Shlukování

Zpracováno s využitím skvělého tutoriálu autorů
Eamonn Keogh, Ziv Bar-Joseph a Andrew Moore

Osnova přednášky

- **Motivace**
- Míra vzdálenosti
- Hierarchické shlukování
- Hodnocení kvality rozkladu
- Shlukování rozkladem
  - k-means (k-středů)
  - EM (expectation maximization) algoritmus, Gaussovská směs
  - Odhad počtu shluků

Co je to shlukování?

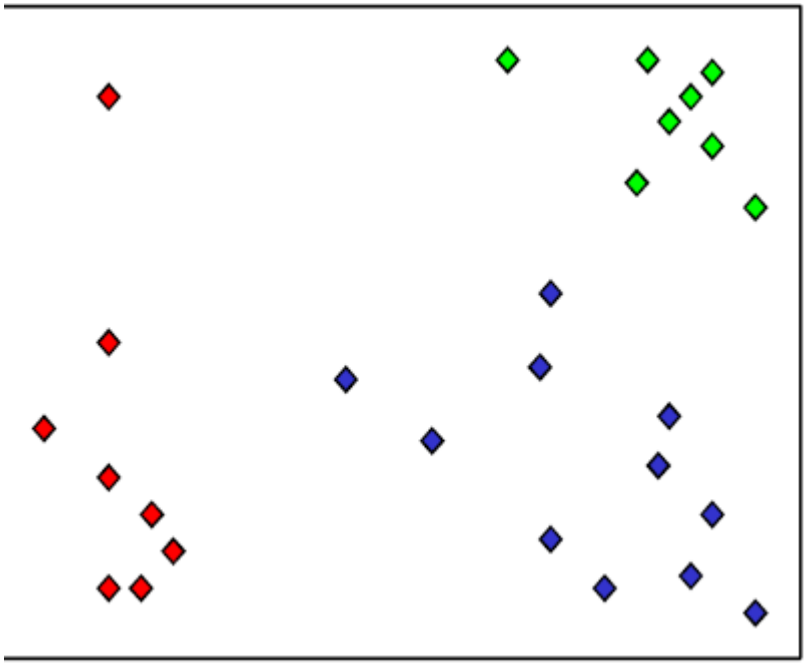
Seskupení dat do takových ***shluků***, že podobnost dat

- uvnitř shluku je vysoká,
- z různých shluků je nízká.

Hledáme „přirozené“ seskupení, které nabízí lepší popis dat.

Proč nás to zajímá?

Dá se to použít pro něco konkrétního?

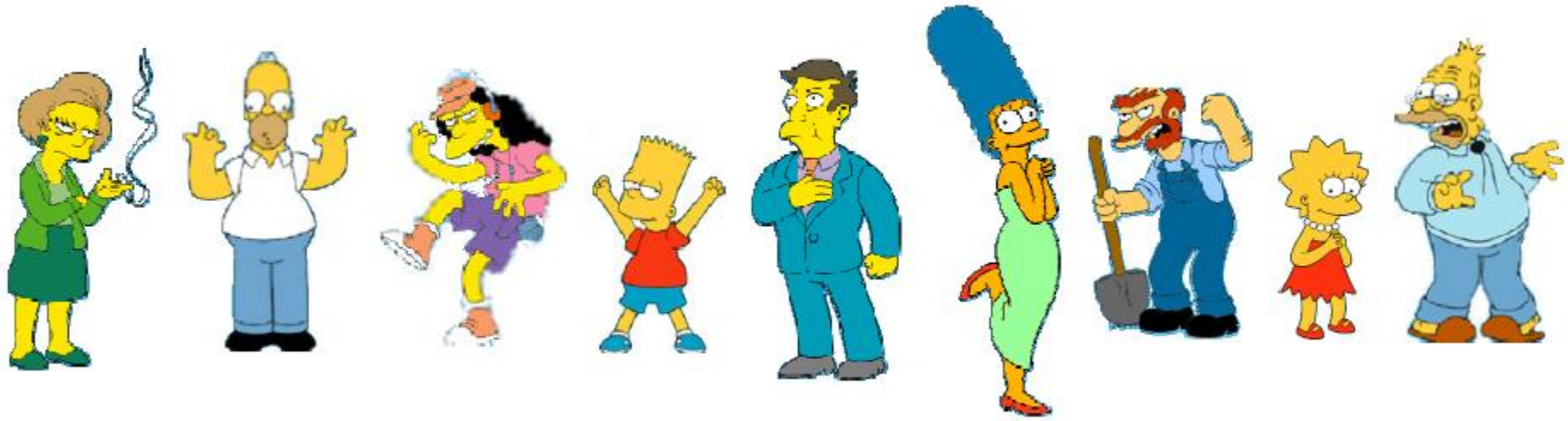


Důležité pro nalezení vnitřní struktury dat: Linného **systém,** Mendělejevova tabulka, **Clusty/Yippy** *pro lepší orientaci ve výsledcích webového vyhledávače, **segmentace obrazů** jako základ pro rozpoznávání objektů či definici hranic, …

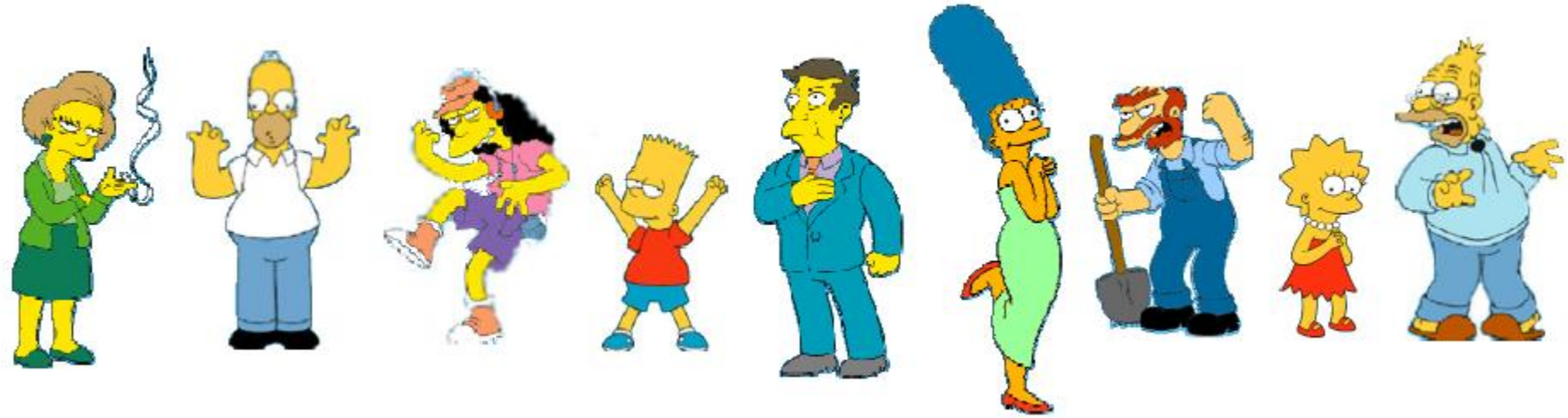
* „ .. **Clusty** search for ‘**pearl**’ organizes the top 250-500 results into subject folders such as Jewelry, Pearl Harbor, Pearl Jam, Steinbeck Novel and Daniel Pearl. Clusty allows users to focus on the area of interest without all the chaff."

Co jsou přirozené shluky z těchto individuí?

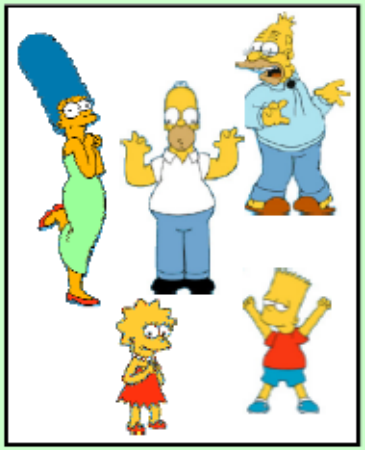
Shlukování je chápáno jako deskriptivní úloha, jejímž cílem je charakterizovat kategorie individuí, které mají smysl a které umožní lépe chápat pozorovaný svět.



Co jsou „přirozené“ shluky z těchto individuí?

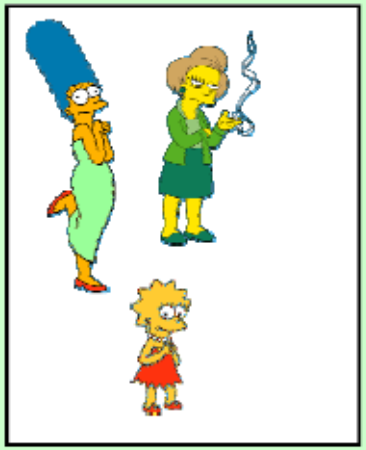


Pojem „přirozenost“ má zde velmi subjektivní charakter!



Rodina Simpsonů

Zaměstn. školy



Ženy

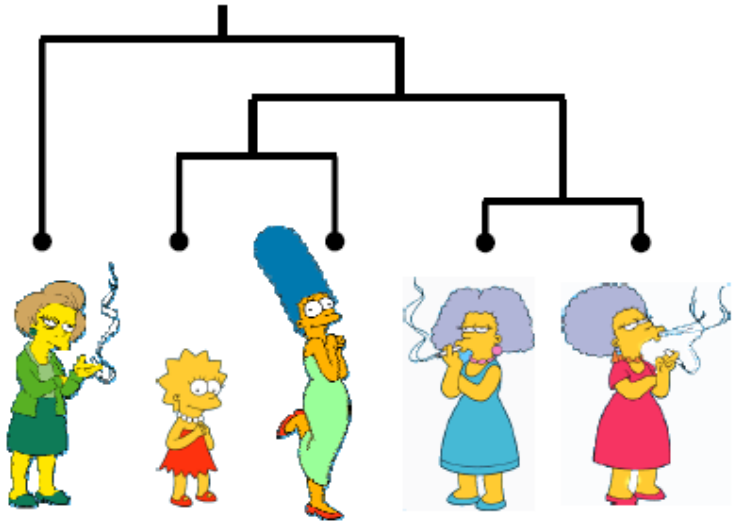
Muži

Dva přístupy ke tvorbě shluků

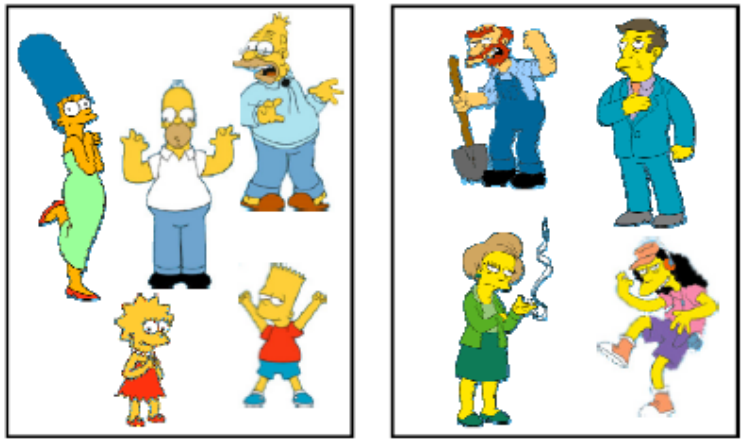
Shlukování rozkladem konstruuje různé rozklady množiny uvažovaných objektů a z těch vybírá nejvhodnější vzhledem k nějakému kritériu.

