# Zadanie grupowania, wykł. 10

Joanna Jędrzejowicz

Instytut Informatyki

# Na czym polega grupowanie

Grupowanie (ang. clustering)- grupowanie rekordów w klasy podobnych obiektów.
Grupa (klaster) jest zbiorem rekordów, które są podobne do siebie nawzajem i niepodobne do rekordów z innych grup.
Grupowanie różni się od klasyfikacji tym, że w przypadku grupowania nie ma zmiennej celu i mamy do czynienia z uczeniem nienadzorowanym - należy znaleźć podobieństwa w danych zgodnie z ich cechami charakterystycznymi i pogrupować.

# Przykłady

- w aplikacjach medycznych - wyodrębnianie grup pacjentów z podobnymi symptomami,
- namierzanie grupy potencjalnych klientów o podobnych zachowaniach w sferze zakupów,
- podział zachowań finansowych na korzystne i niepewne,
- w biologii pomocne przy sporządzaniu taksonomii,
- redukcja wymiarów zbioru danych, gdy zbiór jest opisany przez dużą liczbę atrybutów,

Grupowanie jest często wykonywane jako krok wstępny do zgłębiania danych, z wynikowymi grupami użytymi jako dane wejściowe do innej techniki.

# Proces grupowania

- wybór reprezentacji obiektów - selekcja atrybutów i typów (kategoryczne-nominalne, liczbowe ciągłe, porządkowe itp),
- wybór miary podobieństwa/niepodobieństwa - inaczej odległosci, obiektów,
- algorytm grupowania,
- wybór reprezentacji klastrów

# Sposób przedstawienia danych do grupowania - macierz danych

Załóżmy, że danych jest $n$ obiektów (rekordów) i każdy opisany jest przez $p$ atrybutów. Na przykład $n$ osób opisanych przez *wiek*, *wzrost*, *waga* itp.

$$\begin{bmatrix} x_{11} \cdots & x_{1f} \cdots & x_{1p} \\ \vdots & \vdots & \vdots \\ x_{i1} \cdots & x_{if} \cdots & x_{ip} \\ \vdots & \vdots & \vdots \\ x_{n1} \cdots & x_{nf} \cdots & x_{np} \end{bmatrix}$$

$x_{if}$ jest wartością atrybutu $f$ dla rekordu $i$; dla $p = 2$ łatwo zwizualizować na płaszczyźnie

# Sposób przedstawienia danych do grupowania - macierz odległości

Wiele algorytmów grupowania operuje na macierzy odległości

$$\begin{bmatrix} 0 & & & & \\ d(2,1) & 0 & & & \\ d(3,1) & d(3,2) & 0 & & \\ \vdots & \vdots & \vdots & & \\ d(n,1) & d(n,2) & \cdots & \cdots & 0 \end{bmatrix}$$

Przyjmujemy, że $d(i,j)$ jest miarą różnicy między rekordem $i$ oraz $j$; ponadto:

$$d(i,j) = d(j,i),\ d(i,i) = 0$$

Jeśli dane są reprezentowane przez macierz danych, zaś algorytm grupowania korzysta z macierzy odległości, to pierwszym krokiem algorytmu grupowania jest obliczenie macierzy odległosci.

# Przypadek danych liczbowych

- możliwe określenie odległości

$$d(i,j) = \sqrt{|x_{i1} - x_{j1}|^2 + |x_{i2} - x_{j2}|^2 + \cdots |x_{ip} - x_{jp}|^2}$$

$$d(i,j) = |x_{i1} - x_{j1}| + |x_{i2} - x_{j2}| + \cdots |x_{ip} - x_{jp}|$$

$$d(i,j) = \sqrt{w_1|x_{i1} - x_{j1}|^2 + w_2|x_{i2} - x_{j2}|^2 + \cdots w_p|x_{ip} - x_{jp}|^2}$$

