Metody analizy danych

Proces odkrywania wiedzy

Przetwarzanie wstępne

Grupowania

Plan wykładu

- Proces odkrywania wiedzy
- Przetwarzanie wstępne
  - Dyskretyzacja
  - Niekompletność danych
- Grupowania
  - Przykład
  - Jakość grupowania
  - Miary podobieństwa
  - Metody grupowania

Proces odkrywania wiedzy

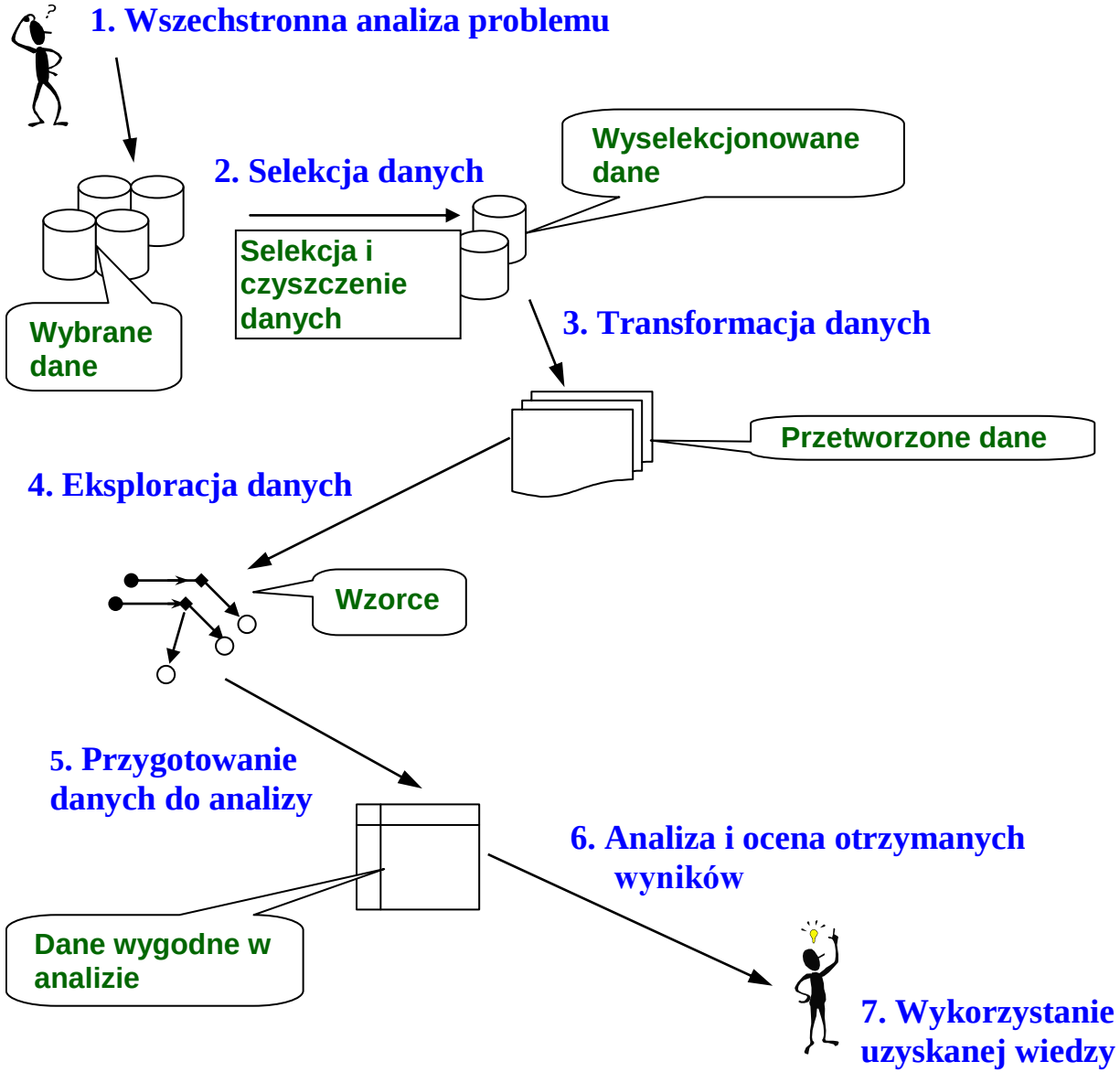
Proces odkrywania wiedzy w bazach danych (ang. *knowledge discovery in data bases*) - nietrywialny proces odkrywania obowiązujących, dotychczas nieznanych, potencjalnie użytecznych i zrozumiałych wzorców w zbiorach danych.

Odkrywanie wiedzy integruje wiele dyscyplin takich jak: statystyka, systemy baz danych, sztuczna inteligencja, optymalizacja, obliczenia równoległe. Głównym etapem tego procesu jest eksploracja danych.

Eksploracja danych

Eksploracja danych (ang. *data mining*) - proces automatycznego i efektywnego wykrywania zależności w zbiorach danych. Choć jest ona najistotniejszym etapem w procesie odkrywania wiedzy, to przeciętnie zabiera tylko od 15% do 25% procent czasu w wykonaniu tego procesu.

Etapy procesu wykrywania wiedzy



Etapy procesu odkrywania wiedzy (1)

1. **Wszechstronna analiza problemu** - poznanie i zrozumienie jego natury oraz zdefiniowanie celu procesu eksploracji danych.
2. **Selekcja danych**
   - Utworzenie bądź wyselekcjonowanie odpowiednich zbiorów danych oraz istotnych tabel, rekordów, atrybutów itd.;
   - Zespolenie danych oraz sprawdzenie poprawności danych;
   - Czyszczenie danych np. z zakłóceń lub tzw. outliers – obiektów leżących daleko od obszaru występowania zdecydowanej większości swojej klasy obiektów.

Etapy procesu odkrywania wiedzy (2)

- **Transformacja danych**
  - Wybór strategii wobec brakujących danych bądź ich wartości. Np. podjęcie decyzji dotyczącej uwzględnienia bądź nieuwzględnienia w dalszej analizie rekordów z brakującymi wartościami poszczególnych pól;
  - Konwersja typów danych;
  - Przekształcenie danych do odpowiedniej postaci np. dyskretyzacja wartości ciągłych, zamiana formy reprezentacji z relacyjnej na transakcyjną.