Hierarchické shlukování postupně sdružuje uvažované objekty podle zvoleného kritéria

Hierarchické shlukování



Shlukování rozkladem



Co je to podobnost?

Podobnost je obtížné definovat, ale lehce ji rozpoznáme, když ji vidíme!

- Jedná se o filosofickou otázku.
- Pragmatická charakteristika staví na definici **vzdálenosti**

Osnova přednášky

- Motivace
- **Míra vzdálenosti**
- Hierarchické shlukování
- Hodnocení kvality rozkladu
- Shlukování rozkladem
  - k-means (k-středů)
  - EM (expectation maximization) algoritmus, Gaussovská směs
  - Odhad počtu shluků

Jak definovat **míru vzdálenosti**?

Funkce ***D***, která každým 2 uvažovaným objektům o_1 a o_2 přiřazuje reálné číslo $D(o_1,o_2)$ tak, aby pro libovolné objekty **A**,**B** a **C** měla funkce ***D*** tyto vlastnosti

1. $D(\mathbf{A},\mathbf{B}) = D(\mathbf{B},\mathbf{A})$ symetrie
2. $D(\mathbf{A},\mathbf{B}) = 0$ iff $\mathbf{A} = \mathbf{B}$ konzistence samo-podobnosti
3. $0 \leq D(\mathbf{A},\mathbf{B})$ pozitivita

Platí-li navíc ještě následující vlastnost

4. $D(\mathbf{A},\mathbf{B}) \leq D(\mathbf{A},\mathbf{C}) + D(\mathbf{C},\mathbf{B})$ trojúhelníková nerovnost

říkáme, že míra ***D*** je metrická míra vzdálenosti

Nejčastější míry vzdálenosti

Minkowského míra pro objekty popsané atributy s reálnými hodnotami

- Nechť objekt je popsán p atributy jako reálný vektor. Uvažujme objekty $\boldsymbol{x} = (x_1, x_2, \ldots, x_p)$ a $\boldsymbol{y} = (y_1, y_2, \ldots, y_p)$
- Minkowského metrická vzdálenost je definována

  $d(\boldsymbol{x},\boldsymbol{y}) = (|x_1-y_1|^g + |x_2-y_2|^g + \ldots (|x_p-y_p|^g)^{1/g}$

  kde g je parametr volený podle potřeb aplikace, např.:
  - Je-li $g = 1$, mluvíme o Manhattanské vzdálenosti
  - Je-li $g = 2$, jedná se Eucleidovu vzdálenost

Nejčastější míry vzdálenosti

Nechť objekt je popsán p atributy jako binární vektor (s hodnotami 0 nebo 1). Uvažujme objekty $\boldsymbol{x}= (x_1, x_2, \ldots, x_p)$ a $\boldsymbol{y}= (y_1, y_2, \ldots, y_p)$

Kontingenční tabulka	pro vektory x a y
$\boldsymbol{q}$ =card $\{i \leq p: x_i = y_i = 1\}$	$\boldsymbol{r}$ =card $\{i \leq p: x_i = 1, y_i = 0\}$
$\boldsymbol{s}$ =card $\{i \leq p: x_i = 0, y_i = 1\}$	$\boldsymbol{t}$=card $\{i \leq p: x_i = y_i = 0\}$

Míra pro objekty s binárními atributy vychází z kontingenční tabulky.

V případě, že obě hodnoty (0 i 1) mají stejnou důležitost, označuje se atribut jako symetrický, jinak je asymetrický.

- Jsou-li **všechny atributy symetrické**, pak $d(x,y) = (r+s) / (q+r+s+t)$
- Jsou-li **všechny atributy asymetrické** a výsledek 1 je významnější než 0, pak se nezpočítává shoda v hodnotách 0: $d(x,y) = (r+s) / (q+r+s)$

Nejčastější míry vzdálenosti

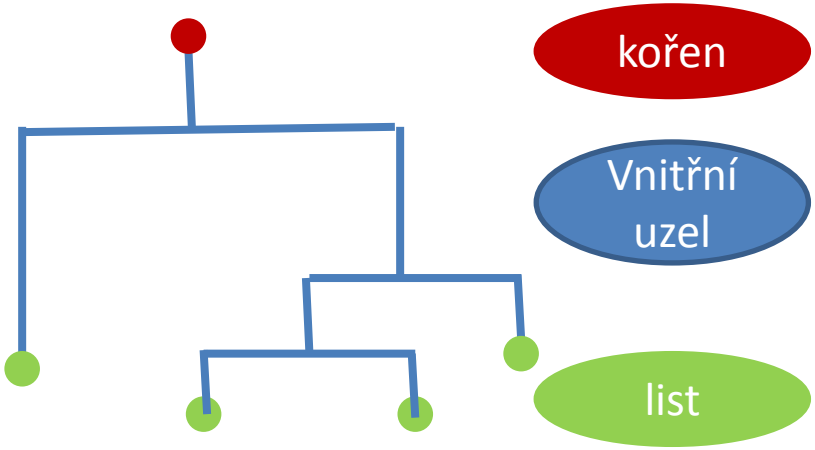
Nechť objekt je popsán *p* atributy jako vektor s nominálními hodnotami, tj. hodnoty mají výčtový typu (např. barvy nebo jména firem). Uvažujme objekty $\boldsymbol{x} = (x_1, x_2, \ldots, x_p)$ a $\boldsymbol{y} = (y_1, y_2, \ldots, y_p)$, jejichž hodnoty se přesně shodují na *m* pozicích.

- Pak se jako míra používá jednoduchá shoda $d(x,y) = (p-m) / p$

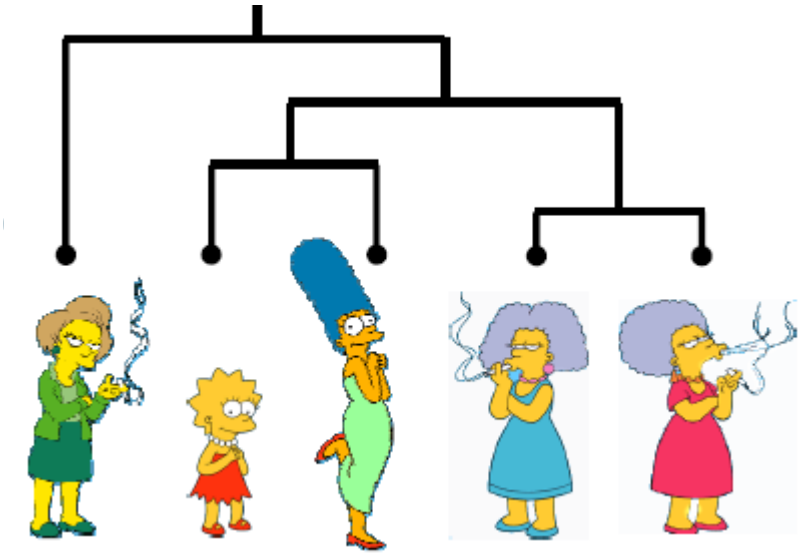
Je-li objekt popsán *p* atributy jako vektor s ordinálními hodnotami, tj. jde o konečný definiční obor, jehož hodnoty jsou uspořádány (např. čísla měsíců), používá se míra podobná jako u reálných hodnot (původní hodnota je nejprve transformována do intervalu [0,1] .

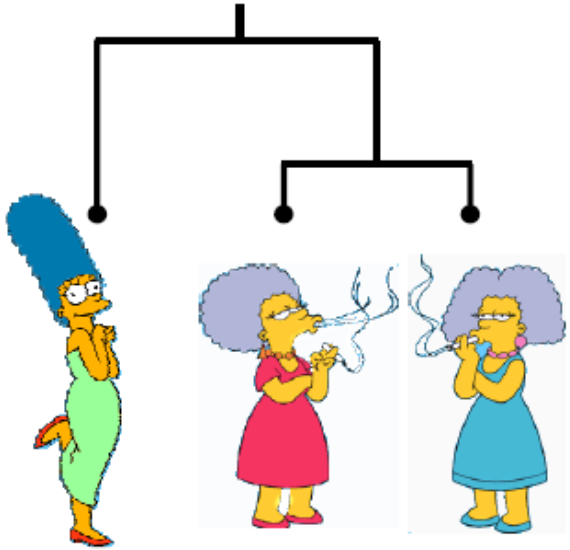
V případě, že různé atributy mají různé typy, míra vzdálenosti vznikne jako součet měr po příslušných typech.

Dendrogram jako užitečný nástroj pro kompaktní znázornění vztahů podobnosti ve skupině



Vzdálenost 2 objektů ***a*** a ***b*** v dendrogramu je vyjádřena výškou (= vzdáleností od listu) nejnižšího uzlu, který leží na společné části cesty od kořene k ***a*** i k ***b.***





Jaké vlastnosti by měl mít algoritmus pro tvorbu shluků?

- Škálovatelnost vzhledem k rozsahu dat (polynom., nejlépe lineární složitost v nárocích na čas i paměť)
- Nezávislost na pořadí vstupu dat
- Interpretovatelnost výsledků
- Schopnost zvládat šum a přítomnost „outliers“
- Schopnost pracovat s různými typy dat
- Schopnost využít omezující podmínky uživatele
- …

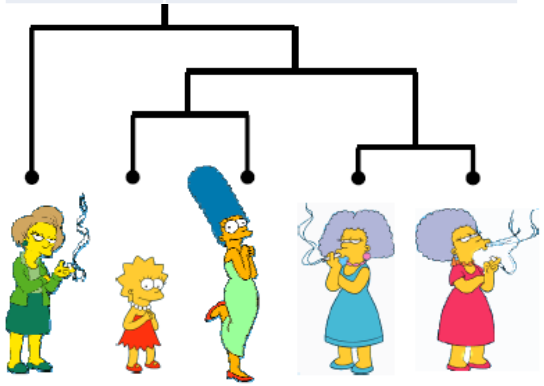
Osnova přednášky

- Motivace
- Míra vzdálenosti
- **Hierarchické shlukování**
- Hodnocení kvality rozkladu
- Shlukování rozkladem
  - k-means (k-středů)
  - EM (expectation maximization) algoritmus, Gaussovská směs
  - Odhad počtu shluků

Postupy pro hierarchické shlukování

Počet dendrogramů s ***n*** listy ? NP úloha $\rightarrow$ k řešení je nutná heuristika

# listů	# Dendrogramů
2	1
3	3
4	15
5	105
…	…
10	34 459 425



Postup zdola-nahoru (aglomerativní):

Začátek: každý objekt každý tvoří vlastní shluk.
Najdeme 2 nejbližší shluky, které sloučíme.