- własności odległości

$$d(i,j) \geq 0,\ d(i,i) = 0,\ d(i,j) = d(j,i)$$

$$d(i,j) \leq d(i,h) + d(h,j)$$

# Przypadek atrybutów binarnych

Tabela różnic: załóżmy, że porównujemy dwa wiersze z atrybutami binarnymi, q - oznacza liczbę atrybutów dla których w obu wierszach jest 1, t - w obu wierszach jest 0 itd. Liczba atrybutów jest p=q+s+r+t

| | | obiekt j | | |
|---|---|---|---|---|
| | | 1 | 0 | suma |
| obiekt i | 1 | q | r | q+r |
| | 0 | s | t | s+t |
| | suma | q+s | r+t | p |

# Atrybuty symetryczne i niesymetryczne

Atrybuty symetryczne - wartości równomiernie rozłożone.
Jeśli atrybuty są symetryczne, to przyjmuje się pierwszy wzór, dla niesymetrycznych - drugi (wzór Jacarda)

$$d(i,j) = \frac{r+s}{q+r+s+t}, \; d(i,j) = \frac{r+s}{q+r+s}$$

# Przykład

| imie | plec | gorączka | kaszel | test1 | test2 | test3 | test4 |
|---|---|---|---|---|---|---|---|
| Jacek | M | T | N | P | N | N | N |
| Maria | K | T | N | P | N | P | N |
| Jan | M | T | T | N | N | N | N |

- płeć jest atrybutem symetrycznym,
- pozostałe atrybuty są binarne niesymetryczne,
- wartości T,P ustalamy na 1, wartość N- na 0,

Korzystamy ze wzoru Jacarda, pomijając atrybut symetryczny:

$$d(Jacek, Maria) = \frac{0+1}{2+0+1} = 0.33,\ d(Jacek, Jan) = \frac{1+1}{1+1+1} = 0.67$$

$$d(Jan, Maria) = \frac{1+2}{1+1+2} = 0.75$$

Wartości dla Jana i Marii są najbardziej oddalone - największa różność

# Przypadek atrybutów nominalnych

Atrybuty nominalne są uogólnieniem binarnych, np. przyjmijmy, że atrybut *kolor* przyjmuje wartości *czerwony, żółty, zielony, różowy, niebieski*
Można przyjąć - metoda 1

$$d(i,j) = \frac{p - m}{p}$$

gdzie $p$ jest liczbą atrybutów, $m$ oznacza liczbę atrybutów dla których w wierszach $i$, $j$ są te same wartości.
Metoda 2: utworzyć nową zmienną binarną dla każdej możliwej wartości atrybutu.

# Atrybuty różnych typów, inny sposób

- metryka z formułę ważoną

$$d(i,j) = \frac{\sum_{f=1}^{p} \delta_{ij}^{(f)} d_{ij}^{(f)}}{\sum_{f=1}^{p} \delta_{ij}^{(f)}}$$

- $\delta_{ij}^{(f)} = 0$ jeśli $x_{if}$ lub $x_{jf}$ jest brakujące, albo $x_{if} = x_{jf} = 0$; w przeciwnym razie $\delta_{ij}^{(f)} = 1$
- jeśli $f$ jest binarny lub nominalny

$$d_{ij}^{(f)} = \begin{cases} 0 & \text{jesli } x_{if} = x_{jf} \\ 1 & \text{inaczej} \end{cases}$$

