CLUSTERING

Metody grupowania danych

Plan wykładu

- Wprowadzenie
  - Dziedziny zastosowania
  - Co to jest problem klastrowania?
- Problem wyszukiwania optymalnych klastrów
- Metody generowania:
  - *k* centroidów (*k* - means clustering)
  - Grupowanie hierarchiczne (hierarchical clustering)
  - Probabilistyczne grupowanie (probability-based clustering)

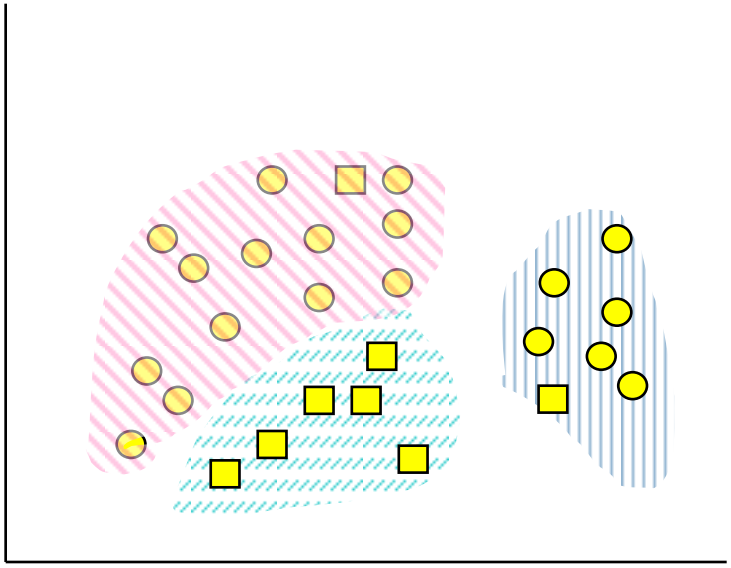
Co to jest klastrowanie

- Klastrowanie (clustering): problem grupowania obiektów o podobnych właściwościach.
- Klaster: grupa (lub klasa) obiektów podobnych (powstająca w wyniku grupowania danych)

Clustering i Klasyfikacja

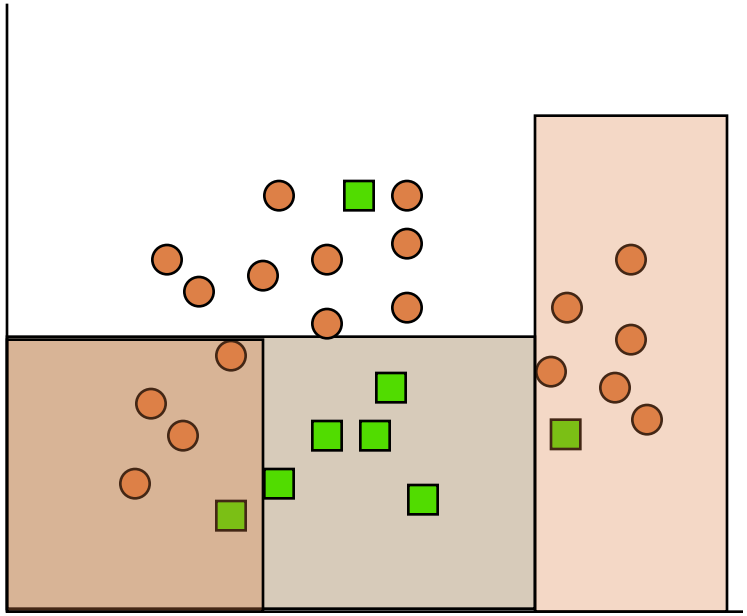
Clustering = uczenie bez nadzoru

Znaleźć „naturalne” skupienia dla zbioru obiektów nie etykietowanych



Klasyfikacja=uczenie z nadzorem

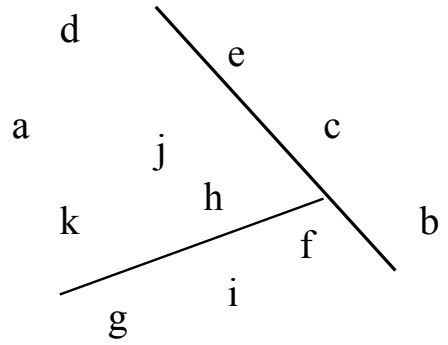
Uczenie metod przewidywania przynależności obiektów do klas decyzyjnych (dane etykietowane)

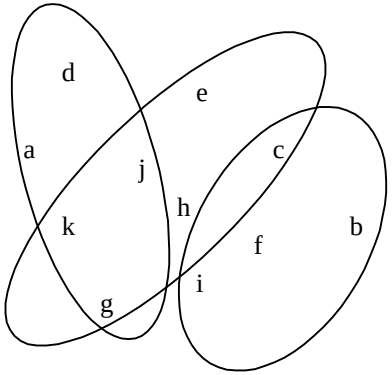


Dziedziny zastosowania

- Medycyna
  - Grupowania chorób
  - Grupowanie objaw u pacjentów np. paranoja, schizophrenia -> właściwa terapia
- Archeologia:
  - taksonomie wydobytych narzędzi ...
- Text Mining
  - Grupowanie podobnych dokumentów -> lepsze wyniki wyszukiwań dokumentów

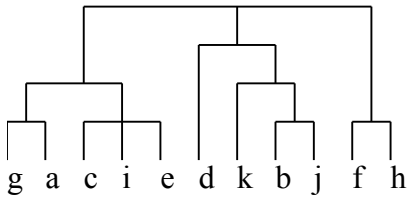
Opis klastrów





	1	2	3
a	0.4	0.1	0.5
b	0.1	0.8	0.1
c	0.3	0.3	0.4
d	0.1	0.1	0.8
e	0.4	0.2	0.4
f	0.1	0.4	0.5
g	0.7	0.2	0.1
h	0.5	0.4	0.1

…



Problem grupowania (clustering)

- Dane są:
  - liczba klastrów *k*
  - funkcja odległości *d* określona na zbiorze obiektów ***P***.
  - funkcja oceny jakości klastrów ***F*** (objective function)

Problem: Podzielić zbiór ***P*** na *k* klastrów tak aby funkcja ***F*** przyjmowała maksymalną wartość.

Klasyfikacja metody

- Unsupervised learning: grupowanie obiektów bez wiedzy o ich kategorii (klasach decyzyjnych).
- Metody klasyfikują się według postaci generowanych klastrów:
  - Czy są one rozłączne czy nierozłączne
  - Czy mają strukturę hierarchiczną czy płaską
  - Czy są określone deterministycznie czy probabilistycznie.