Proces opakujeme až do okamžiku, kdy všechny objekty jsou ve stejném shluku.

Postup shora-dolů (postupné dělení):

Začátek: jediný shluk tvořený všemi daty.
Otestujeme všechny možnosti, jak shluk rozdělit na 2 disjunktní části a vybereme nejlepší variantu.

Rekurzivně pokračujeme na obou vzniklých podmnožinách.

Předpokládejme, že máme k dispozici míru pro vzdálenost a údaje pro všechny páry, viz symetrická tabulka

$D($  $) = 8$

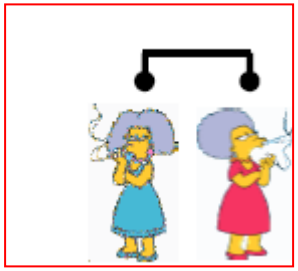
$D($  $) = 1$

0	8	8	7	7
	0	2	4	4
		0	3	3
			0	1
				0

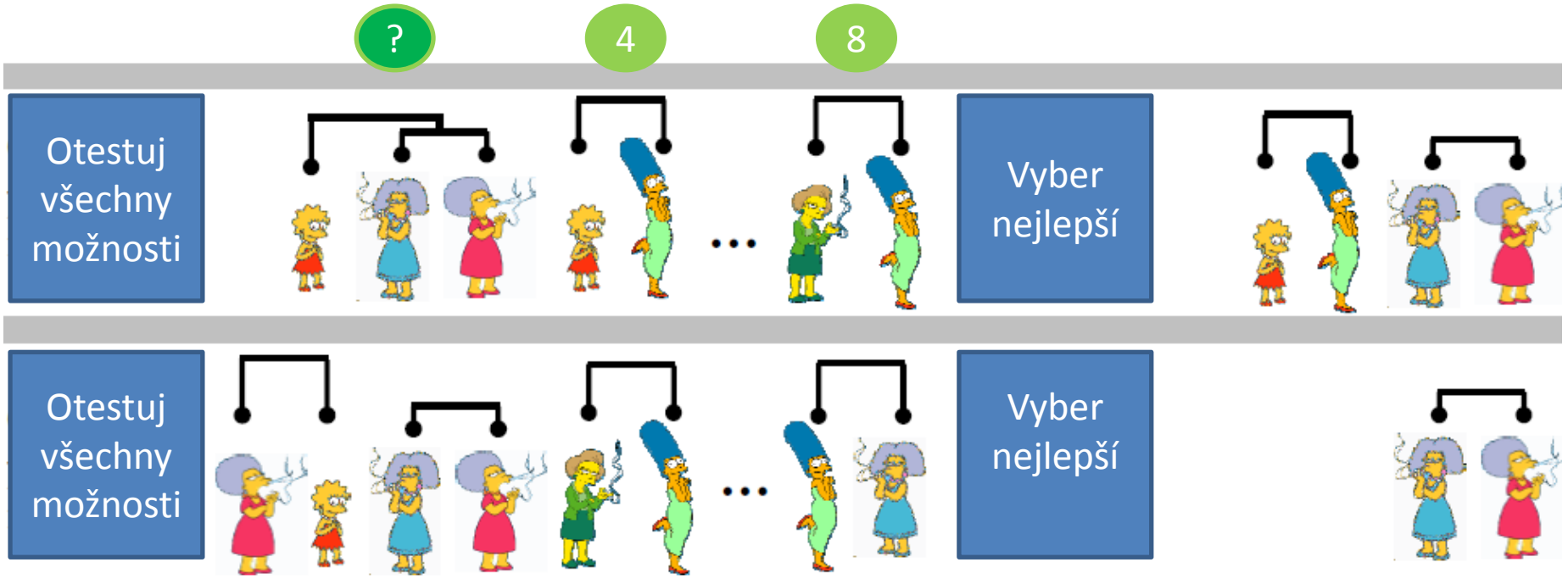
Aglomerativní postup zdola-nahoru: Simpsonovi-1

Otestujeme všech 5*4/2 párů a vybereme ten, jehož objekty jsou si nejblíž!

4 1 8 3

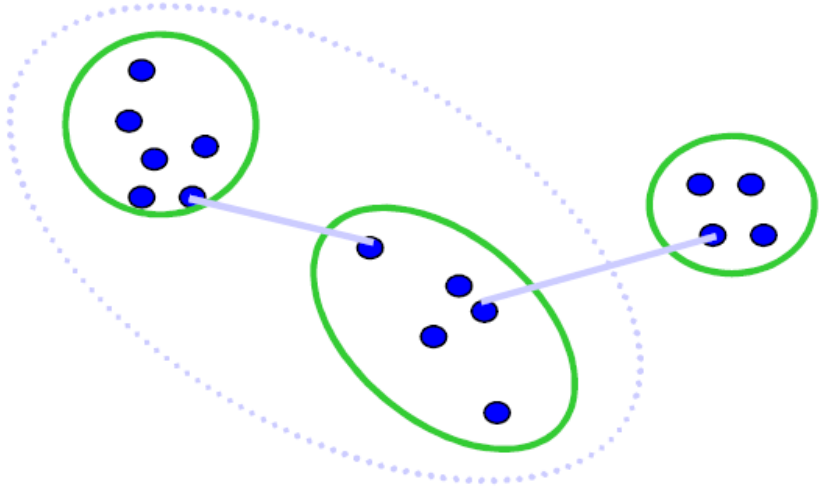


Simpsonovi-2



Určování vzdálenosti 2 shluků : míra1

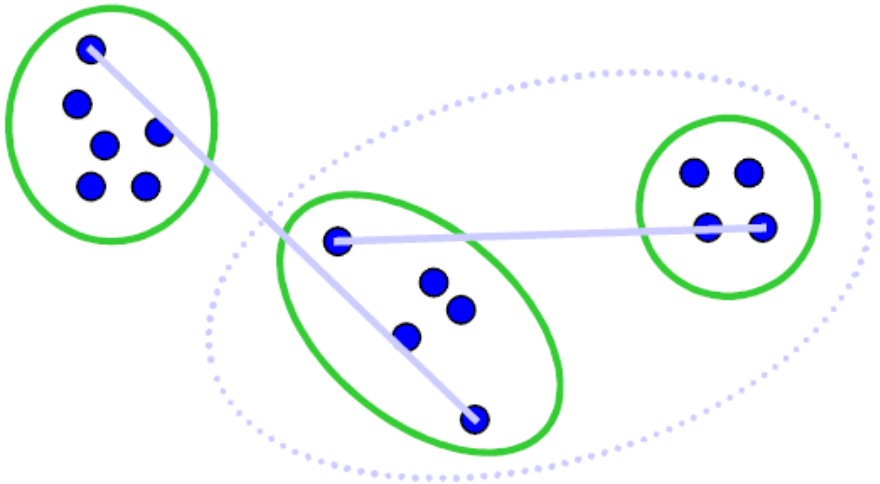
Vzdálenost shluků (*single link*) = vzdálenost jejich 2 **nejbližších prvků**



- Tato míra pro vzdálenost má tendenci k tvorbě řetízků menších shluků

Určování vzdálenosti 2 shluků: míra2

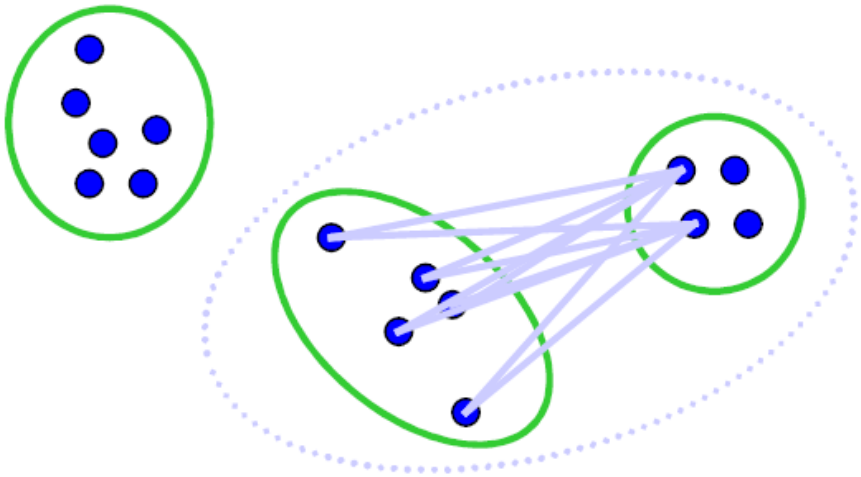
Vzdálenost shluků (complete link) = vzdálenost jejich 2 **nejvzdálenějších prvků**