- jeśli $f$ jest numeryczny używamy zestandaryzowanych wartości,

# Standaryzacja danych

- dlaczego standaryzuje się dane liczbowe: przykład $x = (0.1, 20)$, $y = (0.9, 720)$ odległość euklidesowa $\sqrt{(0.9 - 0.1)^2 + (720 - 20)^2} = 700$ zdominowana przez wartość drugiego atrybutu; po zestandaryzowaniu do przedzialu $(0, 1)$ wartości drugego atrybutu zostaną sprowadzone do, odpowiednio 0.02 i 0.72 oraz odległość wyniesie 1.063,
- standaryzacja do przedziału $[0, 1]$)

$$rg(x_{if}) = \frac{x_{if} - min(f)}{max(f) - min(f)}$$

# Standaryzacja danych, cd

- standaryzacja wykorzystująca odchylenie standardowe; dla atrybutu $f$: średnia wartość

$$m_f = \frac{x_{1f} + x_{2f} + \cdots + x_{nf}}{n}$$

średnie odchylenie
$s_f^2 = \frac{1}{n-1}(|x_{1f} - m_f|^2 + |x_{2f} - m_f|^2 + \cdots + |x_{nf} - m_f|^2)$
atrybut standaryzowany (z-score)

$$z_{if} = \frac{x_{if} - m_f}{s_f}$$

# Podstawowe metody grupowania

- grupowanie przez podział
  - dokonuje się pewną liczbę podziałów i ocenia przy pomocy wybranego kryterium np. minimalizując sumę kwadratów błędów
  - przykłady: metoda k-średnich, k-medoids
- metody hierarchiczne: tworzona jest struktura poprzez rekurencyjne dzielenie lub łaczenie istniejacych grup
- metody oparte o gęstość

# Algorytm k-średnich

1. ustal wartośc k (liczbę grup),
2. losowo ustal k początkowych środków grup (centroidy),
3. dla każdego rekordu danych znajdź najbliższy centroid (nowy środek grupy)- w ten sposób wszystkie rekordy zostaną przydzielone do k grup (klastrów)
4. dla każdej z grup znajdź centroid i uaktualnij położenie środka grupy jako nową wartość centroidu
5. powtarzaj kroki 3-5 dopóki są zmiany (lub nie jest spełnione kryterium końca)

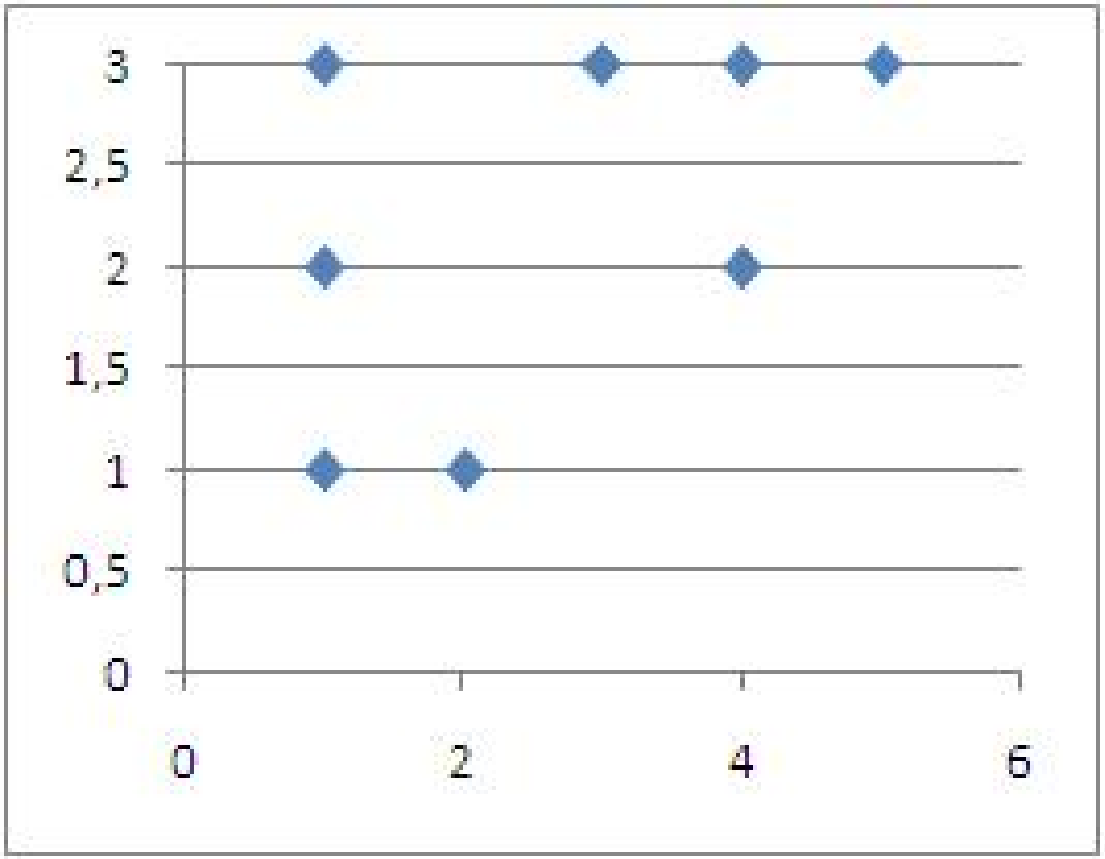
kryterium błędu:

$$E = \sum_{i=1}^{k} \sum_{p \in C_i} |p - m_i|^2$$

$m_i$ jest centroidem klastra $C_i$

# przykladGrupowanie.xls

| a | b | c | d | e | f | g | h |
|---|---|---|---|---|---|---|---|
| (1,3) | (3,3) | (4,3) | (5,3) | (1,2) | (4,2) | (1,1) | (2,1) |



# Przykład, cd

Załóżmy, że $k = 2$, $m_1 = (1, 1)$, $m_2 = (2, 1)$

| pkt | odl. od $m_1$ | odl. od $m_2$ | grupa |
|---|---|---|---|
| a | 2.00 | 2.24 | $C_1$ |
| b | 2.83 | 2.24 | $C_2$ |
| c | 3.61 | 2.83 | $C_2$ |
| d | 4.47 | 3.61 | $C_2$ |
| e | 1.00 | 1.41 | $C_1$ |
| f | 3.16 | 2.24 | $C_2$ |
| g | 0.00 | 1.00 | $C_1$ |
| h | 1.00 | 0.00 | $C_2$ |

błąd $E = 2^2 + 2.24^2 + 2.83^2 + 3.61^2 + 1^2 + 2.24^2 + 0^2 + 0^2 = 36$,
nowe centroidy $m_1 = (1, 2)$, $m_2 = (3.6, 2.4)$

# Przykład, cd

centroidy $m_1 = (1, 2)$, $m_2 = (3.6, 2.4)$

| pkt | odl. od $m_1$ | odl. od $m_2$ | grupa |
|---|---|---|---|
| a | 1.00 | 2.67 | $C_1$ |
| b | 2.24 | 0.85 | $C_2$ |
| c | 3.16 | 0,72 | $C_2$ |
| d | 4.12 | 1.52 | $C_2$ |
| e | 0.00 | 2.63 | $C_1$ |
| f | 3.00 | 0.57 | $C_2$ |
| g | 1.00 | 2.95 | $C_1$ |
| h | 1.41 | 2.13 | $C_1$ |