Etapy procesu odkrywania wiedzy (3)

4. **Eksploracja danych**
   - Wybór odpowiednich narzędzi, algorytmów oraz zestawu parametrów wejściowych i ich wartości.
   - Wykonywanie procesu eksploracji danych.
5. **Przygotowanie uzyskanych wyników do analizy** - wybór formy prezentacji np. drzewa decyzyjne oraz wizualizacja.
6. **Analiza i ocena otrzymanych wyników** - pozyskanie nowej wiedzy; ewentualnie decyzja o zmianie parametrów procesu wykrywania wiedzy i powtórzeniu całego procesu lub wybranych etapów.
7. **Zastosowanie zdobytej wiedzy w praktyce.**

Transformacje danych - dyskretyzacja

- **Dyskretyzacja atrybutów ciągłych**
  - przez eksperta
  - na dwie wartości na podstawie średniej
  - na 4 wartości na podstawie średniej i odchylenia standardowego
  - Na n przedziałów o równym rozmiarze
  - Na n równolicznych przedziałów
  - …

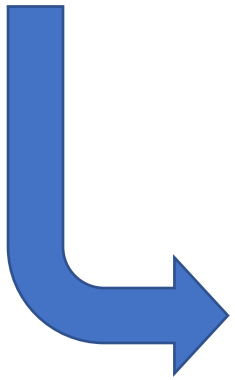
Transformacje danych – atrybuty nominalne

Zastąpienie atrybutów o wartościach nominalnych atrybutami o wartościach liczbowych – dla każdej wartości tworzony jest atrybut o wartościach binarnych: przyjmuje wartość 1, gdy w oryginalnych danych występuje dana wartość, w przeciwnym przypadku przyjmuje wartość 0.

Transformacje danych – atrybuty nominalne

Dane z atrybutem nominalnym (kolor)

ID	Kolor	Rozmiar
1	Biały	122
2	Czerwony	128
3	Zielony	122



Transformacja atrybutu nominalnego na n-atrybutów binarnych

ID	K_biały	K_czerowny	K_zielony	Rozmiar
1	1	0	0	122
2	0	1	0	128
3	0	0	1	122

Transformacje danych
- redukcja liczby atrybutów

Liniowa analiza dyskryminacyjna(ang. LDA - *Linear discriminant analysis*) – wybór najbardziej istotnych atrybutów z atrybutów dostępnych.

Analiza komponentów głównych (ang. PCA – *Principal component analysis*) – utworzenie nowych atrybutów lepiej wyjaśniających zmienność.

Niepełność danych

- **Niepewność** — wynika z subiektywnych błędów i niewystarczających informacji, dotyczących danego problemu.
- **Niedokładność** — jest powodowana przez zastosowanie nieodpowiedniego poziomu precyzyjności; nie obejmuje sytuacji, w których do określenia wartości atrybutu użyto pojęć rozmytych.
- **Niekompletność** — brak potrzebnych wartości.
- **Niespójność** — pojawia się, gdy powtarzające się opisy tych samych obiektów nie zawierają spójnych danych.

Niekompletność

Wartości brakujące

- Chwilowa niedostępność danych — brakujące dane mogą być łatwo uzupełnione z innych źródeł (np. kod pocztowy ze stron poczty).
- Brak danych spowodowany ogólną niedoskonałością metod i urządzeń, służących do ich zbierania i zapisywania — takich danych zazwyczaj nie da się w łatwy sposób uzupełnić.

Wartości niedostępne – w danym typie obiektów pojawiają się instancje, do których nie mają zastosowania pewne fragmenty opisu tego typu.

Niekompletność –metody uzupełnienia danych

Pominięcie obiektów

Proste uzupełnianie danych

- wartością specjalną,
- dominantą,
- średnią

Użycie klasyfikatora:

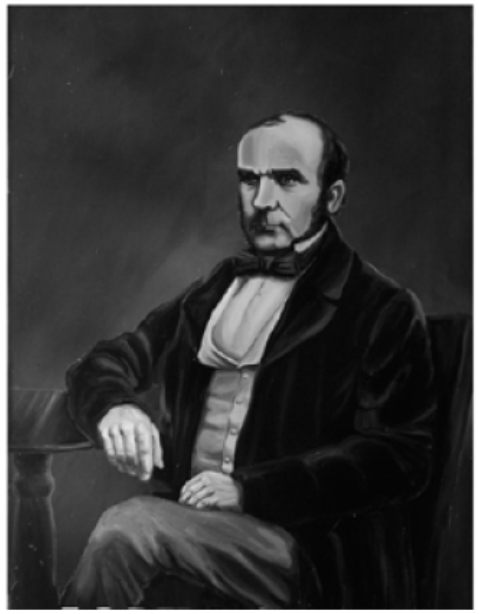
- kNN,
- drzewa decyzyjne,
- teorii zbiorów przybliżonych

Statystyczne

Grupowanie

Historyczny przykład grupowania

- 1848: W Londynie panuje epidemia ‚Azjatyckiej cholery'.
- John Snow zaobserwował rozkład śmiertelnych przypadków choroby w mieście i
- wysnuł hipotezę, że woda z rzeki zanieczyszczona ściekami od osób zakażonych chorobą wyjaśnia przestrzenne zróżnicowanie śmiertelności na terenie Londynu.



John Snow

- 1848: An epidem
  'Asiatic cholera' h
  London
- John Snow obser
  distribution of dea
  throughout the cit
- hypothesized tha
  water contaminat
  cholera evacuatic
  explained spatial
  variations in mort
  throughout Londc

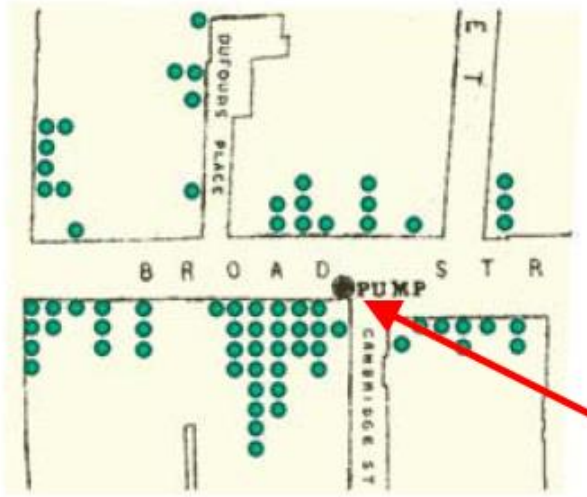
Historyczny przykład grupowania

- Sierpień 1854: epidemia cholery dociera do obszaru Londynu Północnego.
- J. Snow pozyskał z Urzędu Stanu Cywilnego nazwiska i adresy wyszczególnione na 83 aktach zgonu.