- Tato míra pro vzdálenost obvykle tvoří poměrně kompaktní shluky

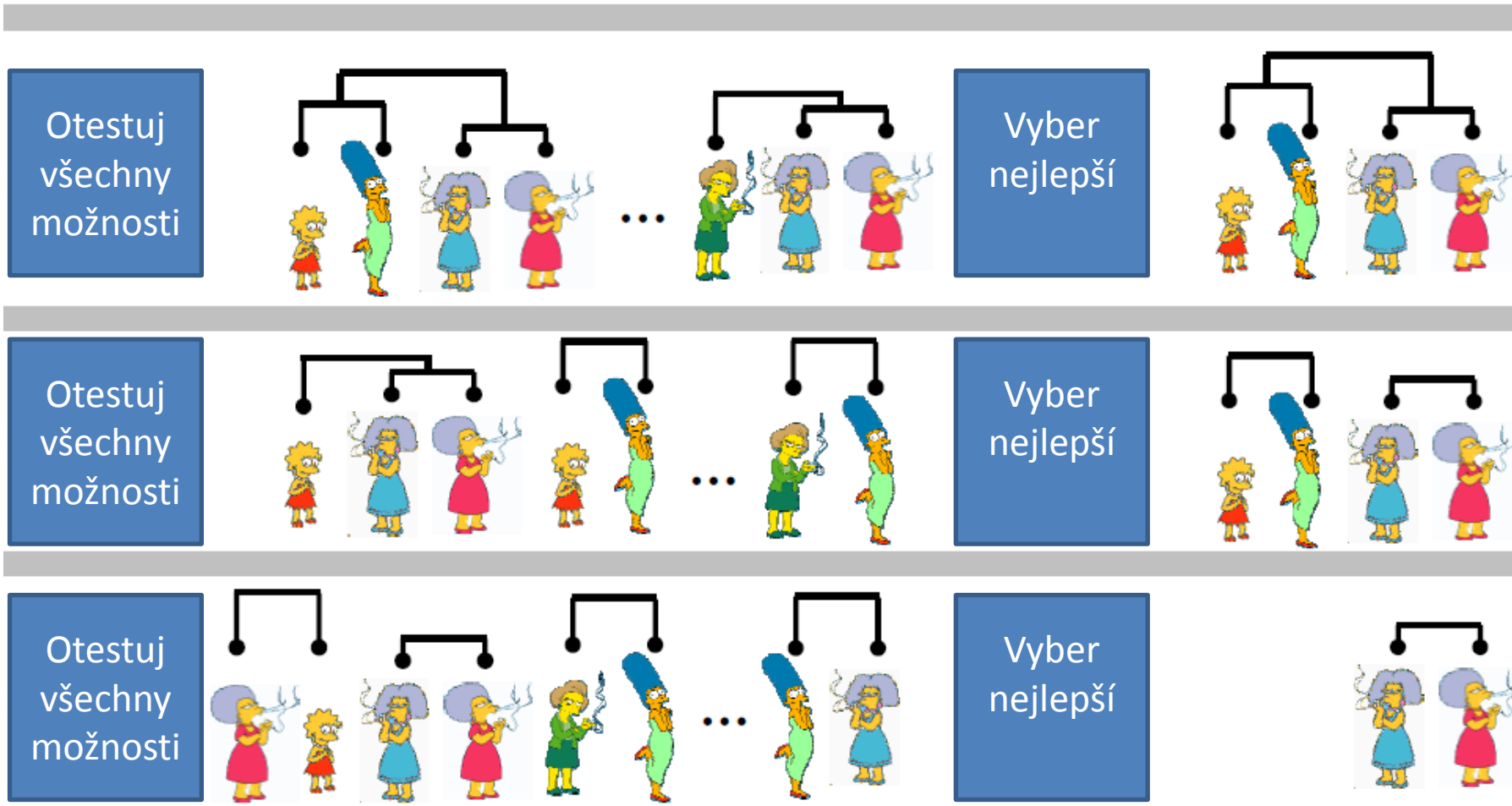
Určování vzdálenosti 2 shluků : míra3

Vzdálenost shluků = průměrná vzdálenost mezi **všemi prvky obou shluků**

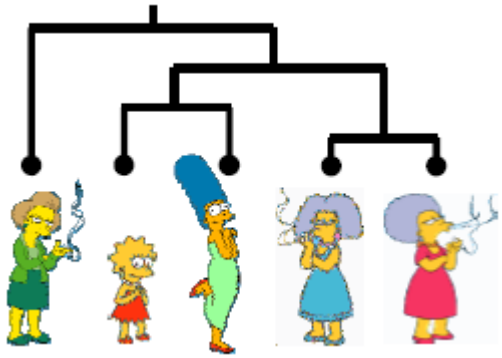


- Nejčastěji používaná míra pro vzdálenost.
- Robustní vůči šumu!

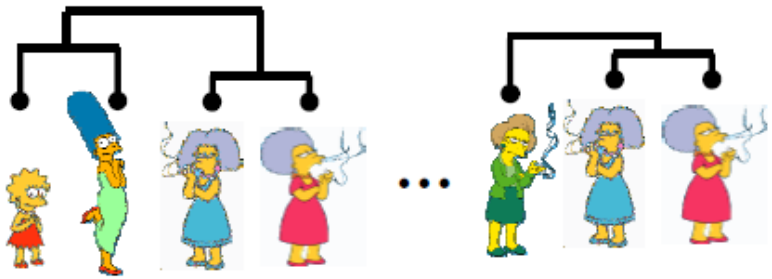
Simpsonovi-3



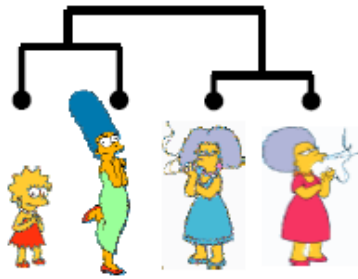
Simpsonovi-4



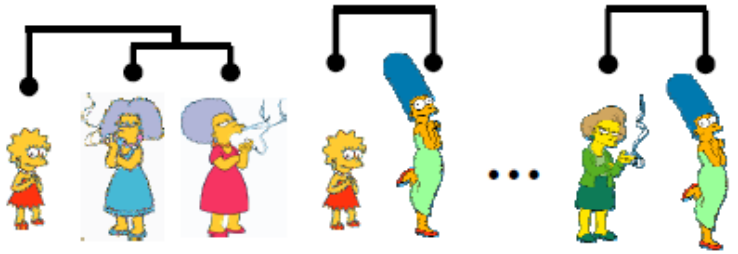
Otestuj všechny možnosti



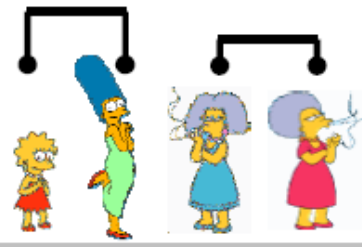
Vyber nejlepší



Otestuj všechny možnosti



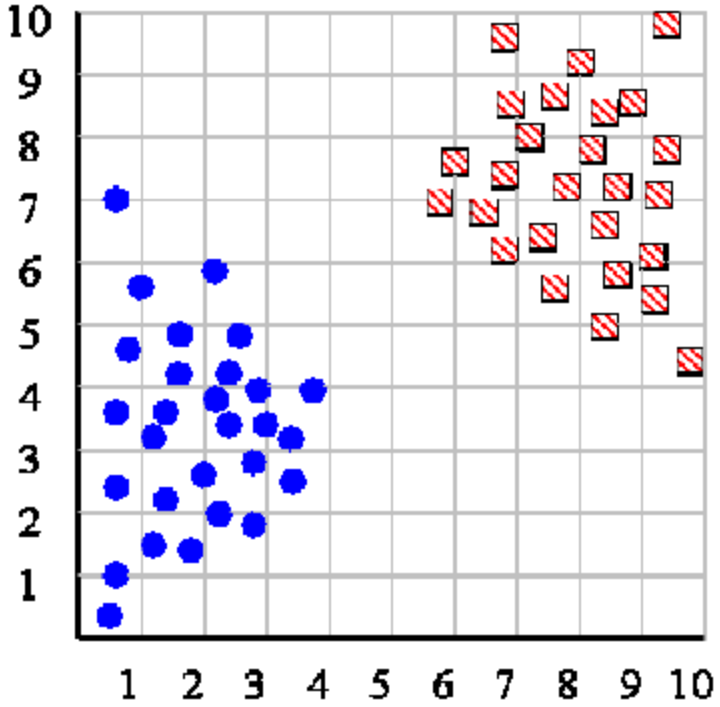
Vyber nejlepší

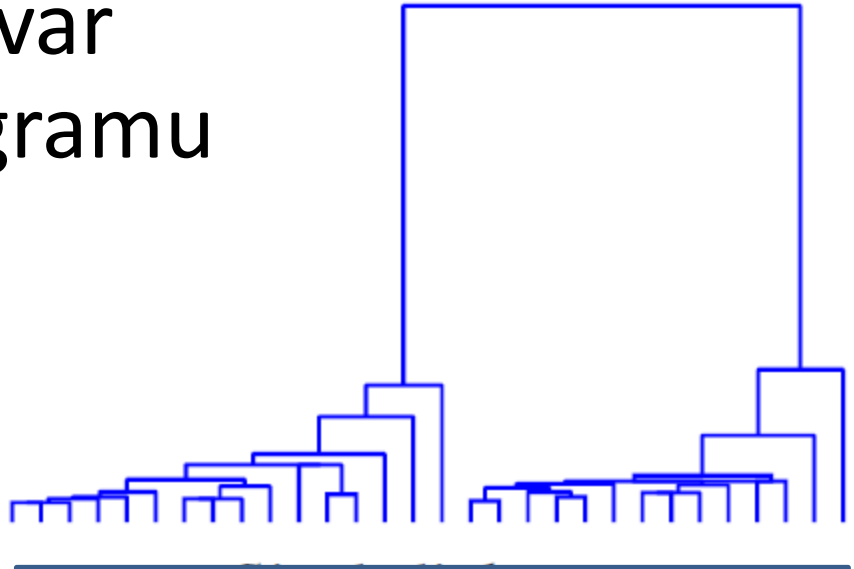


Otestuj všechny možnosti

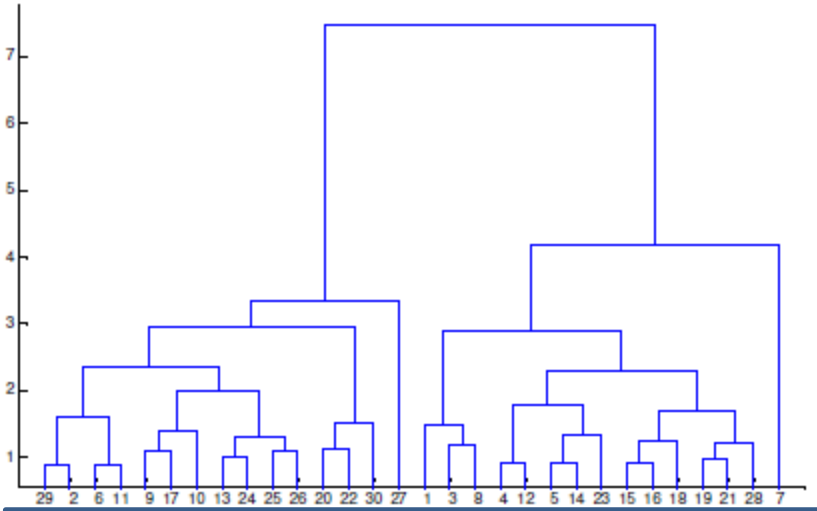
Vyber nejlepší

Vliv volby míry na tvar výsledný tvar dendrogramu





Vzdálenost: nejbližší sousedi (míra 1)



Vzdálenost: průměr všech (míra3)

Shrnutí: typické vlastnosti hierarchických metod shlukování

- **Výhoda**: není třeba předem specifikovat počet shluků
- Hierarchickou strukturu dokáže uživatel často dobře interpretovat – odpovídá „intuici“, ovšem jde pak o subjektivní pohled!

