi na niewielką liczbę obiektów stosowane algorytmy są w swej najprostszej postaci.

Otrzymane wyniki świadczą że algorytm oparty o listy decyzyjne stanowią alternatywną metodę grupowania. Jego podstawową zaletą jest brak konieczności określania miary podobieństwa dla obiektów i grup. Wyniki świadczą, że otrzymywane podziały mogą być lepsze od metody deglomeracyjnej. Można również spróbować utworzyć taką bazę danych dla której również algorytm aglomeracyjny będzie gorszy (testy dla danych z rzeczywistych baz na to wskazują). Podstawową wadą jest stosunkowo długi czas obliczeń, jest to cel dalszych badań.

## 10 Wnioski

Grupowanie za pośrednictwem atrybutów wnosi znacząco inną jakość w zakresie interpretacji i dalszej analizy wyników grupowania. W miejsce długiej tablicy przypisań obiektów (o długości równej ilości obiektów) otrzymuje się stosunkowo krótką listę reguł. Samo zmniejszenie zapisu wyniku ma

Tabela 3. Uzyskane podziały

obiekt	degl	aglom	przełg	DL
a c d f	1	2	2	2
a d e c	1	2	2	2
a w t v	1	1	1	1
b e f g	2	2	2	2
b t z w	1	1	1	1
b v z u	1	1	1	1

pozytywny wpływ na możliwość dalszego przyswojenia wyniku przez człowieka. Dodatkowo lista ta jest w określony sposób uporządkowana, co umożliwia nadanie poszczególnym regułom pewnych cech ważności. Właściwość tą można znakomicie wykorzystać do porównywania rezultatów grupowania niewiele się od siebie różniących zbiorów. Przykładem takiego zagadnienia jest porównywanie tego samego zjawiska ale w różnych momentach czasu. Zbiory obiektów reprezentujące dane zjawisko podlegają zmianom i modyfikacjom w miarę upływu czasu. Do zagadnień eksploracji danych należy poznanie i wyjaśnienie istoty tych zmian. Jedną z metod jest analiza różnic pomiędzy stanem początkowym a końcowym danego okresu. Jeżeli do takiego zagadnienia zastosowane zostanie zaproponowane powyżej grupowanie analiza otrzymanych list reguł z uwagi na względne położenie reguł da możliwość interpretacji i wyjaśniania przy pomocy pojęć w miejsce wyjaśniania poprzez przykłady.

Dalsze badania winny być prowadzone w zakresie polepszenia szybkości algorytmu, gdyż jest to słaby punkt tej metody.

## Literatura

- [1] R. Agrawal, T. Imielinski, and A. N. Swami, *Mining association rules between sets of items in large databases*, Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data (Washington, D.C.) (Peter Buneman and Sushil Jajodia, eds.), 26–28 1993, pp. 207–216.
- [2] R. Agrawal, H. Mannila, R. Srikant, H. Toivonen, and A.I. Verkamo, *Fast discovery of association rules*, Advances in Knowledge Discovery and Data Mining (1996).
- [3] G. H. Ball and D. J. Hall, *A clustering technique for summarizing multivariate data*, Behavioral Science **12** (1967), 153–155.
- [4] J.C. Bezdek, *Pattern recognition with fuzzy objective algorithms*, Plenum Press (1981).
- [5] Sergey Brin, Rajeev Motwani, Jeffrey D. Ullman, and Shalom Tsur, *Dynamic itemset counting and implication rules for market basket data*, SIGMOD 1997, Proceedings ACM SIGMOD International Conference on Management of Data, May 13-15, 1997, Tucson, Arizona, USA (Joan Peckham, ed.), ACM Press, 05.
- [6] Ming-Syan Chen, Jiawei Han, and Philip S. Yu, *Data mining: an overview from a database perspective*, IEEE Trans. On Knowledge And Data Engineering **8** (1996), 866–883.
- [7] P. Cichosz, *Systemy uczące się*, WNT, Warszawa, 2000.

- [8] Peter Clark and Tim Niblett, *The cn2 induction algorithm*, Machine Learning **3** (1989), 261–283.
- [9] Gautam Das, Heikki Mannila, and Pirjo Ronkainen, *Similarity of attributes by external probes*, Knowledge Discovery and Data Mining, 1998, pp. 23–29.
- [10] R. O. Duda and P. E. Hart, *Pattern classification and scene analysis*, Wiley, New York, 1973.
- [11] Doug Fisher, *Iterative optimization and simplification of hierarchical clusterings*, Journal of Artificial Intelligence Research **4** (1996), 147–180.
- [12] Venkatesh Ganti, Johannes Gehrke, and Raghu Ramakrishnan, *CACTUS - clustering categorical data using summaries*, Knowledge Discovery and Data Mining, 1999, pp. 73–83.
- [13] J. Grabmeier and A. Rudolph, *Techniques of cluster algorithms in data mining.*, Data Mining and Knowledge Discovery **6** (2002), no. 4, 303–360.
- [14] S. Guha, R. Rastogi, and K. Shim, *Clustering algorithm for categorical attributes*, Tech. report, Bell Laboratories, Murray Hill, 1997.
- [15] Sudipto Guha, Rajeev Rastogi, and Kyuseok Shim, *ROCK: A robust clustering algorithm for categorical attributes*, Information Systems **25** (2000), no. 5, 345–366.
- [16] Eui-Hong Han, George Karypis, Vipin Kumar, and Bamshad Mobasher, *Clustering based on association rule hypergraphs*, Research Issues on Data Mining and Knowledge Discovery, 1997.
- [17] A.K. Jain and R.C. Dubes, *Algorithms for clustering data*, Prentice Hall, New Jersey, 1988.
- [18] R. Krishnapuram and J. Keller, *A possibilistic approach to clustering*, IEEE Transactions on Fuzzy Systems **1** (1993), no. 2, 98–110.
- [19] R. T. Ng and J. Han, *Efficient and effective clustering methods for spatial data mining*, 20th International Conference on Very Large Data Bases, September 12–15, 1994, Santiago, Chile proceedings (Los Altos, CA 94022, USA) (Jorgeesh Bocca, Matthias Jarke, and Carlo Zaniolo, eds.), Morgan Kaufmann Publishers, 1994, pp. 144–155.
- [20] Ronald L. Rivest, *Learning decision lists*, Machine Learning **2** (1987), no. 3, 229–246.
- [21] Enrique H. Ruspini, *A new approach to clustering*, Information and Control **15** (1969), no. 1, 22–32.