Podstawowe metody clusteringu

- ***k*-centroidów (*k-mean*):**
  - Klastry mają strukturę płaską
  - są określone deterministycznie
- **Grupowanie hierarchiczne:**
  - Klastry mają strukturę drzewiastą
  - są określone deterministycznie
- **Grupowanie w oparciu o prawdopodobieństwo:**
  - Klastry mają strukturę płaską
  - są określone probabilistycznie

Miary odległości

Euclidean distance

$$d(\mathbf{x},\mathbf{y}) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$$

Hamming (city block) distance

$$d(\mathbf{x},\mathbf{y}) = \sum_{i=1}^{n}|x_i - y_i|$$

Tchebyschev distance

$$d(\mathbf{x},\mathbf{y}) = \max_{i=1,2,\ldots,n}|x_i - y_i|$$

Minkowski distance

$$d(\mathbf{x},\mathbf{y}) = \sqrt[p]{\sum_{i=1}^{n}(x_i - y_i)^p},\ p > 0$$

Canberra distance

$$d(\mathbf{x},\mathbf{y}) = \sum_{i=1}^{n}\frac{|x_i - y_i|}{x_i + y_i},\ x_i \text{ and } y_i \text{ are positive}$$

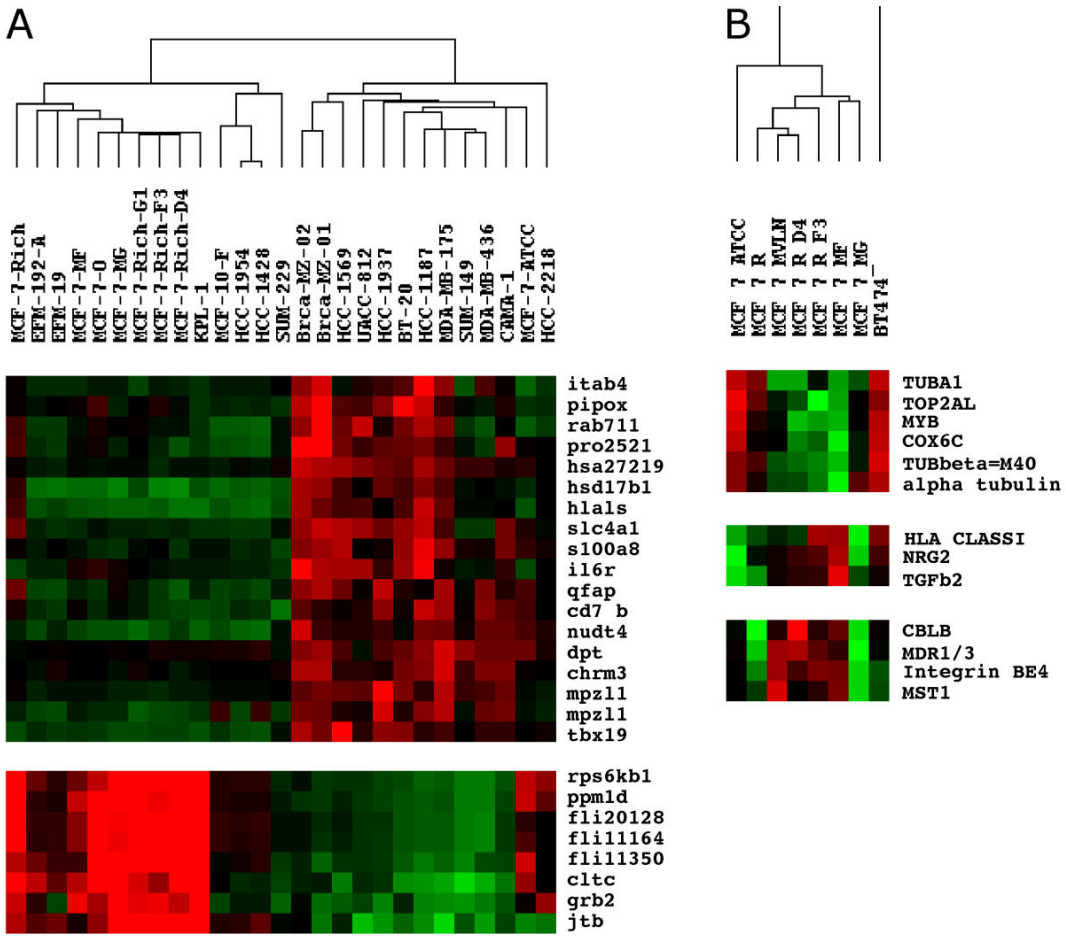
Angular separation

$$d(\mathbf{x},\mathbf{y}) = \frac{\sum_{i=1}^{n} x_i y_i}{\left[\sum_{i=1}^{n} x_i^2 \sum_{i=1}^{n} y_i^2\right]^{1/2}}$$

Grupowanie hierarchiczne

Przykład grupowania profili danych ekspresji RNA

(Nugoli *et al.* *BMC Cancer* 2003 **3**:13)



Algorytm

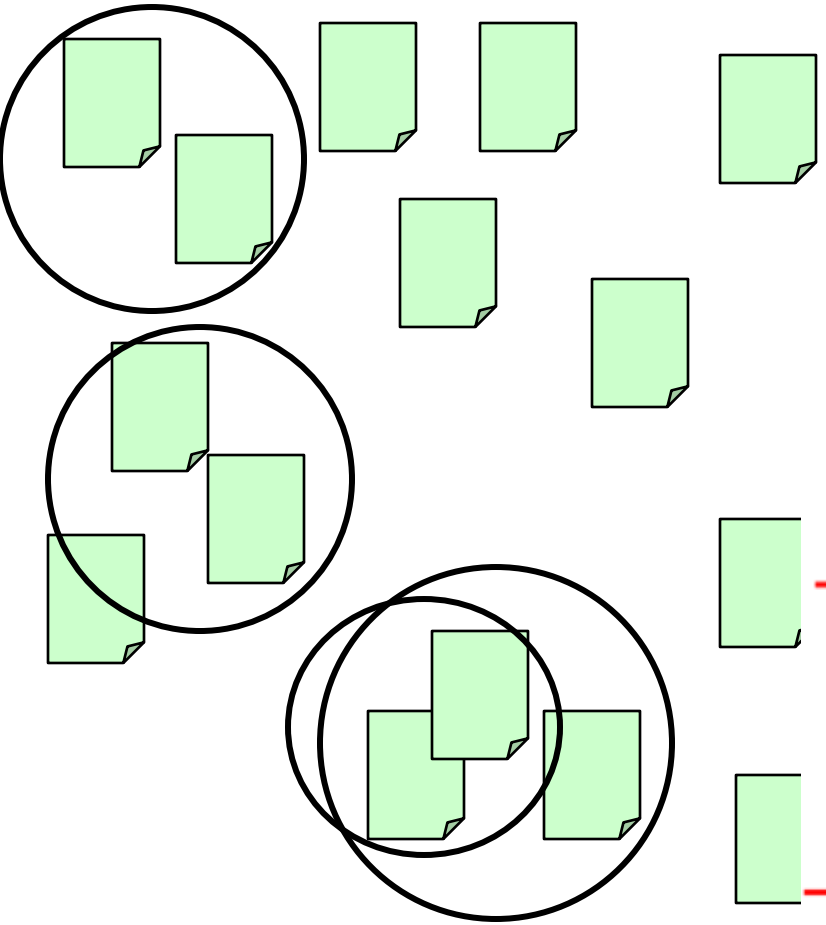
Cel: Budować drzewo klastrów dla zbioru *n* obiektów.