- Problém s rozsáhlými daty, neboť dolní odhad pro složitost shlukování je $\mathcal{O}(n^2)$, kde n je mohutnost shlukované množiny
- Nebezpečí uvíznutí v lokálním optimu

Osnova přednášky

- Motivace
- Míra vzdálenosti
- Hierarchické shlukování
- **Hodnocení kvality rozkladu**
- Shlukování rozkladem
  - ***k***-means (***k***-středů)
  - EM (expectation maximization) algoritmus, Gaussovská směs
  - Odhad počtu shluků

Dá se hodnotit kvalita vzniklého rozkladu?

Nechť shlukování navrhne pokrytí původní množiny pomocí K shluků C_1, C_2, … , C_K .Neexistuje universální definice pro “dobrý rozklad“. Jistě nejdůležitější je **hodnocení uživatele**. Přesto se používají i objektivní míry:

- **Vnitřní míra kvality**, např. *SEE*, t.j. suma odchylek uvnitř shluků (od pomyslného „těžiště“ shluku) přes všechny shluky:
  - Pro každý shluk i najdeme bod μ_i označený jako „těžiště“ shluku i jako průměr hodnot pro všechny prvky $\boldsymbol{x} \in C_i$
  - Pro každý shluk i vypočteme odchylku od těžiště uvnitř shluku $SE_i = \sum_{\boldsymbol{x} \in Ci} d(\boldsymbol{x} - \mu_i)$
  - $SEE = \sum_{i \leq K} SE_i$

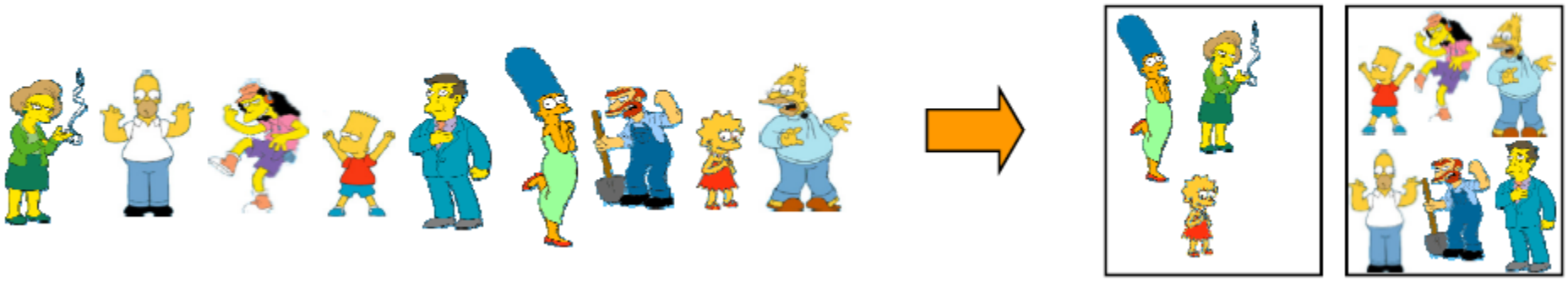
- **Vnější míra kvality** srovnává to, jak navržený rozklad odpovídá nějaké klasifikaci, která existuje pro nějké vybrané instance objektů.

Osnova přednášky

- Motivace
- Míra vzdálenosti
- Hierarchické shlukování
- Hodnocení kvality rozkladu
- **Shlukování rozkladem**
  - ***k***-means (***k***-středů)
  - EM (expectation maximization) algoritmus, Gaussovská směs
  - Odhad počtu shluků

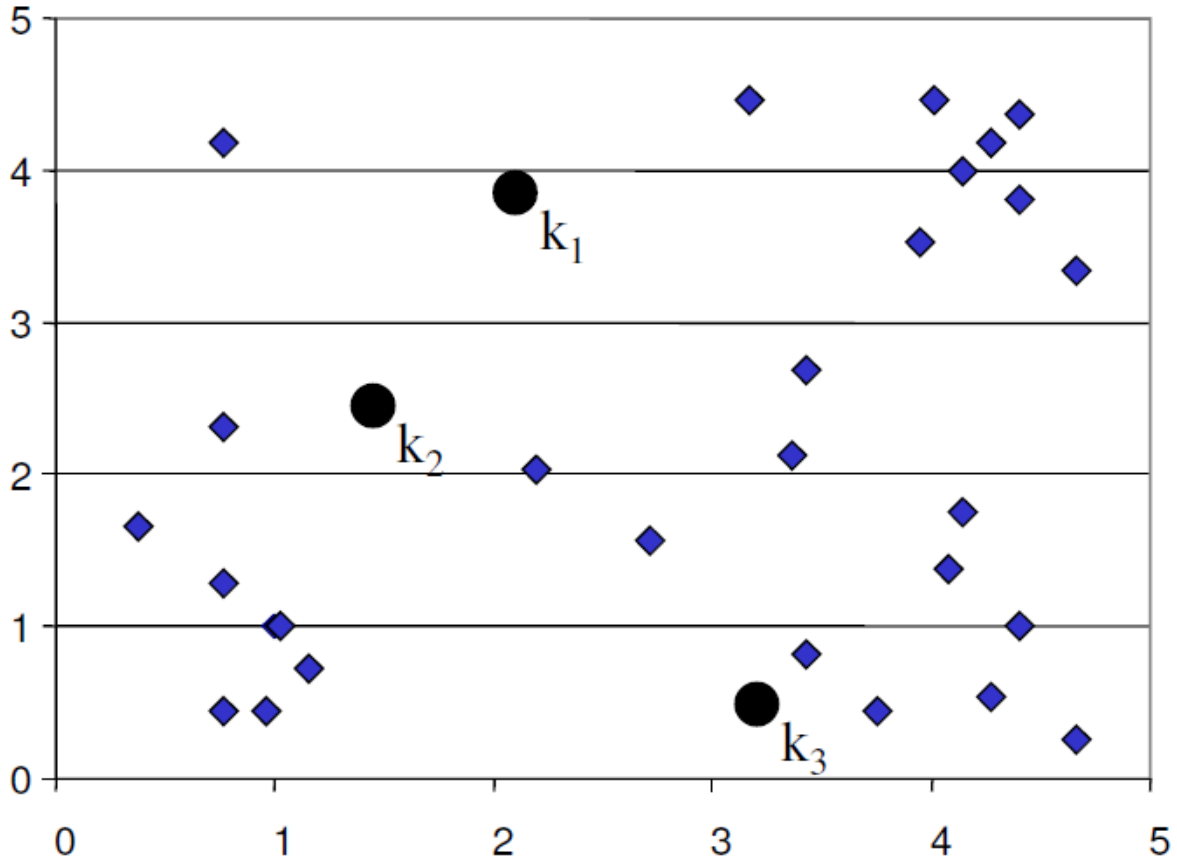
Shlukování rozkladem

- Nehierarchický postup, při němž se každý objekt vloží do jednoho z ***k*** disjunktních shluků.
- Předpokládá se, že uživatel předem stanoví ***k***, tj. požadovaný cílový počet shluků



Shlukování metodou K-jader (means): inicializace

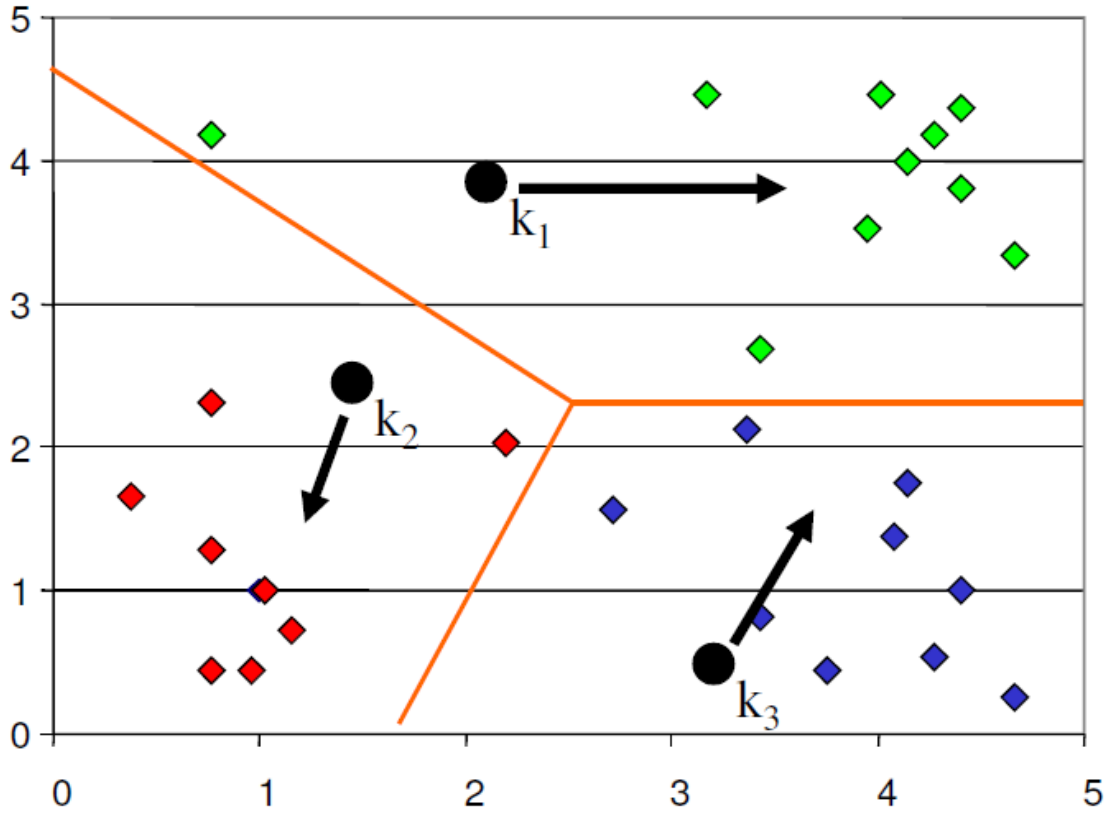
a. Stanov hodnotu ***k*** a vyber náhodně ***k*** bodů (jader) ve výchozí množině