Na mapie zaznaczył przypadki cholery.

Historyczny przykład grupowania

- Zebrał też informacje o potencjalnych źródłach zanieczyszczenia (pompach)
- i połączył te informacje na mapie.



Zaobserwował, że prawie wszystkie przypadki śmiertelne miały miejsce w niewielkiej odległości od pompy przy Broad Street

Historyczny przykład grupowania

- Snow przekonał radę parafialną, aby zamknąć pompę.
- Trudna decyzja: woda dostarczana przez tą pompę była wysoko ceniona przez mieszkańców; ludzie z sąsiednich ulic przychodzili tu po wodę.
- Efekt: epidemia ustała.

Koniec?

- Rada nie uwierzyła analizom Snow-a, więc wikary powtórzył jego badanie i wziął pod uwagę inne czynniki (czystość/brud domów).
- Wikary, który na początku miał wątpliwości co do teorii Snow-a zlokalizował 700 przypadków śmiertelnych w promieniu 230 metrów i pokazał, że korzystanie z pompy przy Broad Street było mocno skorelowane ze śmiertelnymi przypadkami azjatyckiej cholery.

Czym jest grupowanie?

- Grupa: kolekcja obiektów
  - podobne do siebie w ramach tej samej grupy
  - różne od obiektów w innych grupach
- Analiza grup
  - Znajdowanie podobieństw w danych na podstawie charakterystyki znalezionej w danych i łączenie podobnych obiektów w grupy
- Uczenie bez nadzoru: brak predefiniowanych klas
- Typowe zastosowania
  - Jako wydzielone narzędzie umożliwiające wgląd w rozkład danych
  - Jako etap przetwarzania przed innymi algorytmami

Grupowanie

Celem grupowania jest podział zbioru obiektów na klasy (grupy) podobnych obiektów (mających podobne wartości atrybutów).

- W zależności od metody grupowania liczba grup jest albo nie jest określona jako parametr wejściowy.
- Cechą dobrego grupowania jest wysokie podobieństwo obiektów w ramach tej samej grupy oraz niskie podobieństwo obiektów z różnych grup.
- Podobieństwo często jest określane jako pewna miara odległości między dwoma obiektami.

Przykłady zastosowań grupowania

- ***Marketing*:** pomaga sprzedawcom wykryć wyraźne grupy w bazach danych klientów a następnie wykorzystać tą wiedzę do stworzenia programów marketingu celowego
- ***Ubezpieczenia*:** identyfikacja grup posiadaczy ubezpieczenia motoryzacyjnego o przeciętnie wysokim poziomie odszkodowań
- ***Planowanie miejskie*:** identyfikacja grup domów w zależności od rodzaju domu, wartości i lokalizacji geograficznej

Wymagania stawiane grupowaniu w eksploracji danych

- Skalowalność
- Możliwość przetwarzania różnych typów atrybutów
- Możliwość uwzględniania danych dynamicznych
- Wykrywanie grup o dowolnych kształtach
- Minimalne wymagania dotyczące znajomości domeny w celu ustalenia parametrów wejściowych
- Możliwość pracy z danymi zaszumionymi i obiektami niedopasowanymi
- Nieczułość na kolejność rekordów wejściowych
- Uwzględnianie ograniczeń podanych przez użytkownika
- Interpretowalność i użyteczność

Grupowanie – wyznaczanie podobieństwa obiektów

Podobieństwo obiektów – atrybuty liczbowe

- d_i, d_j - reprezentacja obiektu w przestrzeni V,
- $w(d_l,t_p)$ - waga atrybutu t_p w obiekcie d_l.

- Manhattan

$$mn(d_i,d_j) = \sum_{k\in 1..|V|} | w(d_i,t_k) - w(d_j,t_k) |$$

- Euklidesowa

$$eu(d_i,d_j) = \sqrt{\sum_{k\in 1..|V|} \left[w(d_i,t_k) - w(d_j,t_k)\right]^2}$$

- Kosinusowa

$$\cos(d_i,d_j) = \frac{< d_i,d_j >}{| d_i || d_j |} = \frac{\sum_{k=1}^{|V|} w(d_i,t_k) * w(d_j,t_k)}{\sqrt{\sum_{k=1}^{|V|} w(d_i,t_k)^2} \sqrt{\sum_{k=1}^{|V|} w(d_j,t_k)^2}}$$

Podobieństwo obiektów – atrybuty binarne

- d_i, d_j - reprezentacja obiektu w przestrzeni V,
- $w(d_l,t_p)$ - waga atrybutu t_p w obiekcie d_l.

- Dice's

D(A,B) = |A∩B| / |A|+|B|

$$dice(d_i,d_j) == \frac{2\sum_{k=1}^{|V|} w(d_i,t_k) * w(d_j,t_k)}{\sum_{k=1}^{|V|} w(d_i,t_k) + \sum_{k=1}^{|V|} w(d_j,t_k)}$$

- Jaccard's

J(A,B) = |A∩B| / |A∪B|

$$jacc(d_i,d_j) == \frac{\sum_{k=1}^{|V|} w(d_i,t_k) * w(d_j,t_k)}{\sum_{k=1}^{|V|} w(d_i,t_k) + \sum_{k=1}^{|V|} w(d_j,t_k) - \sum_{k=1}^{|V|} w(d_i,t_k) w(d_j,t_k)}$$

Podobieństwo obiektów – różne atrybuty

- d_i, d_j - reprezentacja obiektu w przestrzeni V,
- $w(d_l,t_p)$ - waga atrybutu t_p w obiekcie d_l.

Niezgodność procentowa

$$proc\_dif(d_i, d_j) = \frac{\sum_{k=1}^{|V|} \left[\left(w(d_i, t_k) \neq w(d_j, t_k) \right) \ ? \ 1 \ : \ 0 \right]}{|V|}$$

Podobieństwo obiektów – różne atrybuty

- d_i, d_j - reprezentacja obiektu w przestrzeni V,s
- $w(d_l,t_p)$ - waga atrybutu t_p w obiekcie d_l.