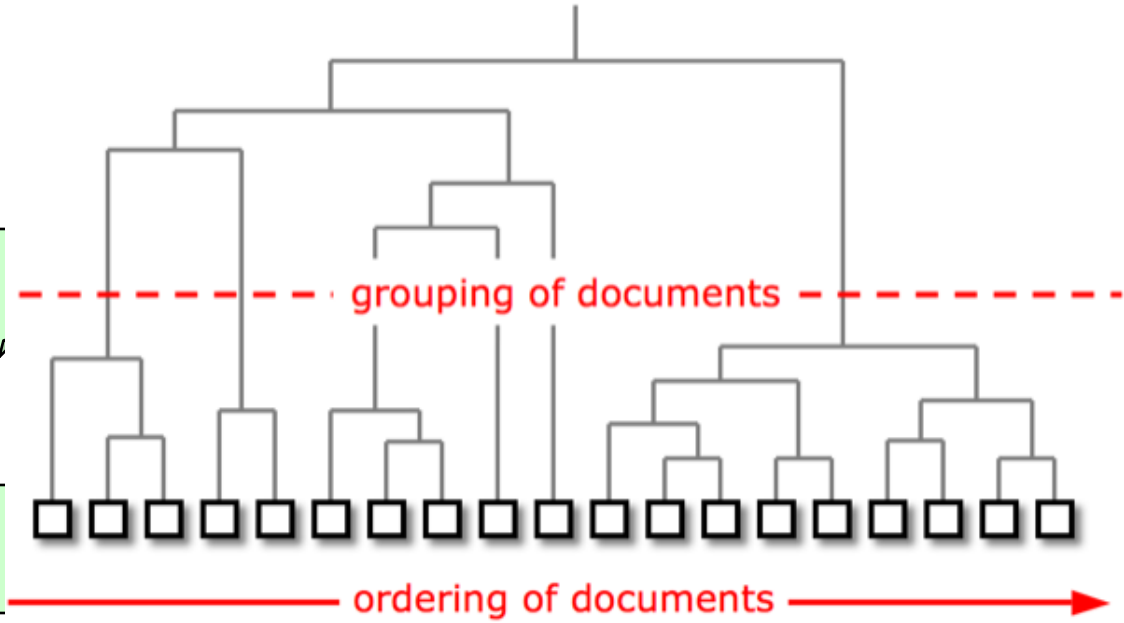
Jakość klastra: suma odległości pomiędzy obiektami w klastrze.

1. Na najniższym poziomie drzewa: *n* liści. Każdy liść (zawierający 1 obiekt) jest klastrem
2. **Repeat**
   - Znajdź najbliższą parę klastrów (parę poddrzew)
   - Połącz te klastry (poddrzewa) w jeden większy

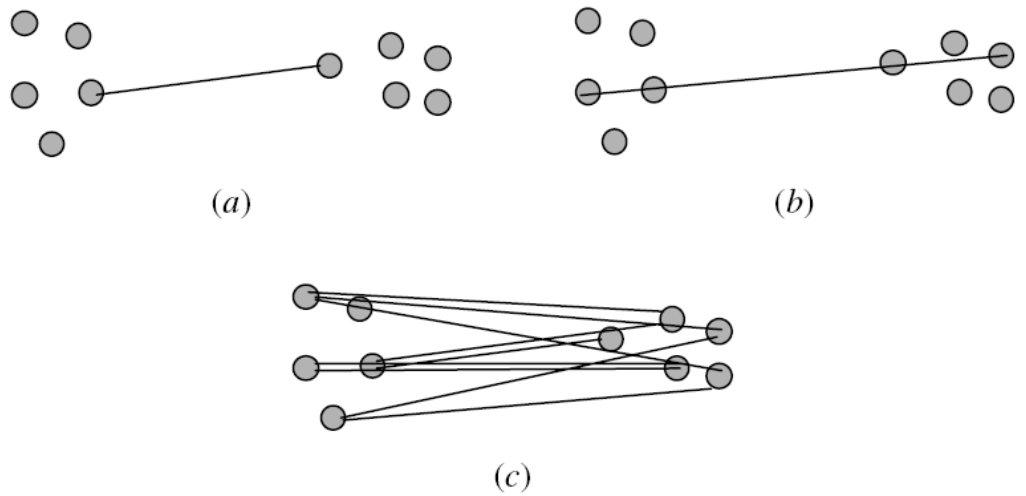
   until STOP

Przykład





Odległość między klastrami



1. **Single linkage (nearest neighbor)**
2. **Complete linkage (furthest neighbor)**
3. **Unweighted pair-group average**
4. **Weighted pair-group average**
5. **Unweighted pair-group centroid**
6. **Weighted pair-group centroid (median).**

Metoda *k*-centroidów (k-means, MacQueen, 1967)

Dane:

- N punktów

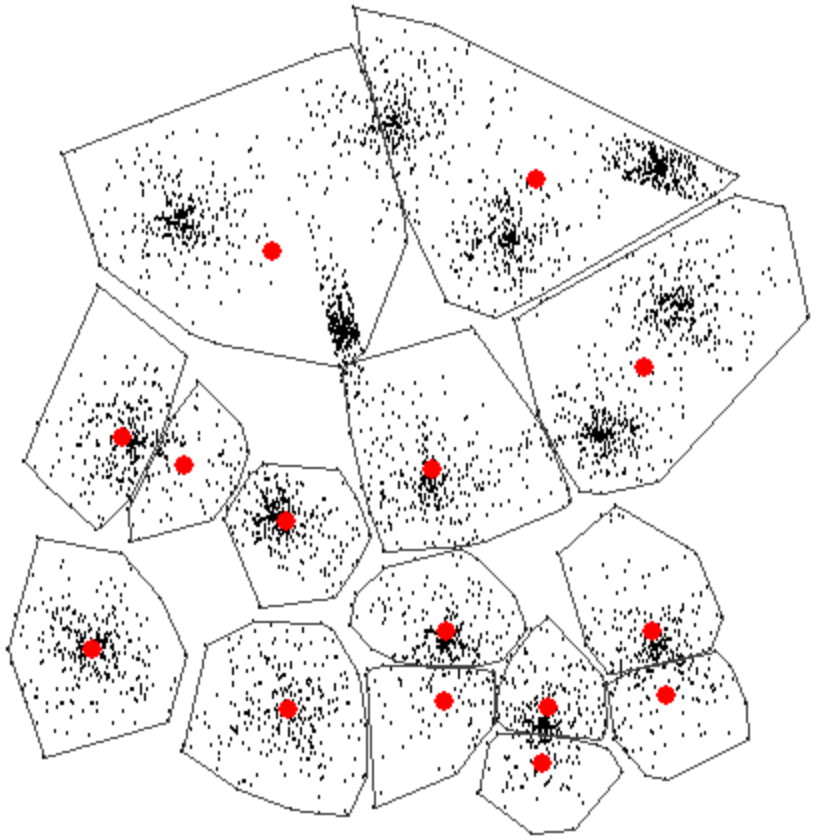
  $\mathbf{x}_1, \mathbf{x}_2 \ldots \mathbf{x}_N$

  w przestrzeni $\boldsymbol{R}^n$
- Parametr $k < N$

Szukane:

- k punktów $\mathbf{c}_1 \ldots \mathbf{c}_k$ (zwanych środkami lub centroidami) będących optymalnymi punktami ze względu na funkcję:

$$F(c_1,\ldots,c_k) = \sum_{i=1}^{N} \min_{j=1,..,k} d\left(x_i, c_j\right)$$



Przykład: kwantyzacja wektorowa

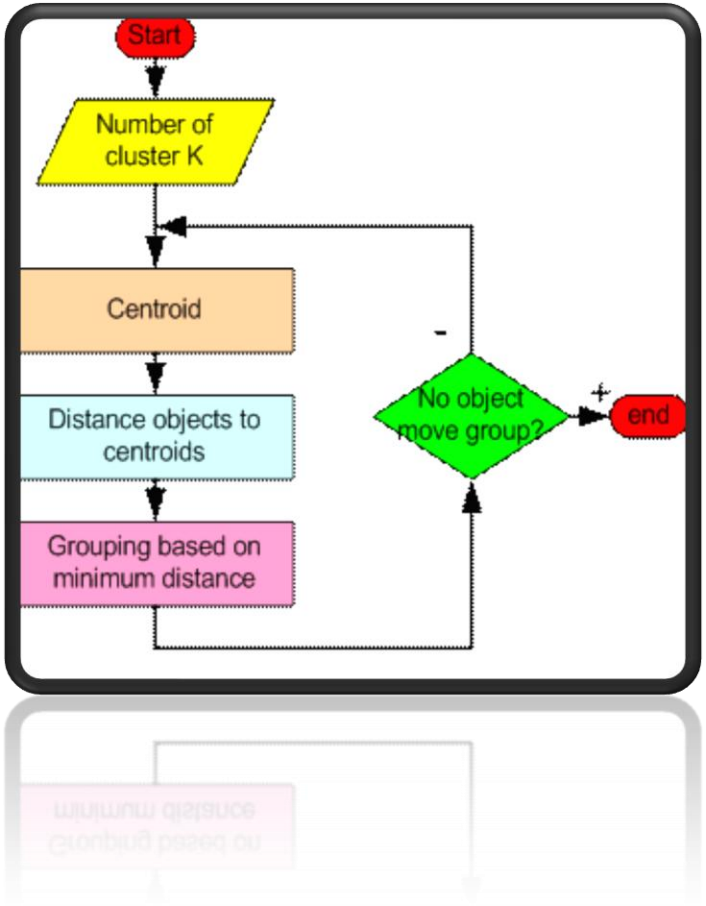
Mały 100x100 obraz kolorowy wymaga 10000*24 = 29.3 kB; (N=10000)