K-means shlukování: iterační krok 1

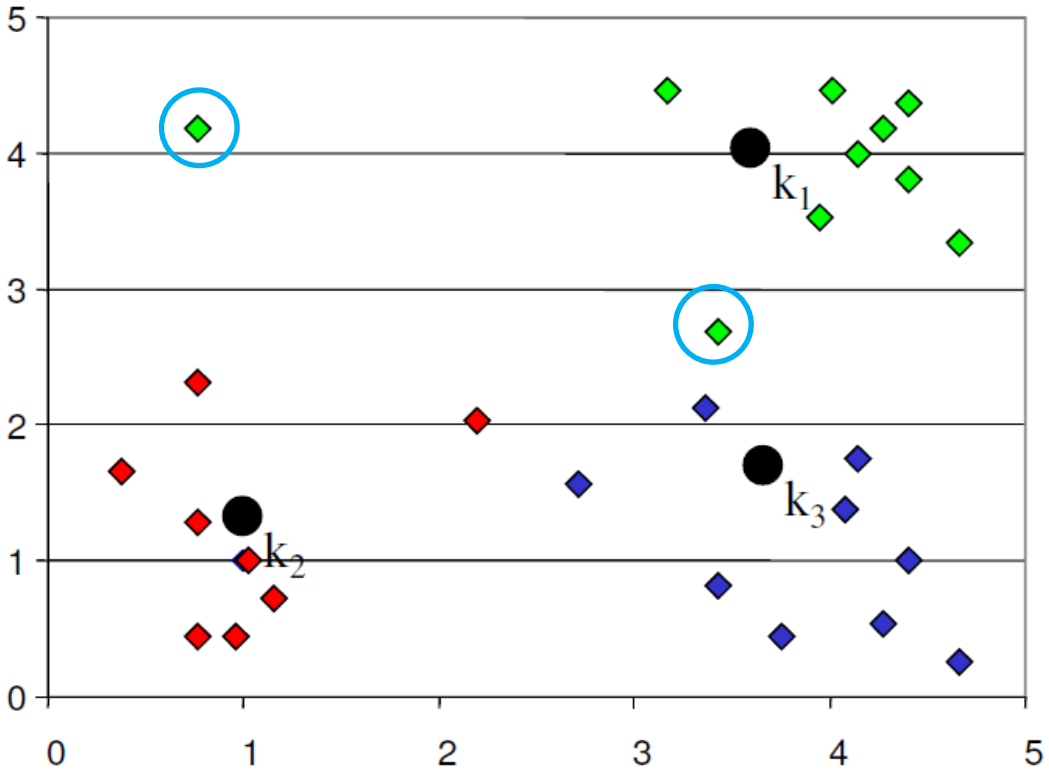
a. Přiřaď každý bod výchozí množiny k **nejbližšímu** z vybraných ***k*** jader.

b. V každé ze vzniklých ***k*** množin bodů nadefinuj **nové jádro** jako „průměr“ všech prvků této množiny



K-means shlukování: iterační krok 2

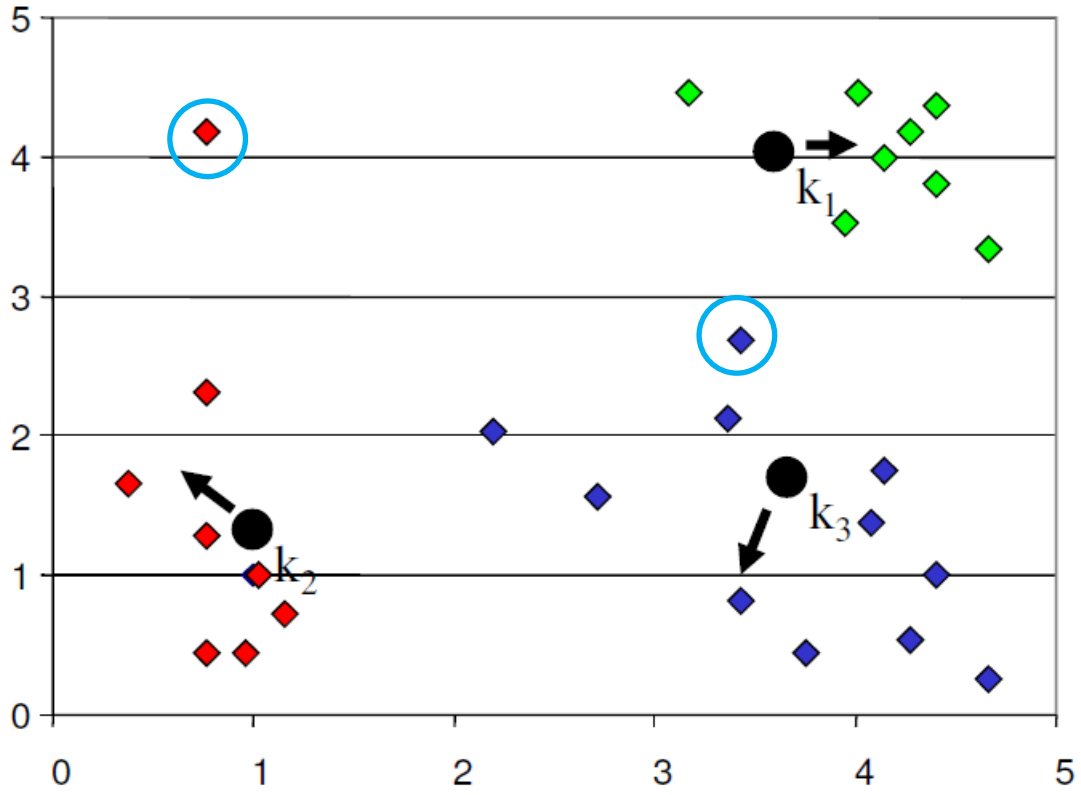
a. Pro každý bod původní množiny proveď přiřazení k nejbližšímu z jader **nadefinovaných v předchozím kroku**.
b. V každé ze vzniklých ***k*** množin bodů nadefinuj **nové jádro** jako „průměr“ všech prvků této množiny



Zařazení některých prvků se změnilo!

K-means shlukování: iterační krok 2

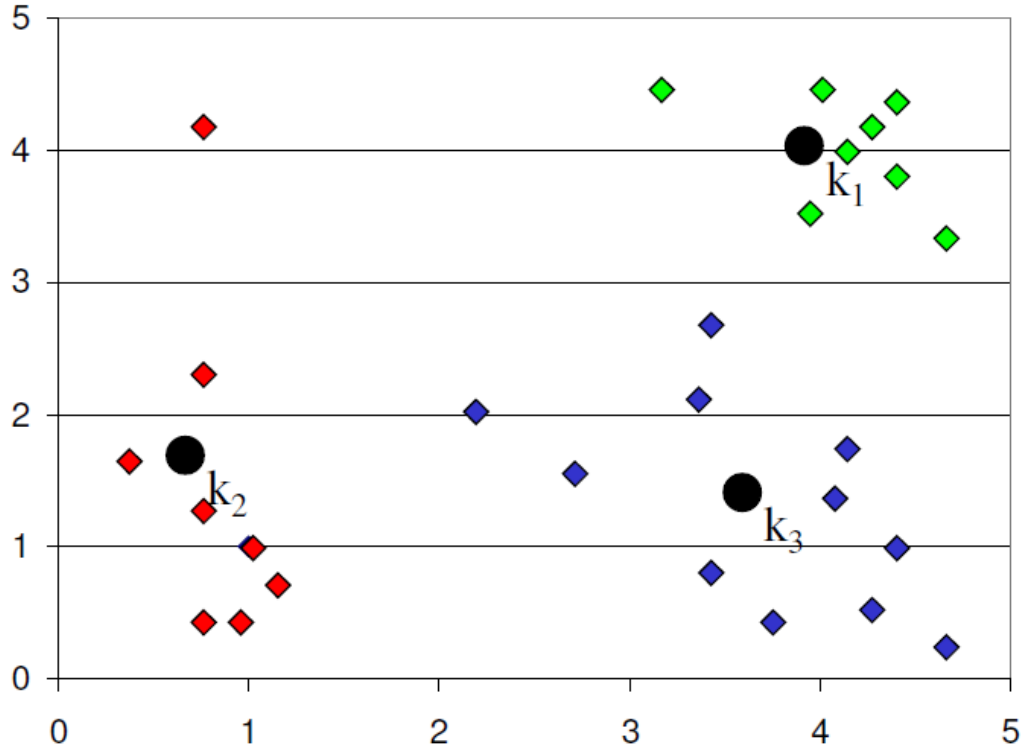
a. Pro každý bod původní množiny proveď přiřazení k nejbližšímu jádru z těch, která byla nadefinovaná v předchozím kroku.
b. V každé ze vzniklých ***k*** množin bodů nadefinuj **nové jádro** jako „průměr“ všech prvků této množiny



K-means shlukování: kriterium pro ukončení

a. Opakuj iterační krok až do té doby, než

„Při iteraci nedojde ke **změně zařazení do shluku** pro **žádný** prvek původní množiny.“



Algoritmus ***k***-means pro ***N*** objektů

1. Stanov požadovaný počet ***k*** shluků.
2. Vyber náhodně výchozích ***k*** jader.
3. Přiřaď každému z ***N*** objektů číslo shluku, které odpovídá číslu nejbližšího jádra.
4. Předefinuj pozice jader všech ***k*** shluků tak, že bude použit průměr hodnot prvků v daném shluku.
5. Opakuj kroky 3. a 4. až do situace, kdy se příslušnost do shluků stabilizuje (po iteraci není žádný objekt zařazen do jiného shluku než před ní).

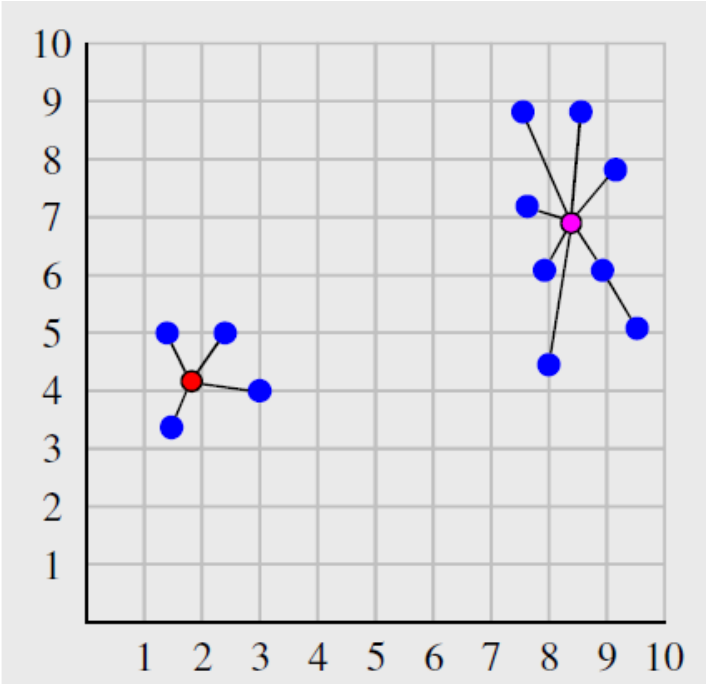
Proč algoritmus *k*-means funguje?