Podobieństwo Gower'a

$$gow(d_i, d_j) = \frac{\sum_{k=1}^{|V|} d_k S_k\left(w(d_i,t_k), w(d_j,t_k)\right)}{\sum_{k=1}^{|V|} d_k}$$

Dla atrybutów liczbowych:

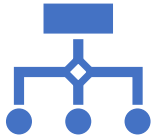
$$S_k(d_i, d_j) = 1 - \frac{|(w(d_i,t_k) - w(d_j,t_k)|}{r_k}$$

gdzie: r_k- różnica między wartością maksymalną a wartością minimalną dla atrybutu k

Dla atrybutów nominalnych:

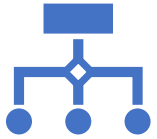
$$S_k(d_i, d_j) = [w(d_i, t_k) \neq w(d_j, t_k)\ ?\ 0\ :\ 1]$$

Grupowanie - algorytmy



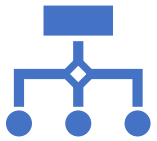
Podejścia algorytmiczne w grupowaniu

- Hierarchiczne
  - wszystkie punkty w jednej grupie
  - następnie podziały i/lub łączenia do osiągnięcia warunku stopu
  - typowe metody: BIRCH, CURE, ROCK
- Partycjonujące
  - rozpocznij od losowo wybranego punktu centralnego
  - przypisz punkty do najbliższego punktu centralnego
  - uaktualnij punkty centralne
  - typowe metody: k-środków, k-średnich, CLARANS



Podejścia algorytmiczne w grupowaniu

- Gęstościowe
  - znajdź grupy na podstawie gęstości regionów
  - typowe metody: DBSCAN, OPTICS, DenClue
- Oparte na siatce
  - podziel przestrzeń grupowania na skończoną liczbę komórek
  - użyj progów aby wybrać komórki z wysoką gęstością
  - połącz sąsiednie komórki w celu stworzenia grup
  - typowe metody: STING, BANG



Algorytmy partycjonujące: podstawowe koncepcje

- Metoda partycjonowania: Stwórz podział bazy danych *D* złożonej z *n* obiektów na *k* grup, min. sumę odległości w kwadracie

$$\sum_{i=1}^{k} \sum_{p \in C_i} dist(p, c_i)^2$$

- Mając dane *k*, znajdź taki podział na *k* grup, który optymalizuje wybrane kryterium podziału
  - metody heurystyczne: algorytmy *k-średnich* i *k-środków*
    - *k-średnich* (MacQueen'67): każda grupa jest reprezentowana przez środek grupy
    - *k-środków* lub PAM (Partition around medoids) (Kaufman & Rousseeuw'87): każda grupa jest reprezentowana przez jeden z obiektów w grupie

Algorytm k-średnich (1)

Cel: Znaleźć k środków tak, aby suma odległości punktów do najbliższego centroida była minimalna.

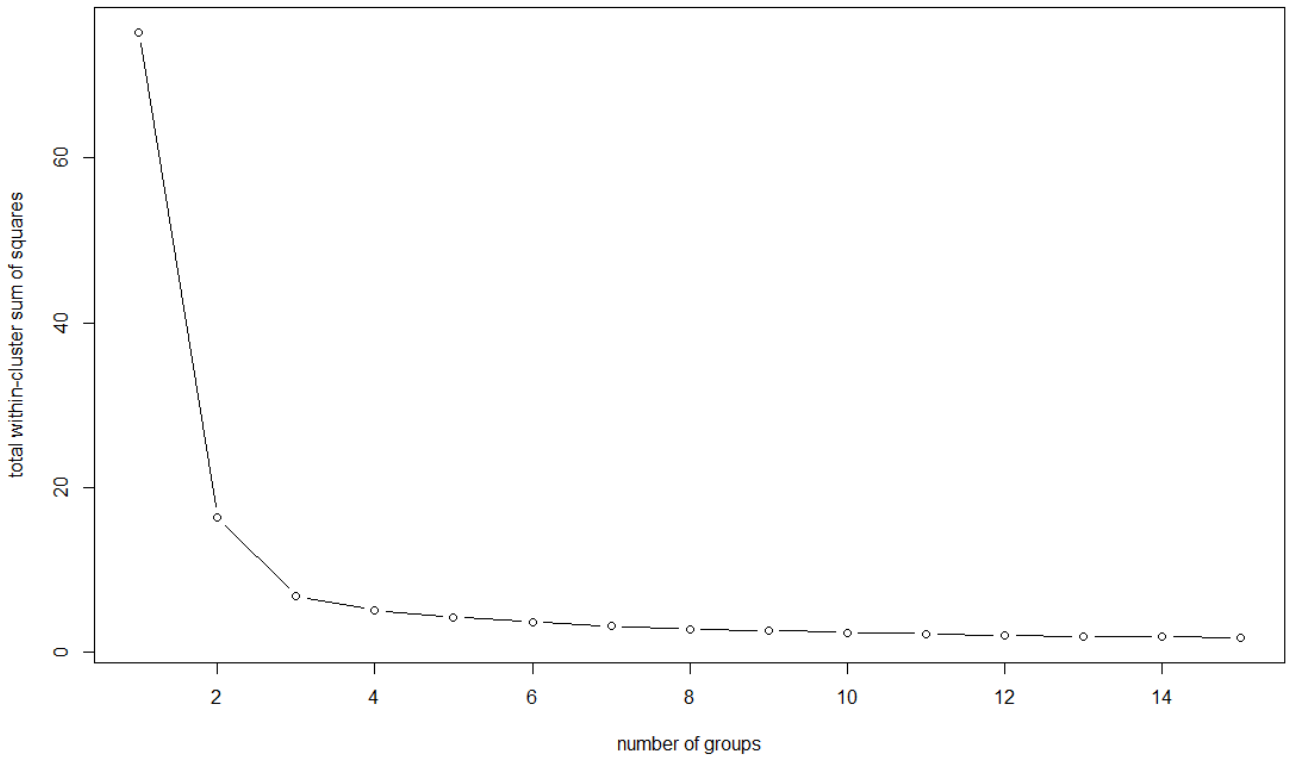
Krok 1. Wybierz losowo dowolnych k centrum klastrów (centroidów)
Krok 2. Przydziel każdy obiekt do najbliższego centroida.
Krok 3. Wyznacz nowy układ centroidów
Krok 4. Powtarzaj Krok 2 i Krok 3, aż:

- układ centroidów się nie zmienił
- brak wystarczającej poprawy jakości grupowania
- osiągnięto maksymalną dopuszczalną liczbę iteracji.