- Jeśli zdołamy reprezentować ten obraz używając jedynie k=32 kolorów
- => możemy kodować każdy punkt za pomocą 5 bitów
- => redukcja pamięci do 6.1 kB + 32*24 bitów na książkę kodową

Algorytm

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**. Powtórz krok 2 dopóty, póki poprawa jakości będzie mała.



Metoda *k* centroidów (c.d.)

Wyznaczanie nowego układu centroidów

- **Idea:** Nowy centroid jest środkiem ciężkości powstającego (w poprzednim przebiegu algorytmu) klastra.
- Współrzędne nowego centroida c:

$$c(x_i) = \frac{p_1(x_i) + \ldots + p_k(x_i)}{k}$$

gdzie

$p_1, p_2, \ldots, p_k$ – punkty należące do klastra.

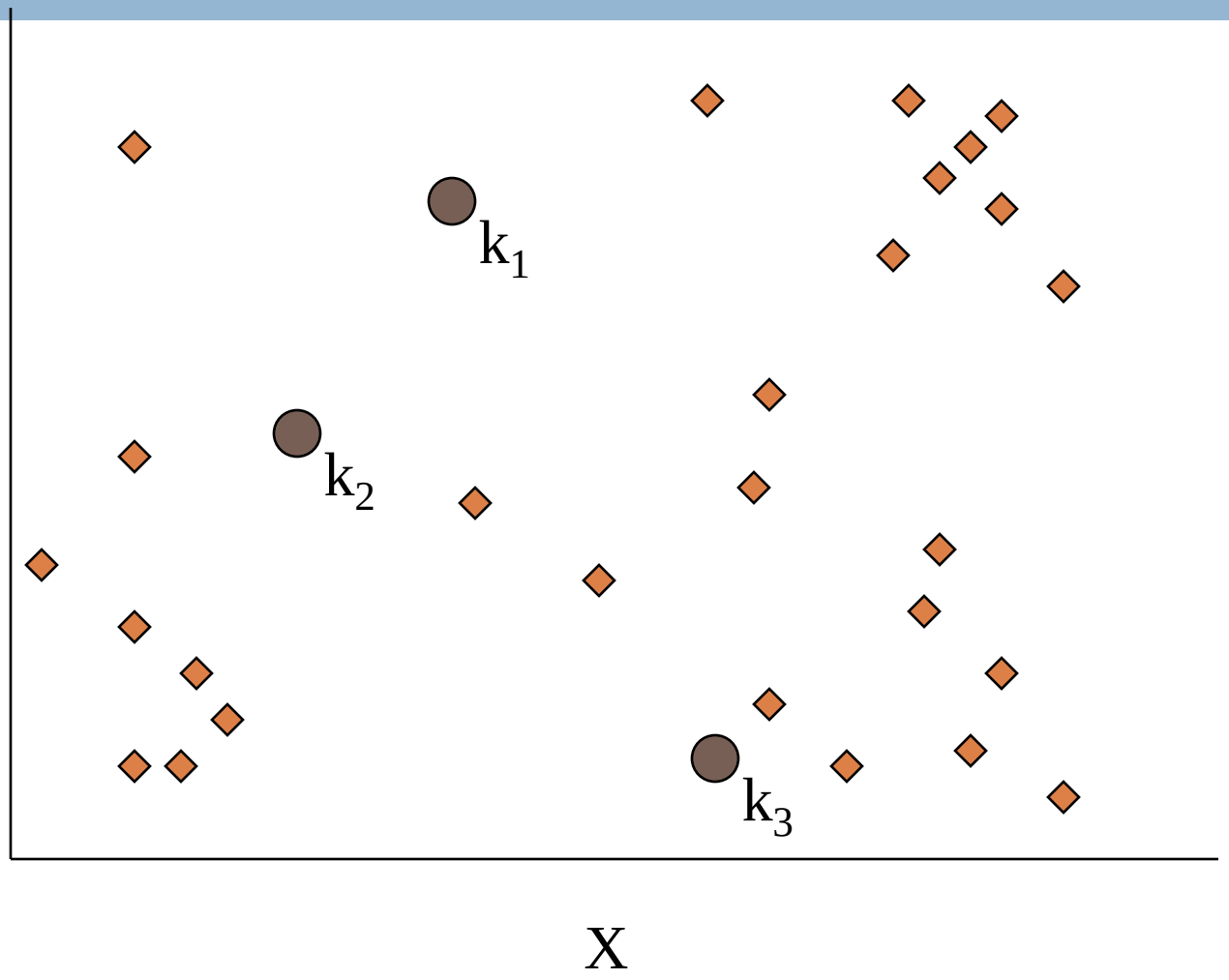
$c(x_i)$, $p_1(x_i), \ldots, p_k(x_i)$ – *i*-ta współrzędna.

Własciwości metody k - centroidów

- 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.