- **Předpoklad:** Dobré shlukování zajišťuje vysokou podobnost uvnitř shluku.
- ***K*-means** minimalizuje průměrnou vzdálenost mezi prvky téhož shluku vypočtenou jako

$$\sum_{k=1}^{K} \frac{1}{n_k} \sum_{i=1}^{n_k} \sum_{j=1}^{n_k} \left\| x_{ki} - x_{kj} \right\|^2$$

- Tato hodnota se rovná 2x suma vzdáleností ke středům jednotlivých shluků, čili výsledné střední chybě

$$\sum_{k=1}^{K} \sum_{i=1}^{n_k} \left\| x_{ki} - \mu_k \right\|^2$$



Shrnutí: vlastnosti *k*-means algoritmu

- **Výhody**
  - Jednoduchý (lehká implementace i ladění).
  - Intuitivní objektivní funkce, která optimalizuje podobnost uvnitř shluků.
  - *Poměrně efektivní*: složitost $\mathcal{O}(T * K * m * N)$, kde m je počet objektů, K je počet shluků, N počet atributů a T počet iterací. Obvykle bývají hodnoty t a $k \ll m$.
- **Nevýhody**
  - Použitelné, jen tam, kde *umíme spočítat průměr*. Co kategorická data?
  - Velmi záleží na inicializaci – nebezpečí uvíznutí v *lokálním minimu*.
  - Požaduje se znalost *počtu shluků*.
  - Nevhodné pro zašuměná data s *výjimkami* (outliers).
  - Nevhodné pro situace, kdy *shluky nemají konvexní tvar* .

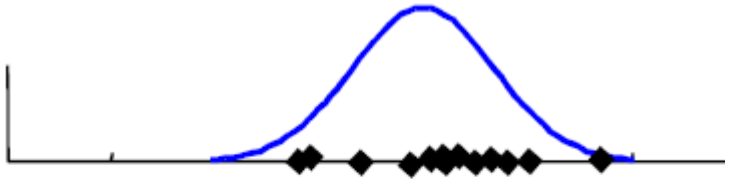
Osnova přednášky

- Motivace
- Míra vzdálenosti
- Hierarchické shlukování
- Hodnocení kvality rozkladu
- Shlukování rozkladem
  - k-means (k-středů)
  - **EM (expectation maximization) algoritmus, Gaussovská směs**
  - Odhad počtu shluků

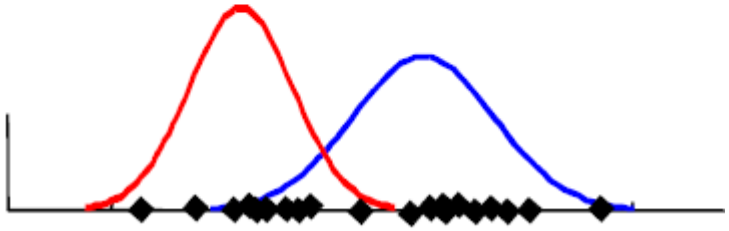
Jednorozměrný model typu **GMM** „Gaussovská směs“

- Gaussian

$$P(x) = \varphi(x;\mu,\sigma) = \frac{1}{\sigma\sqrt{2\pi}}\exp(-\frac{(x-\mu)^2}{2\sigma^2})$$



  - např. výška populace



- Gaussovská směs

$$P(C=i) = \omega_i, \quad P(x \mid C=i) = \varphi(x;\mu_i,\sigma_i)$$

$$P(x) = \sum_{i=1}^{K} P(x, C=i) = \sum_{i=1}^{K} P(C=i)P(x \mid C=i) = \omega_i\varphi(x;\mu_i,\sigma_i)$$

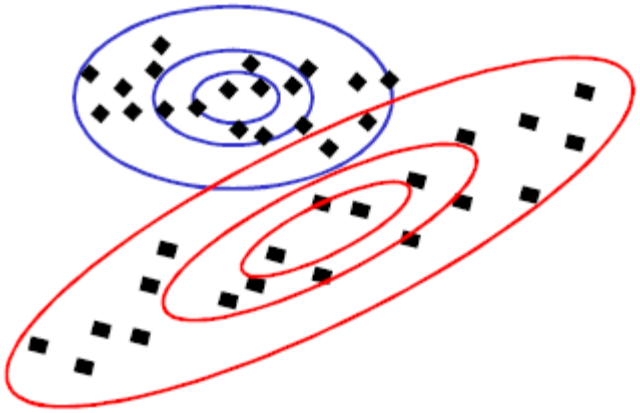
  - např. výška 2 různých populací

Vícerozměrný model typu **GMM** „Gaussovská směs“

- Směs vícerozměrných Gaussiánů

$$P(C=k)=\omega_i, \quad P(x \mid C=i)=\varphi(x;\mu_i,\Sigma_i)$$

  - Např. pro situaci, kdy **y** je hodnota krevního tlaku a **x** je věk



GMM + EM = „Soft k-means“

1. Stanov požadovaný počet ***k*** shluků.
2. Vyber náhodně výchozích ***k*** jader.
3. **E-krok**: přiřaď pravděpodobnostní hodnotu příslušnosti ke shlukům

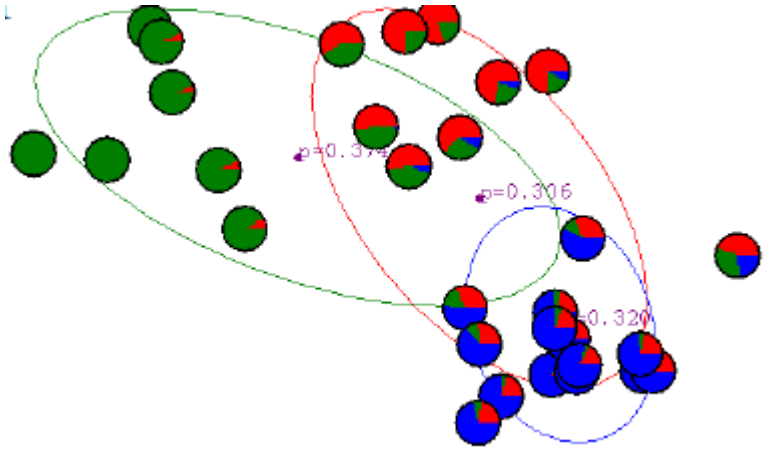
   $p_{ij} = \boldsymbol{P}(C = i \mid \boldsymbol{x}_j) = \alpha\, \boldsymbol{P}(\boldsymbol{x}_j \mid C = i)\, \boldsymbol{P}(C = i)$

   $p_i = \sum_{\boldsymbol{j}} p_{ij}$
4. **M-krok:** proveď nové odhady parametrů s využitím právě vypočtených hodnot. / $^{\mathsf{T}}$

   $\mu_i \leftarrow \sum_{\boldsymbol{j}} p_{ij} \boldsymbol{x}_j / p_i$

   $\sum_{\boldsymbol{j}} \leftarrow \sum_{\boldsymbol{j}} p_{ij} \boldsymbol{x}_j \boldsymbol{x}_j^{\mathsf{T}} / p_i$

   $\omega_i \leftarrow p_i$
4. Opakuj kroky 3. a 4. až do té doby, kdy změny všech parametrů jsou menší než zvolená hranice.



Hodnocení GMM

Výhody

- Interpretovatelnost: vzniká dokonce generativní model!
- Efektivita srovnatelná s ***k***-means
- Intuitivní objektivní funkce
- Lze zobecnit i pro směsi různých typů dat:
  - Kategorická data
  - Místo průměru lze použít např. max.
  - Citlivost na šum a výjimky záleží na distribuční funkci

Nevýhody

- Často uvázne v lokálním optimu – vliv inicializace!
- Je nutné správně zvolit hodnotu ***k.***
- Nevhodné v případě, že shluky nejsou konvexní!
- $\mathcal{O}(m^3)$

Srovnání metod shlukování

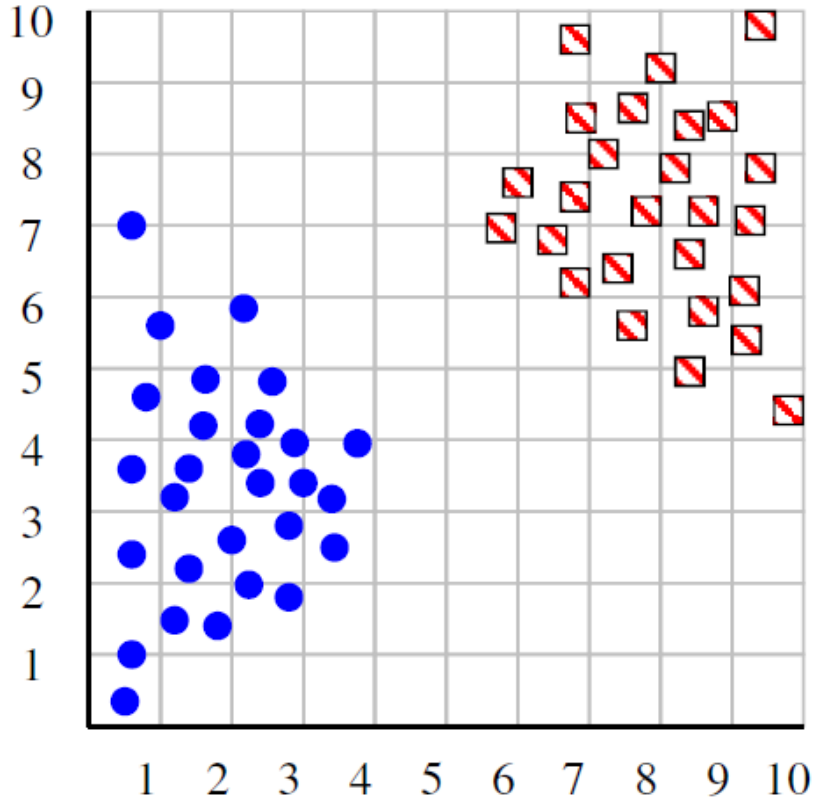
	Hierarchické	K-means	GMM
Časové nároky	$\mathcal{O}(m^3)$	nejrychlejší	rozumně rychlé
Předpoklady	Je třeba mít míru podobnosti/ vzdálenosti	+ další silné předpoklady (konvexní shluky, ..)	Nejsilnější předpoklady
Vstupní parametry	žádné	Pevně zvolený parametr ***k*** (počet shluků)	Pevně zvolený parametr ***k*** (počet shluků)
Navržené shluky	Vzniká pouze strom, který lze subjektivně interpretovat	Přesně ***k*** shluků	Přesně ***k*** shluků