błąd $E = 7.88$, nowe centroidy $m_1 = (1.25, 1.75)$, $m_2 = (4, 2.75)$

# Przykład, cd

centroidy $m_1 = (1.25, 1.75)$, $m_2 = (4, 2.75)$

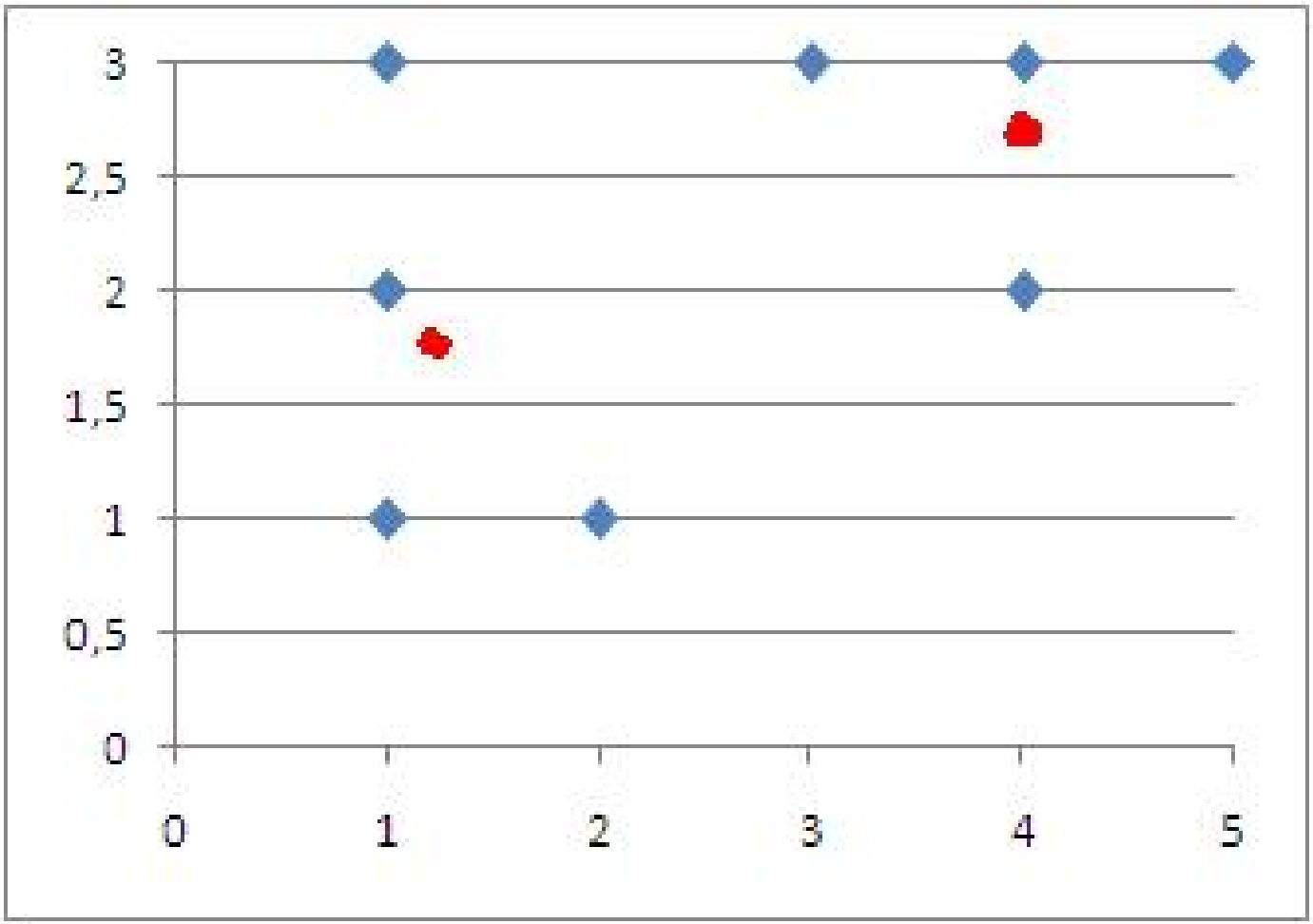
| pkt | odl. od $m_1$ | odl. od $m_2$ | grupa |
|---|---|---|---|
| a | 1.27 | 3.01 | $C_1$ |
| b | 2.15 | 1.03 | $C_2$ |
| c | 3.02 | 0.25 | $C_2$ |
| d | 3.95 | 1.03 | $C_2$ |
| e | 0.35 | 3.09 | $C_1$ |
| f | 2.75 | 0.75 | $C_2$ |
| g | 0.79 | 3.47 | $C_1$ |
| h | 1.06 | 2.66 | $C_1$ |

błąd $E = 6.25$, nowe centroidy $m_1 = (1.25, 1.75)$, $m_2 = (4, 2.75)$ - czyli koniec algorytmu