Algorytm k-średnich (2)

- Jakości klastrów zależą od wyboru początkowego układu centroidów.
- Algorytm może trafić w lokalne minimum
- Aby unikać lokalne minimum: startować z różnymi układami losowo wybieranych centroidów

Wyznaczenie liczby grup – metoda łokcia



Algorytm k-średnich (3)

Zalety

- niska złożoność, a co za tym idzie wysoka wydajność działania
- przy dużych zbiorach i niskich liczbach grupa algorytm ten będzie zdecydowanie szybszy niż pozostałe algorytmy tej klasy
- Tworzy grupy sferyczne

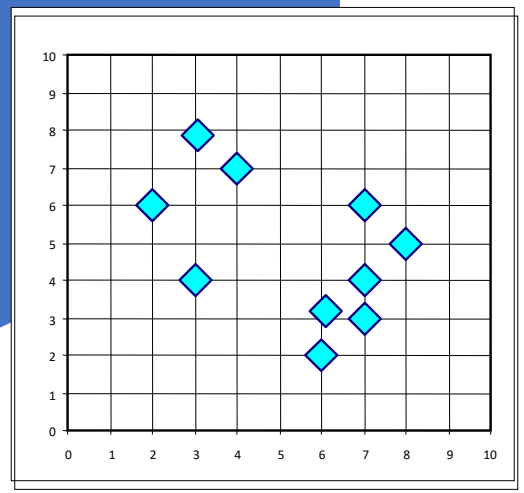
Wady:

- nie pomaga w określeniu liczby grup (K)
- różne wartości początkowe prowadzą do różnych wyników
- działa dobrze tylko dla „sferycznych” skupisk o jednorodnej gęstości
- Wszystkie przykłady są przydzielone do skupień
- Problem z tzw. outliers (duża wrażliwość)

Algorytm k-środków

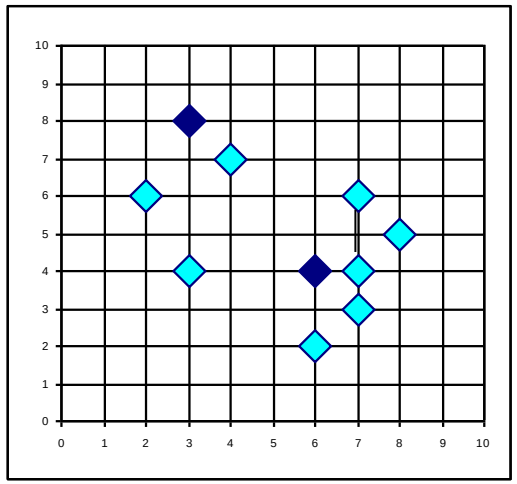
- Znajdź obiekty reprezentatywne w grupach, zwane środkami
- PAM (Partitioning Around Medoids, 1987)
  - rozpoczyna od początkowego zbioru środków i iteracyjnie zamienia jeden ze środków na jeden element nie będący środkiem, jeżeli poprawia to całkowitą odległość wynikowej grupy
  - PAM działa efektywnie dla małych zbiorów danych i nie skaluje się dobrze dla dużych zbiorów
- *CLARA* (Kaufmann & Rousseeuw, 1990)
- *CLARANS* (Ng & Han, 1994): Randomized sampling

Algorytm k-środków (PAM)