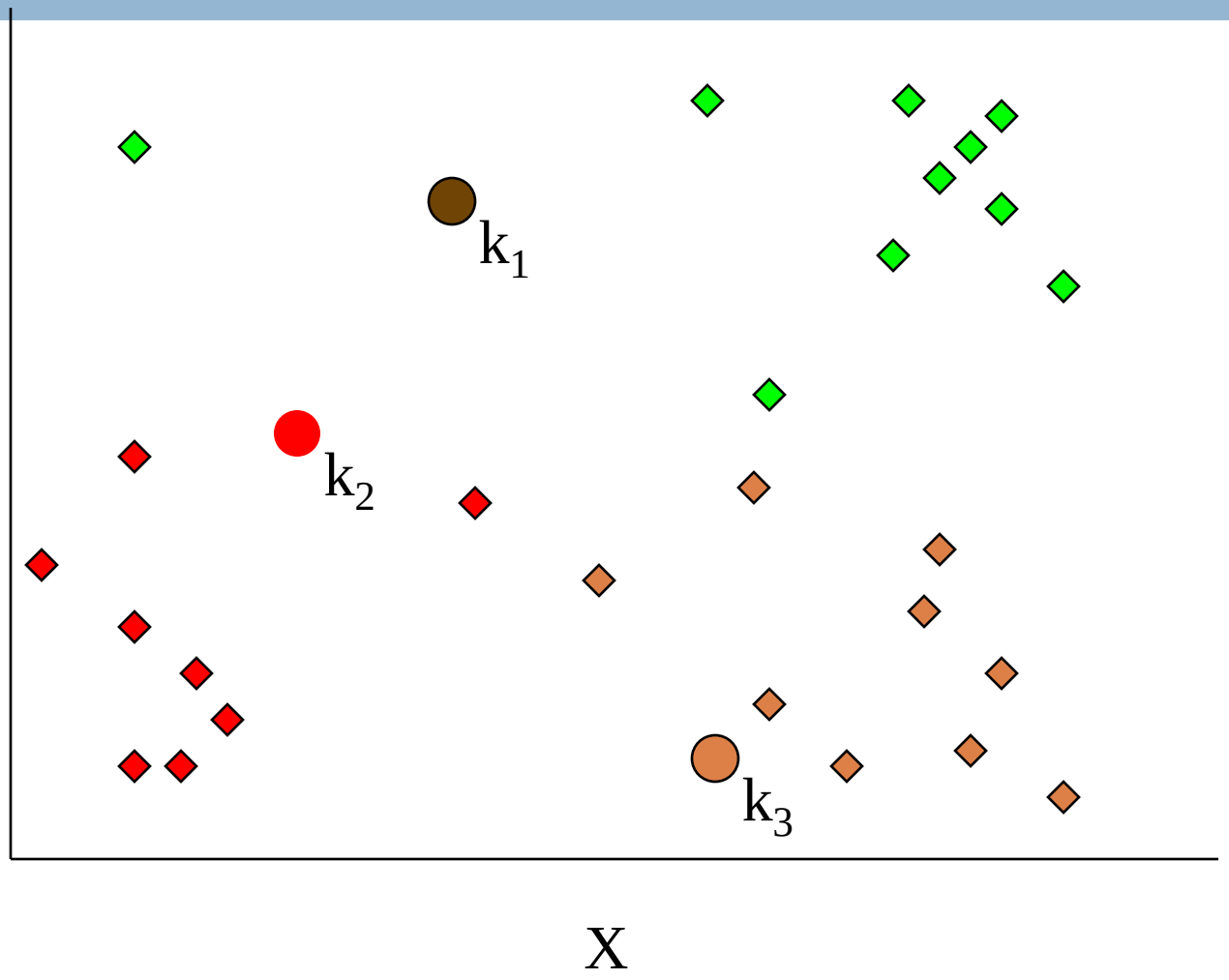
Przykład k=3, Krok 1

Wybierz losowo 3 punkty jako początkowy zbiór środków.