# Przykład,cd

centroidy $m_1 = (1.25, 1.75)$, $m_2 = (4, 2.75)$

| a | b | c | d | e | f | g | h |
|---|---|---|---|---|---|---|---|
| (1,3) | (3,3) | (4,3) | (5,3) | 1,2) | (4,2) | (1,1) | (2,1) |



# Miary zmienności BCV, WCV

- Zmienność między grupami (between cluster variation BCV),
- Zmienność wewnątrz grupy (within cluster variation WCV),

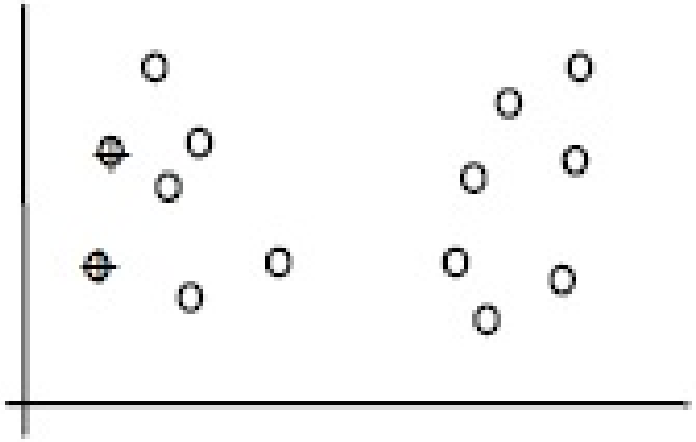
jeżeli przyjąć, że BCV to odległość między centroidami, WCV - suma odległości elementów od odpowiednich centroidów, to w przykładzie iloraz BCV przez WCV zmienia się nast.:

$$\frac{BCV}{WCV} = \frac{1}{36} = 0.028, \quad \frac{BCV}{WCV} = 0.33, \quad \frac{BCV}{WCV} = 0.47$$

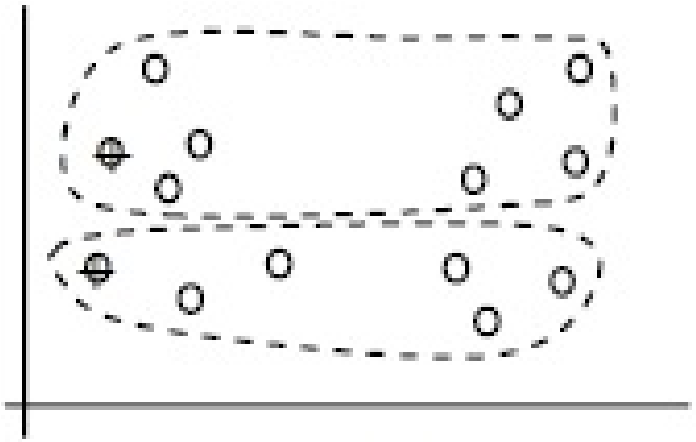
# Komentarz na temat metody k-średnich

- względnie efektywna - złożoność $O(tkn)$, gdzie $n$ jest liczbą rekordów, $k$ - liczbą klastrów, $t$- liczbą iteracji,
- często obliczenia kończą się w lokalnyn optimum,
- końcowy podział obiektów pomiędzy klastrami silnie zależy od początkowego podziału
- konieczność podania wartości $k$,
- źle radzi sobie z wyjątkami i danymi zaszumionymi,
- można stosować tylko do danych dla których określona jest metryka i srednia.