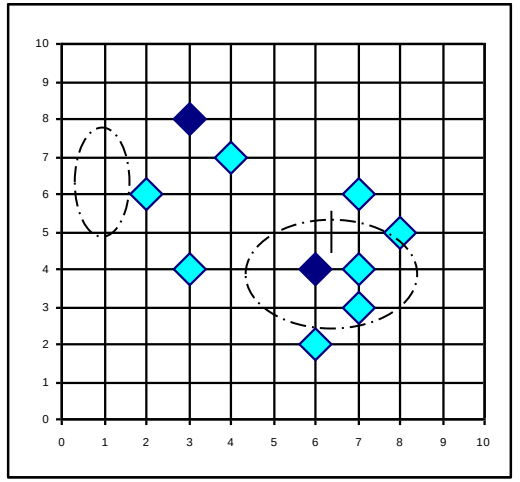
K=2

Dowolnie wybierz k obiektów jako pocz. środki

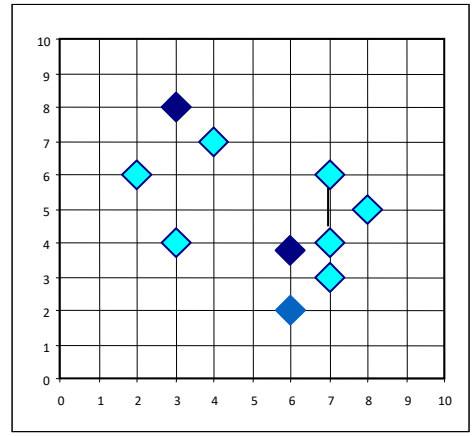


Przypisz każdy pozostały obiekt do najbliższego środka

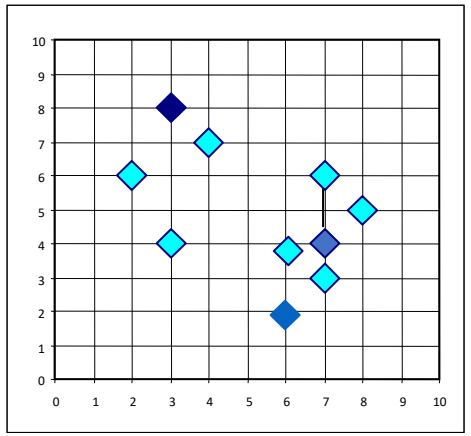
Całkowity koszt = 20



Losowo wybierz obiekt nie będ. środkiem, O_{ramdom}



Oblicz całkowity koszt zamiany



Całkowity koszt = 26

Zamiana O i O_{ramdom}

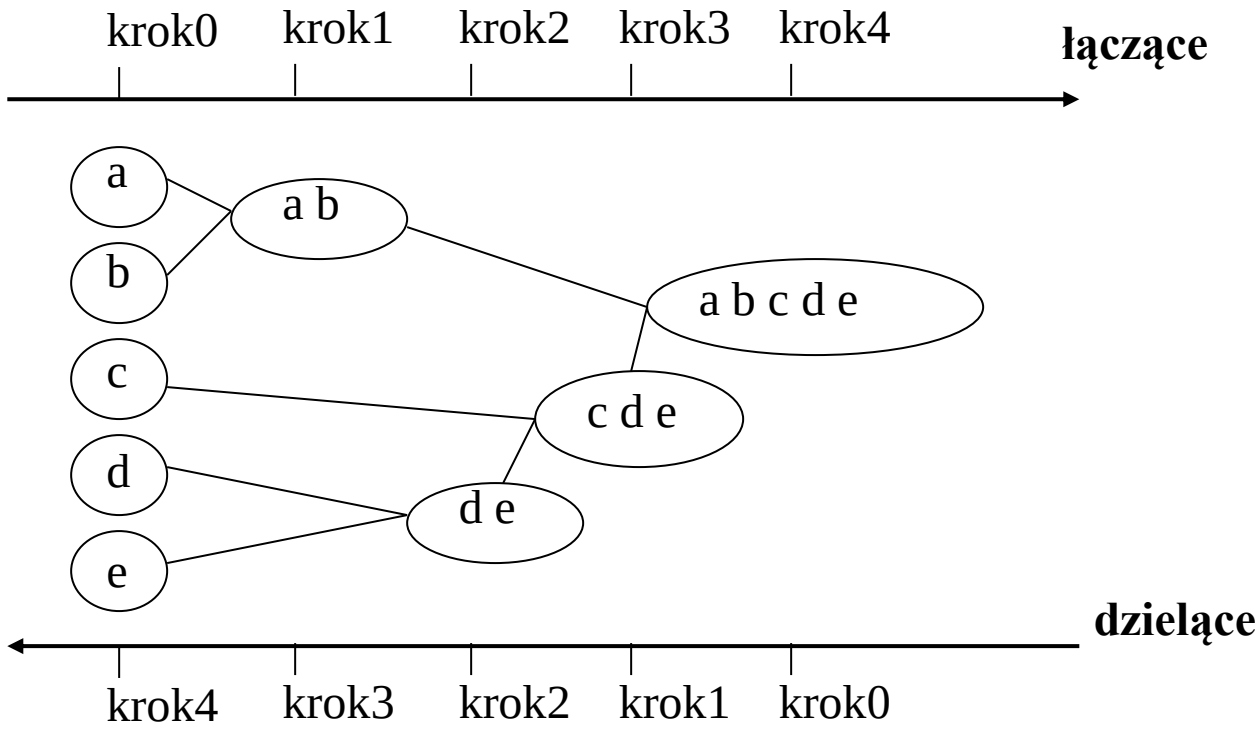
Jeżeli poprawa jakości

Do loop

Until brak zmian

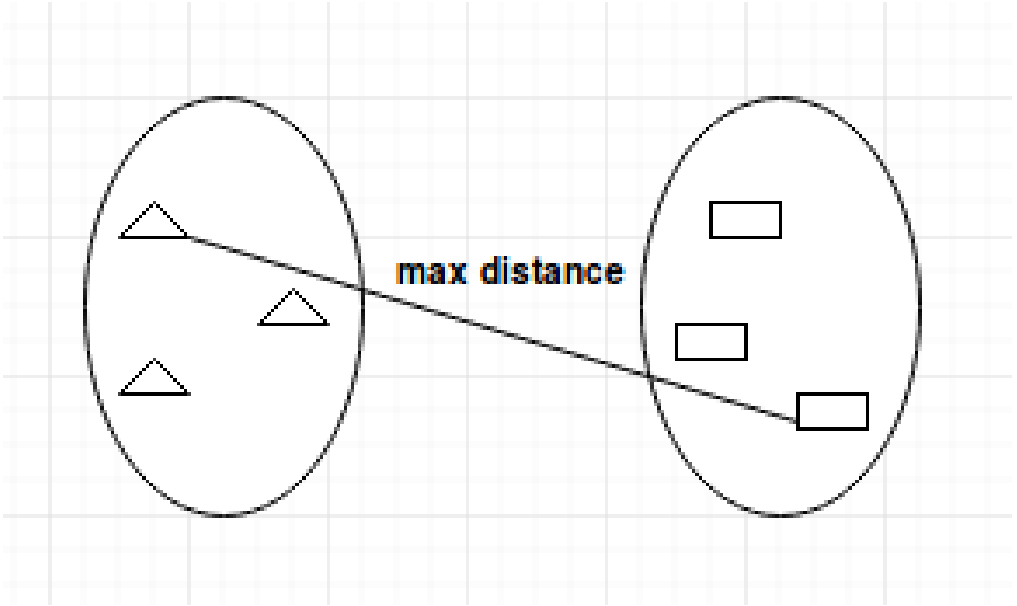
Etapy procesu wykrywania wiedzy

- Łączące (większość metod należy do tej kategorii)
- Dzielące (kończą działanie po napotkaniu kryt. stopu, np. ustalona liczba grup, osiągnięto ustaloną średnicę grupy)