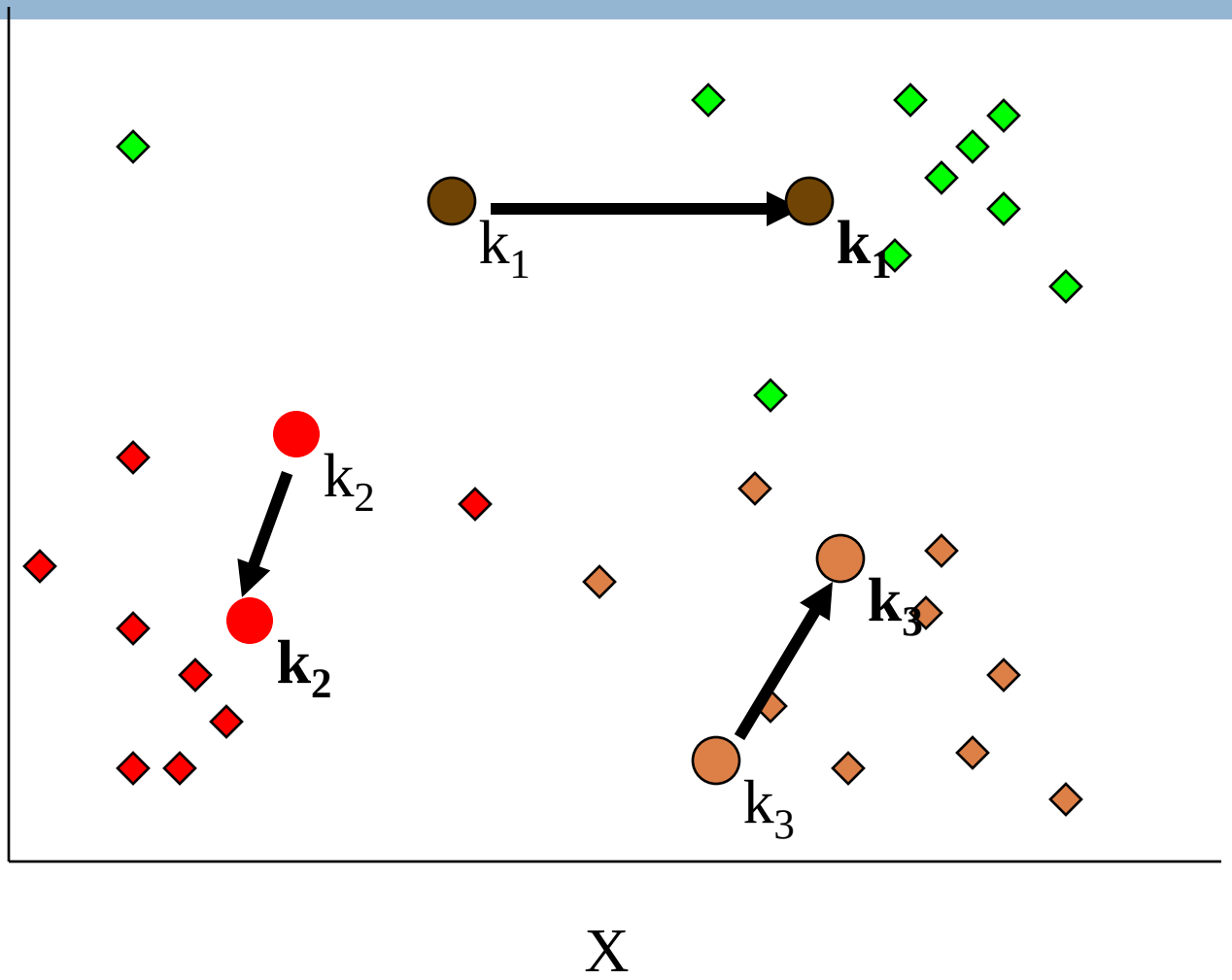
Przykład k=3, Krok 2

Przypisanie każdego z punktów do najbliższego środka



Przykład k=3, Krok 3

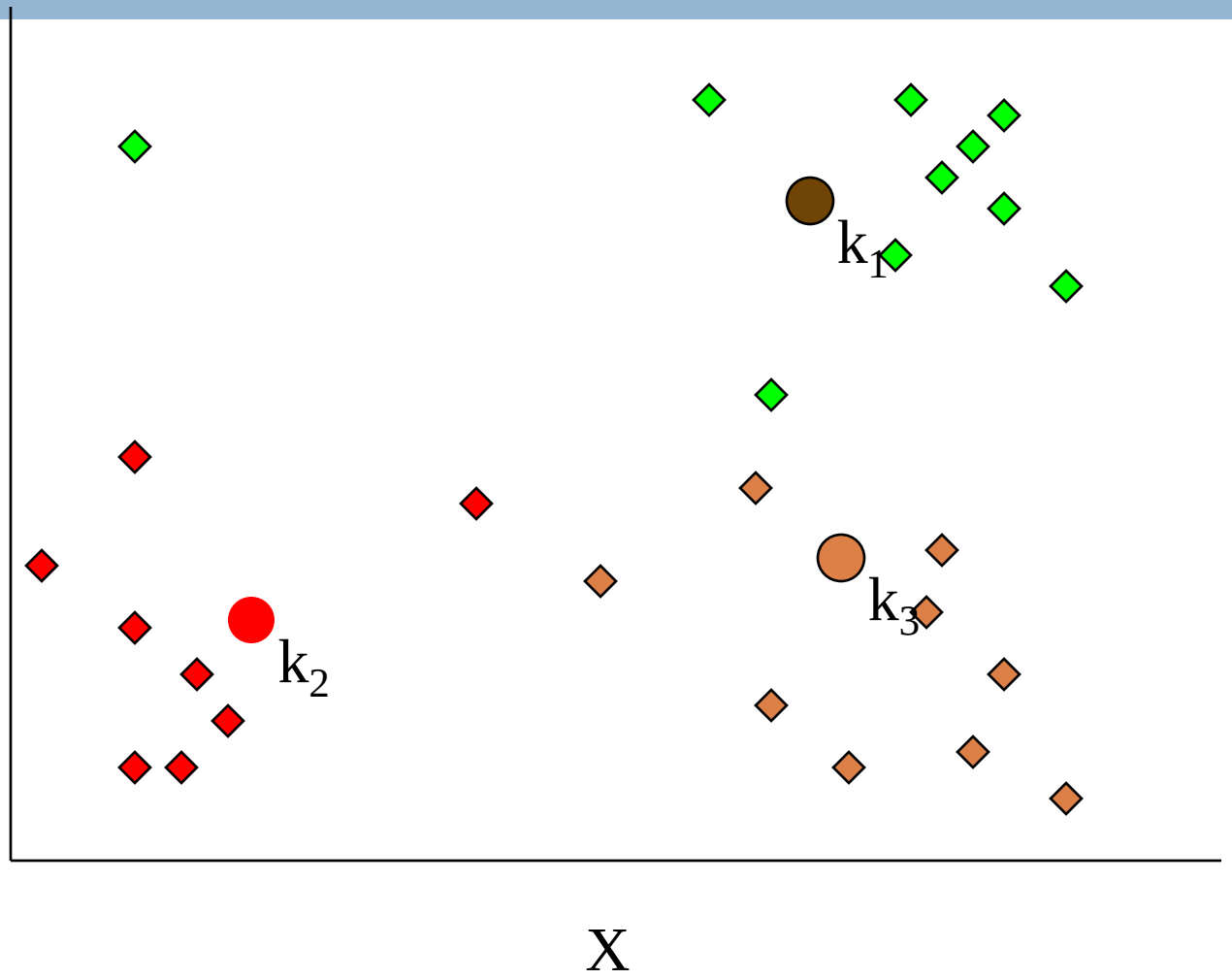
Przesuwanie centroidów do środków klastrów.



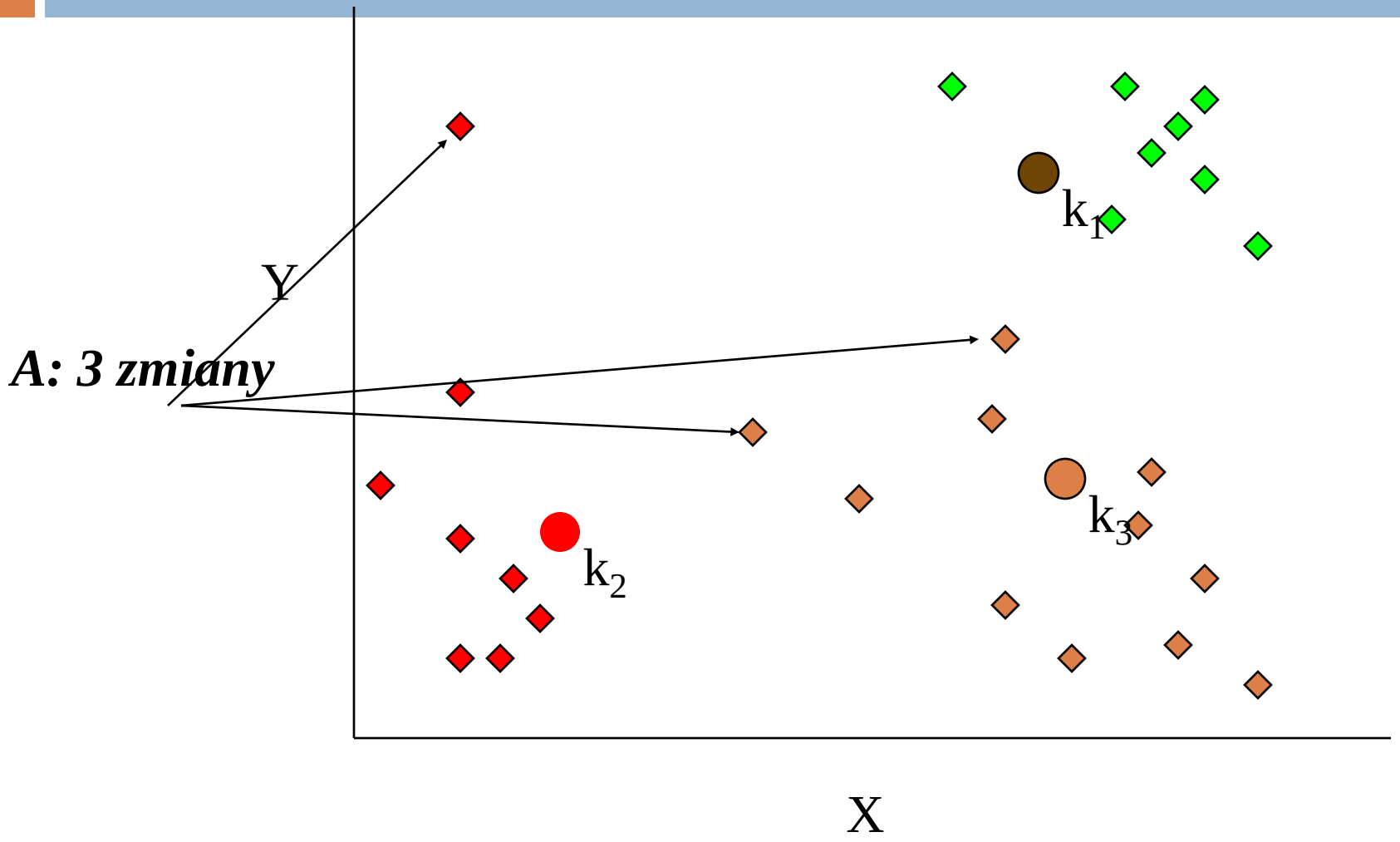
Przykład k=3, Krok 4

Znów przypisać punkty do najbliższych środków

Pyt.: Które z tych punktów zmienia grupę?