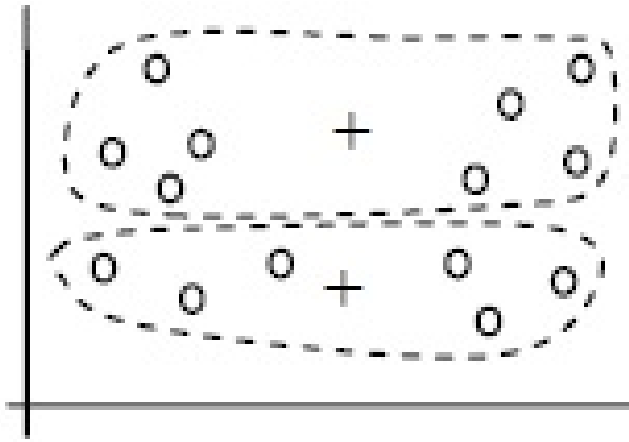
# Przykład - wpływ wyboru centroidów pocz., wer. 1



(A). Random selection of seeds (centroids)

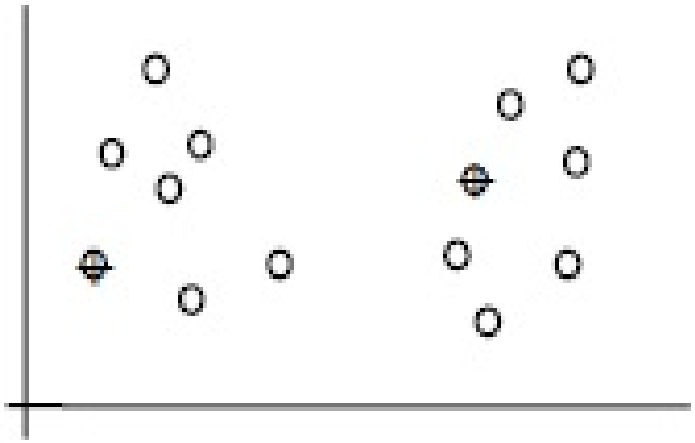


(B). Iteration 1

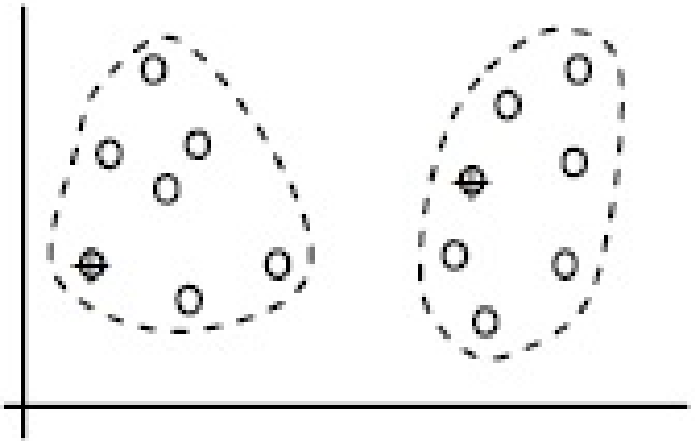


(C). Iteration 2

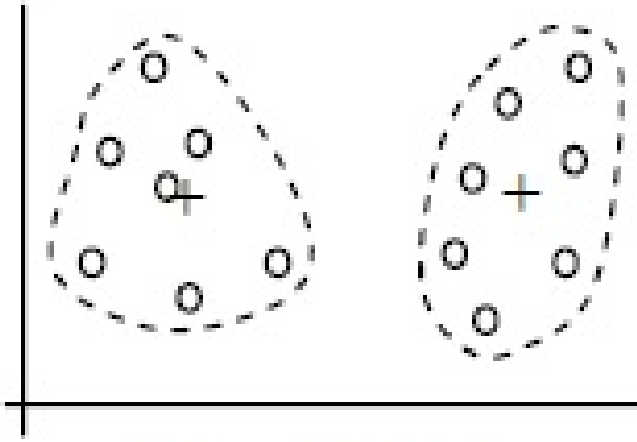
# Przykład - wpływ wyboru centroidów pocz., wer2.