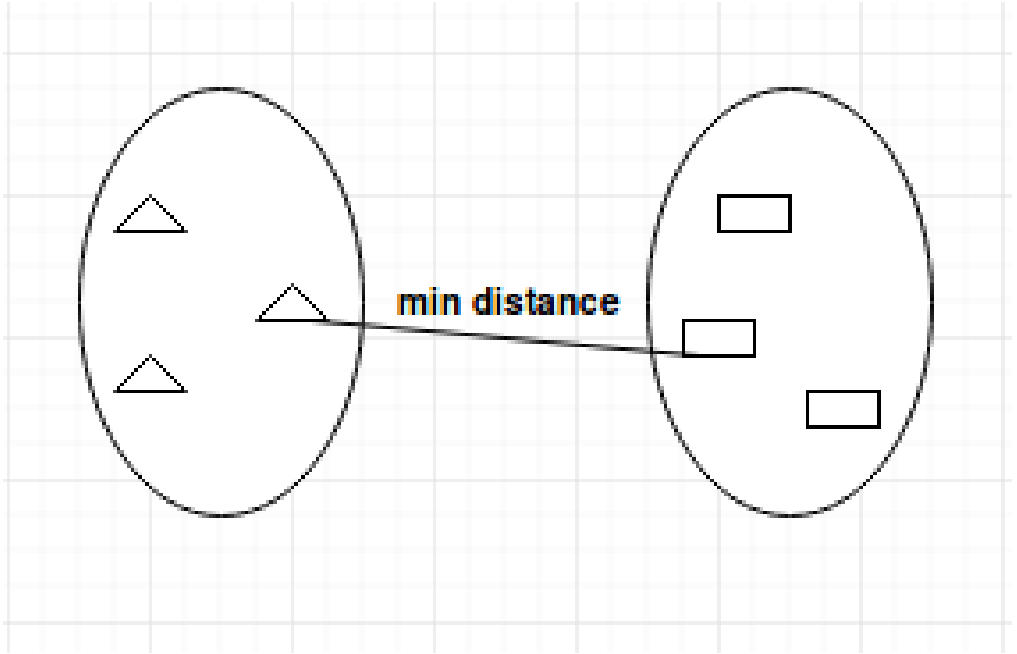
Grupowanie hierarchiczne: metody łączenia grup

- Maksymalna (kompletna): odległość między dwoma grupami to maksymalna wartość wszystkich par odległości między elementami w grupie 1 i w grupie 2. Tworzy grupy bardziej kompaktowe.



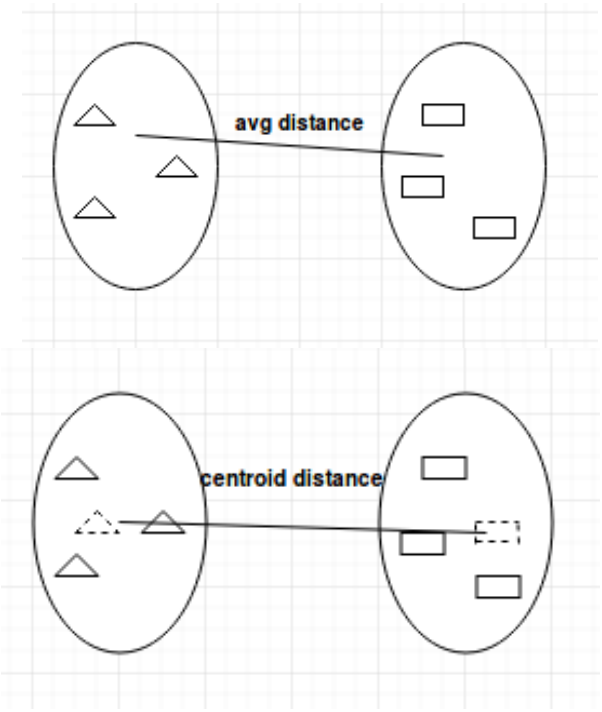
Grupowanie hierarchiczne: metody łączenia grup

- Minimalna (pojedyncza): odległość między dwoma grupami to minimalna wartość wszystkich par odległości między elementami w grupie 1 i w grupie 2. Tworzy grupy bardziej „luźne”.



Grupowanie hierarchiczne: metody łączenia grup

- Średnia: odległość między dwoma grupami to średnia odległość między elementami w grupie 1 i w grupie 2.
- Łączenie środków ciężkości (centroidów) grup: odległość między dwoma grupami to odległości między centroidami dla grupy 1 i dla grupy 2.



W każdym kroku grupowania łączone są dwie grupy mające najmniejszą odległość połączenia.

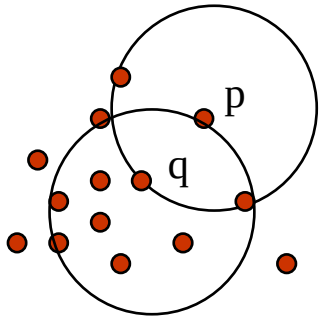
Metody grupowania hierarchicznego

- Słabe strony łączących metod grupowania
  - nie skalują się dobrze: złożoność czasowa co najmniej $O(n^2)$, gdzie n jest całkowitą liczbą obiektów
  - nie mogą wycofać poprzednich zmian
- Integracja metod hierarchicznych z grupowaniem opartym na odległości
  - BIRCH (1996): wykorzystuje CF-drzewo i inkrementacyjnie dostosowuje jakość podgrup
  - ROCK (1999): grupowanie danych kategorycznych poprzez analizę sąsiedztwa i połączenia
  - CHAMELEON (1999): grupowanie hierarchiczne z wykorzystaniem modelowania dynamicznego

Grupowanie gęstościowe: podstawowe koncepcje

- Dwa parametry*:*
  - *Eps*: Maksymalny promień sąsiedztwa
  - *MinPts*: Minimalna liczba punktów w sąsiedztwie Eps tego punktu
- $N_{Eps}(p)$: *{q należy do D | dist(p,q) <= Eps}*
- Bezpośrednio gęstościowo-osiągalny: Punkt *p* jest bezpośr. gęstościowo osiągalny z punktu *q* w odniesieniu do *Eps* i *MinPts* jeżeli
  - *p* należy do $N_{Eps}(q)$
  - warunek punktu głównego:

$$|N_{Eps}(q)| >= MinPts$$

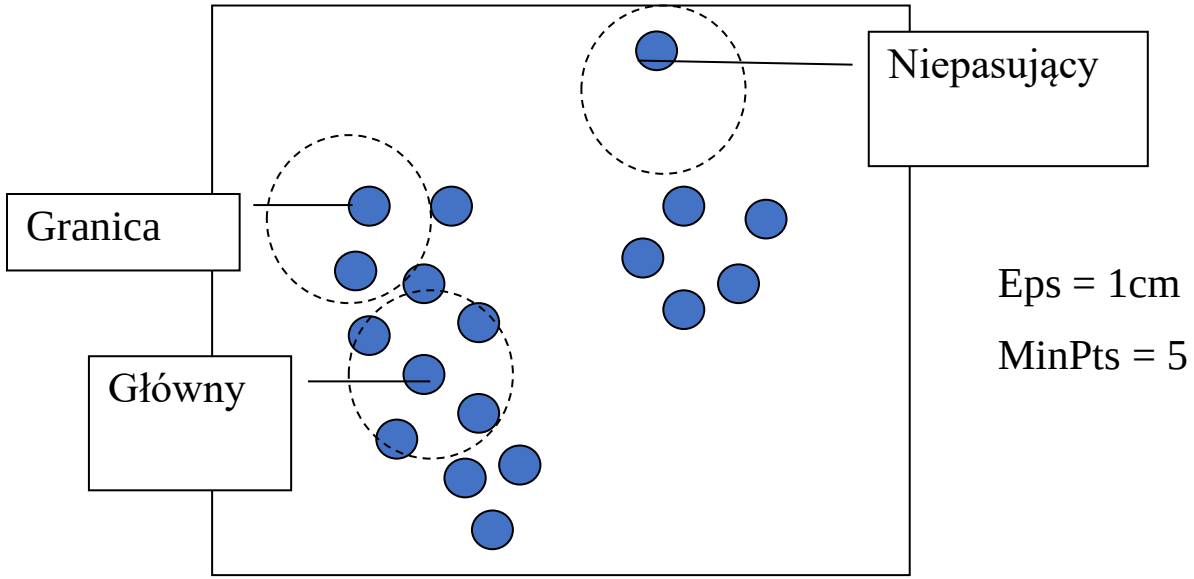


MinPts = 5

Eps = 1 cm

DBSCAN

Opiera się na pojęciu grupy opartym na gęstości: grupa jest zdefiniowana jako maksymalny zbiór gęstościowo połączonych punktów



Mierzenie jakości grupowania

- Miara podobieństwa/niepodobieństwa: typowo podobieństwo jest wyrażane za pomocą funkcji odległości, zwykle metryki: $d(i, j)$
- Istnieje oddzielna funkcja ‘jakości’, która mierzy jak ‘dobra’ jest grupa
- Definicje funkcji odległości są zwykle bardzo różne dla zmiennych interwałowych, boolowskich, kategorycznych, porządkowych i wektorowych
- Trudno jest zdefiniować ‘wystarczająco podobne’ lub ‘wystarczająco dobre’
  - odpowiedź jest zwykle wysoce subiektywna

Ocena jakości grupowania

- Metody wymagające grupowania wzorcowego
  - porównanie uzyskanego grupowania z grupowaniem wzorcowym
- Metody niewymagające grupowania wzorcowego
  - wyznaczenia jakości uzyskanych grup

Ocena jakości grupowania (1)

Podobieństwo grup dla grupowania C

$$G_{sim}(C) = \frac{\overline{d(p,q) : G \in C, p,q \in G, q \neq p}}{\overline{d(p,q) : G \in C, H \in C \setminus G, p \in G, q \in H}}$$

Jakość liczby grup dla grupowania C

$$GQ(C) = \frac{|C|}{|p : p \in G, G \in C|}$$

Jakość grupowania dla grupowania C

$$CQ(C) = G_{sim}(C) + GQ(C)$$

Ocena jakości grupowania (2)

Rand index

W – grupowanie wzorcowe, G - grupowanie oceniane

A – liczba par obiektów należących do tej samej grupy w grupowaniu W i G

B – liczba par obiektów należących do różnych grup w grupowaniu W i G

a – liczba par obiektów należących do tej samej grupy w grupowaniu W, ale należących do różnych grup w grupowaniu G

b – liczba par obiektów należących do różnych grup w grupowaniu W, ale do tych samych grup w grupowaniu G

$$R = \frac{A+B}{A+B+a+b} = \frac{A+B}{n(n-1)/2}$$

Homogeniczność (ang. *homogeneity*)

$$\text{homogeneity } h = \begin{cases} 1 & \text{gdy } \mathrm{H}(C|K) = 0 \\ 1 - \dfrac{H(C|K)}{H(C)} & \text{w przeciwnym przypadku} \end{cases}$$

gdzie:

n – liczba obiektów

C = {c_i |1, …, n} – zbiór klas

K= {k_i |1, …, m} – zbiór grup

a_{ij} - liczba obiektów należących do klasy c_i i grupy k_j

$$H(C \mid K) = -\sum_{k=1}^{|K|} \sum_{c=1}^{|C|} \frac{a_{ck}}{n} \log \frac{a_{ck}}{\sum_{c=1}^{C} a_{ck}}$$

$$H(C) = -\sum_{c=1}^{|C|} \frac{\sum_{k=1}^{|K|} a_{ck}}{n} \log \frac{\sum_{k=1}^{|K|} a_{ck}}{n}$$

Kompletność (ang. *completeness*)

$$\text{completeness } c = \begin{cases} 1 & \text{gdy } H(K|C) = 0 \\ 1 - \dfrac{H(K|C)}{H(K)} & \text{w przeciwnym przypadku} \end{cases}$$

gdzie:

n – liczba obiektów

C = {c_i |1, …, n} – zbiór klas

K= {k_i |1, …, m} – zbiór grup

a_{ij} - liczba obiektów należących do klasy c_i i grupy k_j

$$H(K \mid C) = -\sum_{c=1}^{|C|} \sum_{k=1}^{|K|} \frac{a_{ck}}{n} \log \frac{a_{ck}}{\sum_{k=1}^{K} a_{ck}}$$

$$H(K) = -\sum_{k=1}^{|K|} \frac{\sum_{c=1}^{|C|} a_{ck}}{n} \log \frac{\sum_{c=1}^{|C|} a_{ck}}{n}$$

Miara oparta na entropii

$$E = \frac{2 * homogeniczność * komplentość}{homogeniczność + komplentość}$$

Ocena jakości grupowania (3)

Indeks Silhouette

$$Silhouette(x) = \frac{b(x) - a(x)}{\max(b(x), a(x))}$$

gdzie: a(x) – średnia odległość obiektu *x* do innych obiektów w swojej grupie

b(x) – minimalna odległość obiektu *x* od najbliższej grupy

Indeks przyjmuje wartości <-1, 1>, gdzie 1 oznacza, że dany obiekt jest przydzielony do najlepszej z możliwych grup, 0 – obiekt znajduje się między dwoma grupami, -1 – zły przydział obiektu.

$$GSilhouette = \frac{1}{N} \sum_{i=1}^{N} Silhouette(x_i)$$

gdzie: N – liczba obiektów w zbiorze