Osnova přednášky

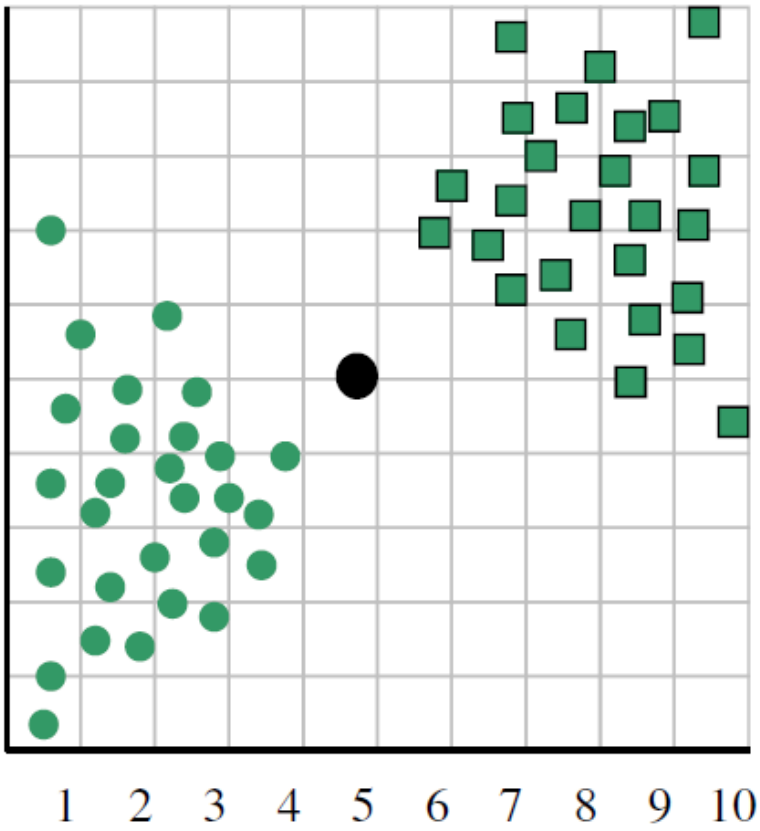
- Motivace
- Míra vzdálenosti
- Hierarchické shlukování
- Hodnocení kvality rozkladu
- Shlukování rozkladem
  - k-means (k-středů)
  - EM (expectation maximization) algoritmus, Gaussovská směs
  - **Odhad počtu shluků**

Jak se rozhodnout pro správný počet shluků?

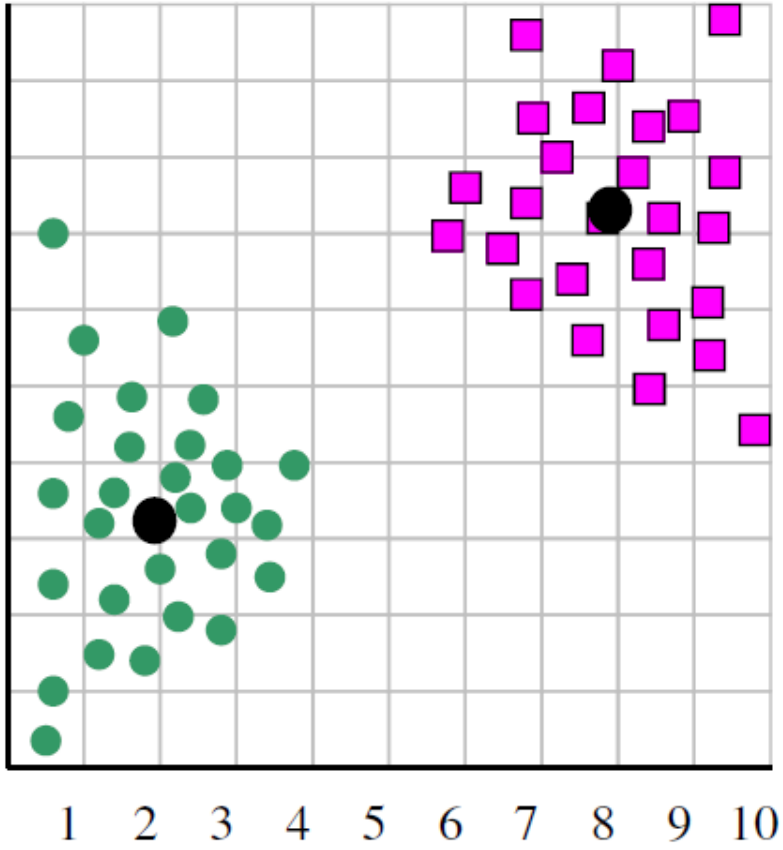
- Obecně jde o otevřený problém.
- Používá se řada heuristik
- Jedna z nich srovnává hodnoty objektivní funkce (=celkový součet vzdáleností uvnitř shluků) pro různé volby počtu shluků



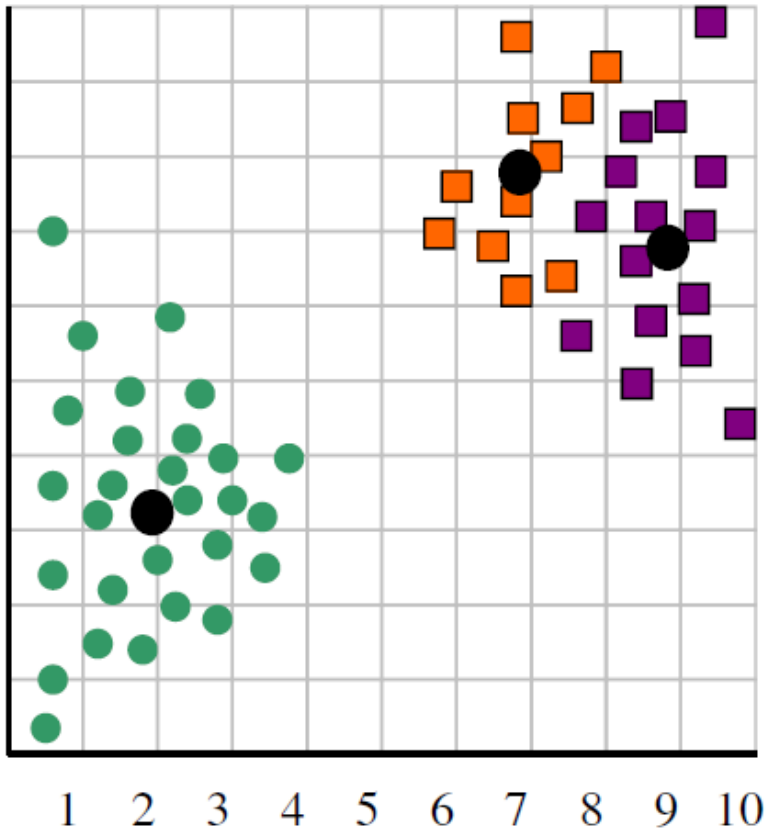
- Je-li ***k*** = 1, je výsledná hodnota objektivní funkce 873



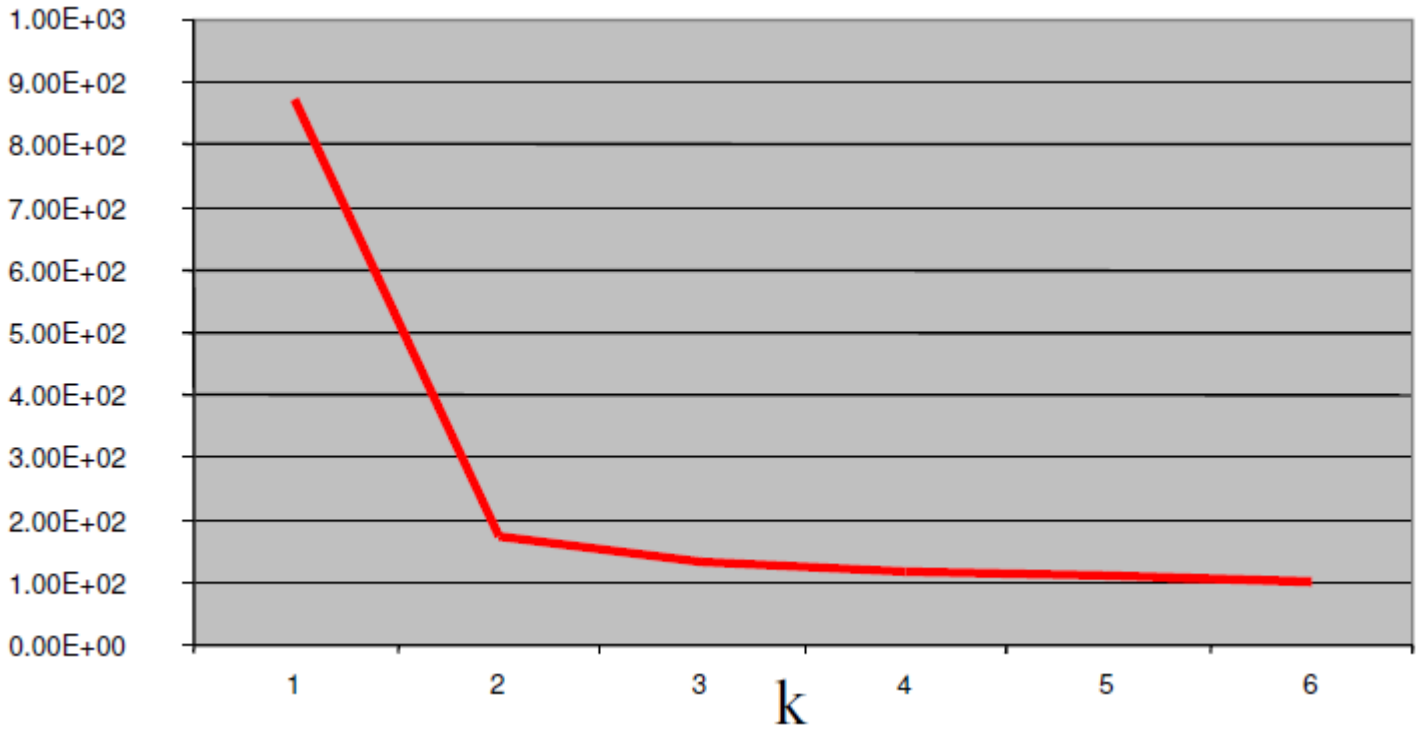
- Je-li ***k*** = 2, je výsledná hodnota obj. funkce 173,1



- Je-li ***k*** = 3, je výsledná hodnota obj. funkce 133,6



- Sledujme hodnotu objektivní funkce pro ***k***= 1,2,..,6
- Náhlý pokles pro ***k*** = 2 svědčí pro volbu 2 shluků.
- Obecně hledáme „prudký ohyb“ (knee/elbow finding)



Pozor! Průběh objektivní funkce obvykle není tak jednoduchý jako v tomto umělém příkladě.