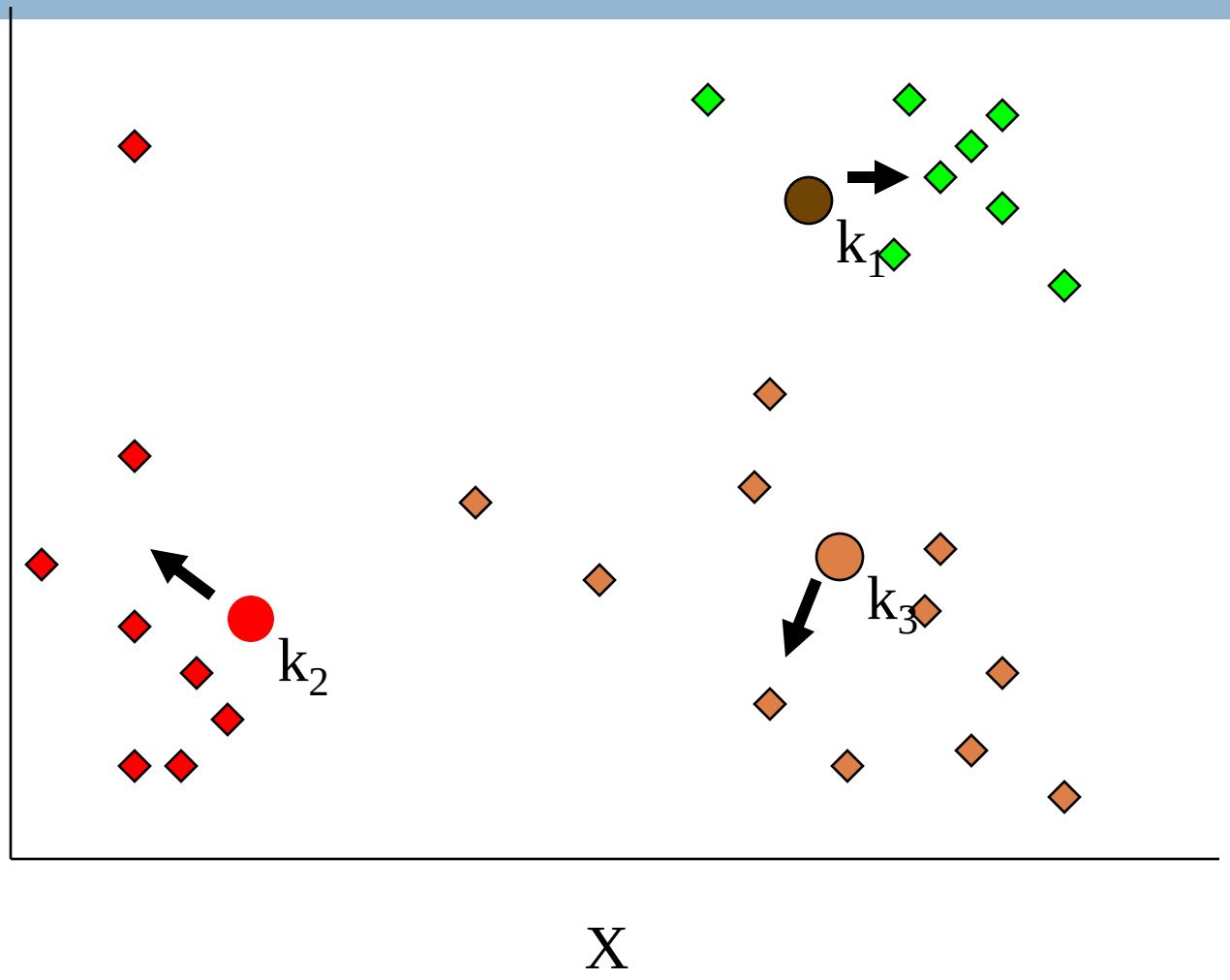


Przykład k=3, Krok 4a



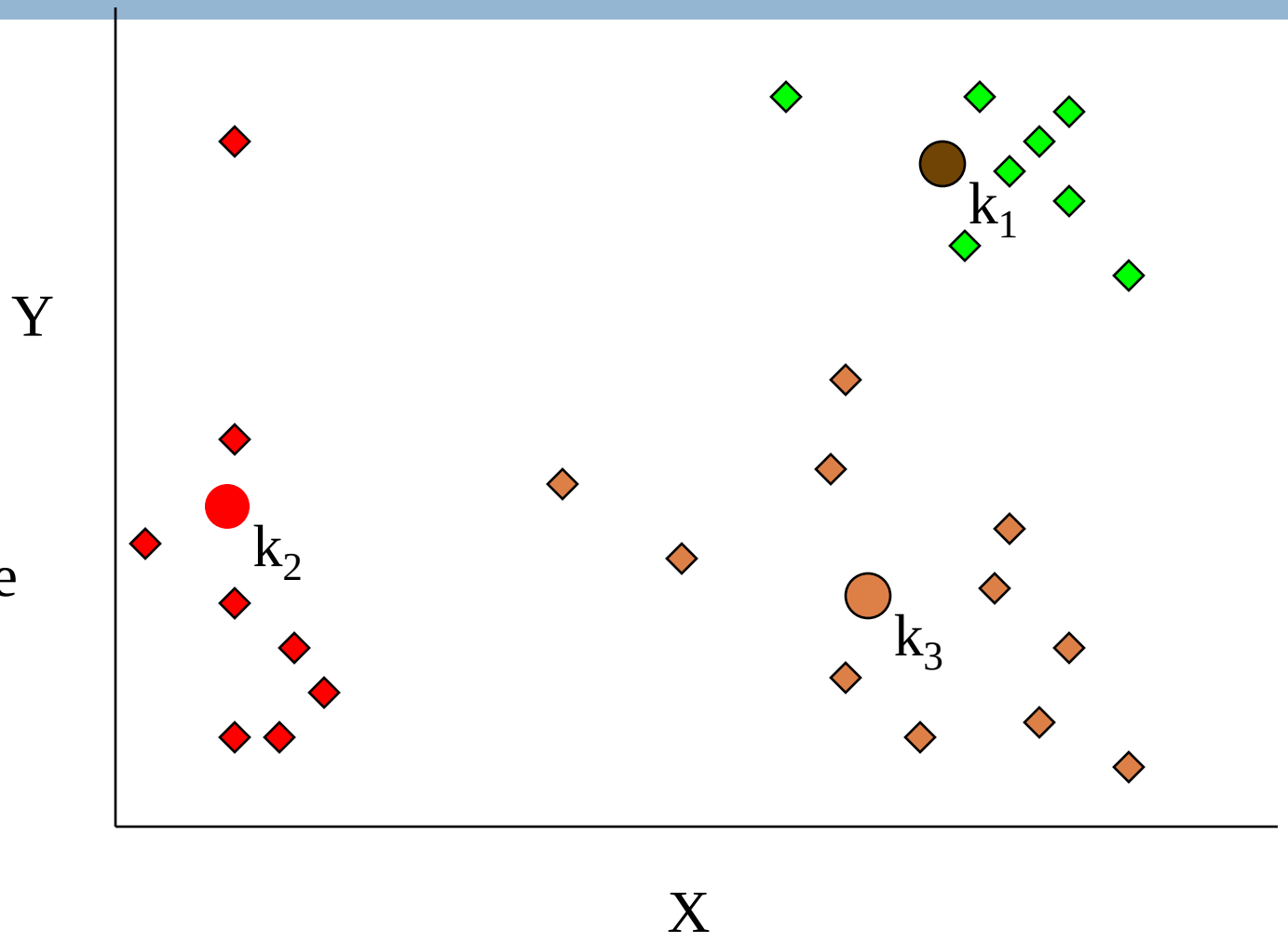
Przykład k=3, Krok 4b

re-compute cluster means

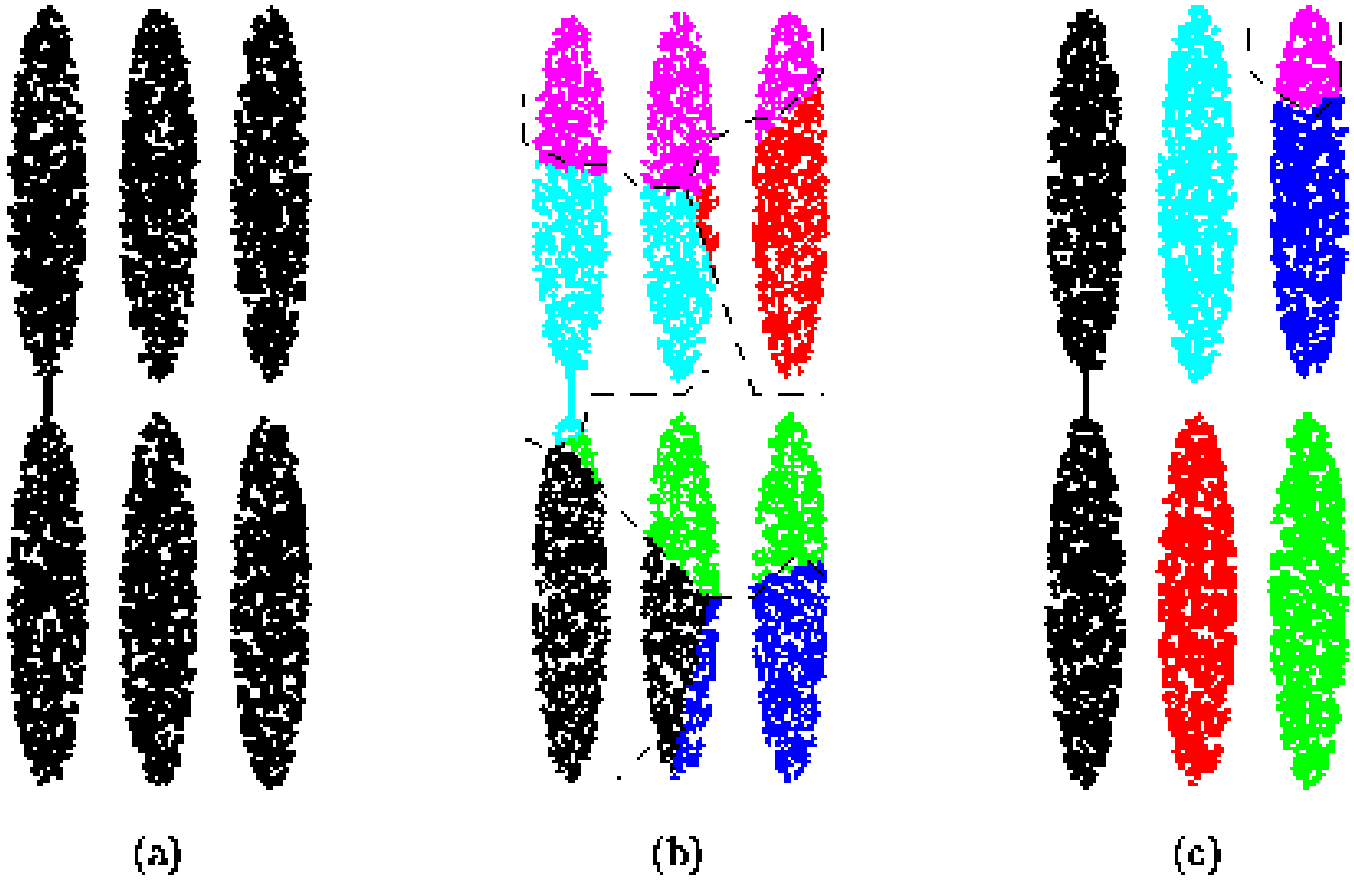


Przykład k=3, Krok 5

Przesuwanie centroidów do środków nowych klastrów



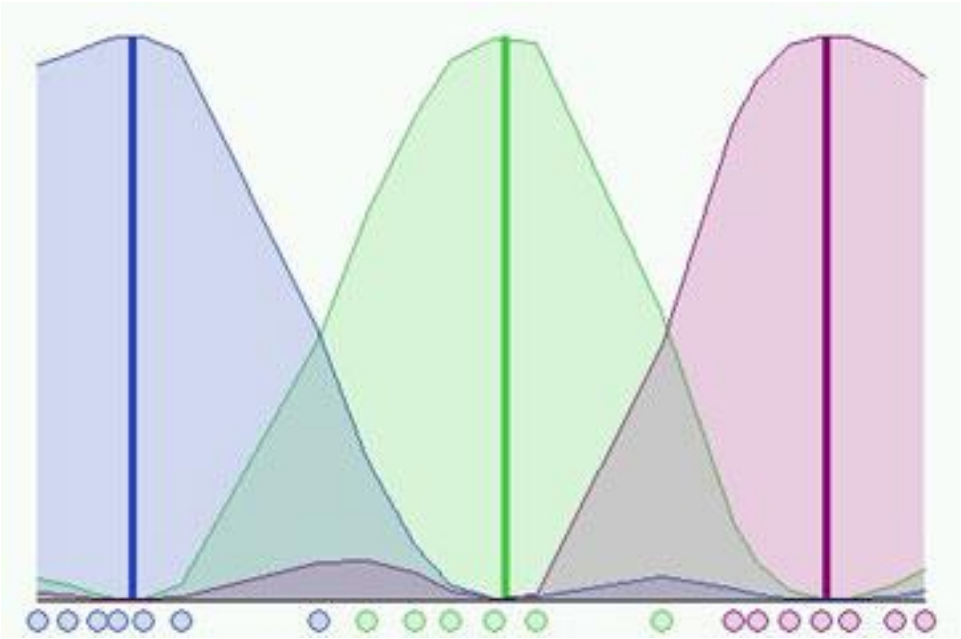
Anomalie metody centroidów



Probabilistyczne grupowanie

- Obiekt należy do klastra z pewnym stopniem prawdopodobieństwa.
- **Idea**: Każdy klaster jest opisany jednym rozkładem prawdopodobieństwa.
- **Założenie**:
  - Wszystkie rozkłady są rozkładami normalnymi.
  - Rozkłady różnią się *wartościami oczekiwanymi* (μ) i *odchyleniami standardowymi* (σ).

Przykład: Trzy probabilistyczne klastry



Stopień należności obiektu do klastra

- Obiekt x należy do klastra *A* z prawdopodobieństwem:

$$P[A \mid x] = \frac{P[x \mid A].P[A]}{P[x]} = \frac{f(x;\mu_A;\sigma_A).p_A}{P[x]}$$

- gdzie $f(x;\mu_A;\sigma_A)$ - rozkład normalny

$$f(x;\mu;\sigma) = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

Algorytm EM

- EM – Expection - Maximization
- Dana jest liczba klastrów k.
- **Cel**: Dla każdego atrybutu rzeczywistego znaleźć k układów parametrów (μ_i, σ_i, p_i) dla $i = 1,..,k$ (opisów k klastrów).
- Dla uproszczenia opisu, niech $k = 2$