(A). Random selection of $k$ seeds (centroids)



(B). Iteration 1



(C). Iteration 2

# Jak dla danych nominalnych, inaczej kategorycznych

Załózmy, że rekord jest opisany przez $r$ wartości nominalnych.
Klaster $C_j$ jest reprezentowany przez

$$m_j = (m_{j1}, \cdots, m_{jr})$$

gdzie $m_{ji}$ jest względną najczęściej występującą w $C_j$ wartością atrybutu $i$-tego,
odległość jest określona tak jak dla wartości nominalnych

# Algorytm k-średnich harmonicznych (K-Harmonic Means)

B. Zhang (2001) zaproponował użycie sredniej harmonicznej

$$HA(\{a_1, \ldots, a_k\}) = \frac{k}{\sum_{i=1}^{k} \frac{1}{a_i}}$$

która zachowuje się podobnie jak funkcja minimum (tzn. jest mała jeżeli jedna z wartosci $a_i$ jest mała), której używa się w metodzie k-srednich gdzie minimalizujemy błąd:

$$E = \sum_{i=1}^{k} \sum_{p \in C_i} |p - m_i|^2$$

$m_i$ jest centroidem klastra $C_i$

Projekt - sprawdzic, że metoda z użyciem srednich harmonicznych nie jest czuła na początkowy wybór centroidów

# Metoda k-medoids

Metoda k-medoids używa zamiast centroidów - elementów (rekordów). Medoid jest najbardziej centralnie położonym elementem w klastrze.

Algorytm:

1. ustal wartośc k (liczbę grup),
2. losowo przypisz k rekordów (ze zbioru danych) jako medoidy Mi, i=1,...,k,
3. dla każdego rekordu znajdź najbliższy medoid i przypisz rekord do klastra z tym medoidem,
4. w każdym klastrze losowo wybierz element Or różny od medoidu,
5. oblicz koszt $S$ zamiany $Mi$ z $Or$ - czyli rożnicę między błędem dla poprzedniego i aktualnego zbioru medoidów
6. jesli $S < 0$ to dokonaj zmiany medoidu
7. powtarzaj kroki 3-6 dopóki są zmiany (lub nie jest spełnione kryterium końca)

# Rozmyta metoda k-średnich (Fuzzy C-means clustering)

- opiera się na rozmytej funkcji przynależności - jeden element może należeć do kilku klastrów,
- metoda polega na minimalizacji funkcji

$$J_m = \sum_{i=1}^{N} \sum_{j=1}^{noCL} u_{ij}^m |x_i - cl_j|$$

gdzie $m$ jest parametrem większym niż 1 (np. 2),
$N$ jest liczbą rekordów, $noCl$ jest liczbą grup,
$cl_j$ jest centroidem grupy $j$, $u_{ij}$ określa stopień przynależności $i$-tego wiersza $x_i$ do grupy $j$.

- metoda sprowadza się do iteracyjnego procesu modyfikacji $u_{ij}$ oraz $cl_j$.

# Rozmyta metoda k-średnich cd

Modyfikacja $u_{ij}$ oraz $cl_j$:

$$u_{ij} = \frac{1}{\sum_{k=1}^{noCl} \left( \frac{|x_i - cl_j|}{|x_i - cl_k|} \right)^{\frac{2}{m-1}}}$$

$$cl_j = \frac{\sum_{i=1}^{N} u_{ij}^m \cdot x_i}{\sum_{i=1}^{N} u_{ij}^m}$$

# Rozmyta metoda k-średnich, cd

- algorytm rozpoczyna działanie od losowego wyboru wartości $u_{ij}$ z przedziału $(0, 1)$
- iteracje kontynuuje się aż do spełnienia

$$max_{ij}|u_{ij}^{(k+1)} - u_{ij}^{(k)}| < \epsilon$$

  $\epsilon$ jest zadaną dokładnością
- algorytm służy do wyznaczenia centroidów,
- po wyznaczeniu centroidów okresla się dla każdego elementu $x_i$ najbliższy centroid.