- Algorytm jest opisany dla wybranego atrybutu.

Algorytm EM

- **Idea:** adoptować algorytm *k* centroidów.

Krok 1. Wybierz losowo 2 układy parametrów (μ_A, σ_A, p_A) i (μ_B, σ_B, p_B);

Krok 2. Oblicz oczekiwane stopnie przynależności obiektów do klastrów („expectation” step)

Krok 3. Wyznacz nowe układy parametrów w celu *maksymalizacji funkcji jakości* „likelihood” („maximization” step);

Krok 4. Powtórz krok 2 dopóty, póki poprawa jakości będzie mała.

Funkcja oceny jakości

- Funkcja dopasowania „*likelihood*”:

dla dwóch klastrów *A* i *B*:

$$\sum_i (p_A . P[x_i \mid A] + p_B . P[x_i \mid B])$$

Wyznaczanie nowych parametrów

- Szuka się dwóch układów parametrów:

$$(\mu_A, \sigma_A, p_A) \text{ i } (\mu_B, \sigma_B, p_B)$$

- Jeśli w_i jest stopniem przynależności i – tego obiektu do klastra A to :

$$\mu_A = \frac{w_1 x_1 + w_2 x_2 + \ldots + w_n x_n}{w_1 + w_2 + \ldots + w_n}$$

$$\sigma_A^2 = \frac{w_1 (x_1 - \mu_A)^2 + w_2 (x_2 - \mu_A)^2 + \ldots + w_n (x_n - \mu_A)^2}{w_1 + w_2 + \ldots + w_n}$$

Bibliografia

- Brian S. Everitt (1993). *Cluster analysis*. Oxford University Press Inc.
- Ian H. Witten, Eibe Frank (1999). *Data Mining. Practical ML Tools and Techniques with Java Implementations*. Morgan Kaufmann Publishers.