

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)
 - Odhad počtu shluků
 - * EM (expectation maximization) algoritmus, Gaussovská směs

Co je to shlukování?

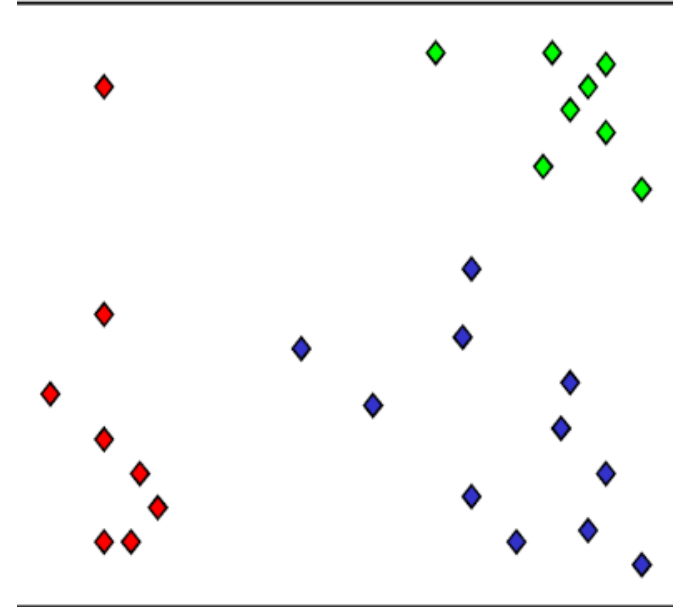
Seskupení dat do takových **shluků**, že podobnost dat

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

Proč nás to zajímá?

Hledáme „přirozené“ seskupení, které

- podporuje interpretaci dat,
- dá se to použít pro něco užitečného.



Důležité pro nalezení vnitřní struktury dat: Linného systém, Mendělejevova tabulka, personalizované nabízení zboží, **segmentace obrazů** jako základ pro rozpoznávání objektů či definici hranic, ...

Clusty/Yippy umožňuje lepší orientaci ve výsledcích webového vyhledávače,

„ .. **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.”

MOTIVACE

Co jsou přirozené shluky z těchto individuí?

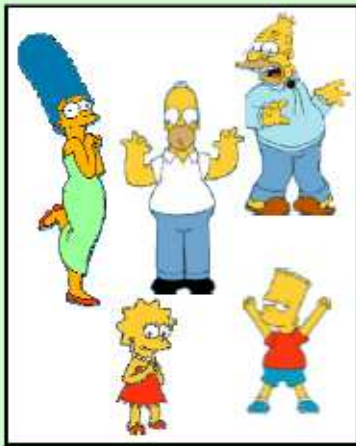
Shlukování je chápáno jako deskriptivní úloha, jejímž cílem je charakterizovat takové kategorie individuí, které mají smysl a které umožní lépe chápat pozorovaný svět.



Co jsou „přirozené“ shluky z těchto individuí?



Pojem „přirozenost“ má zde velmi subjektivní charakter!



Rodina Simpsonů



Zaměstn. školy



Ženy



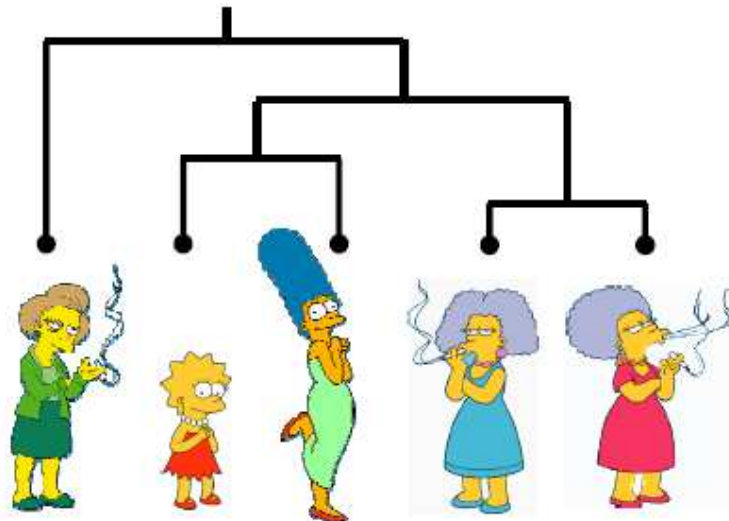
Muži

Dva přístupy ke tvorbě shluků

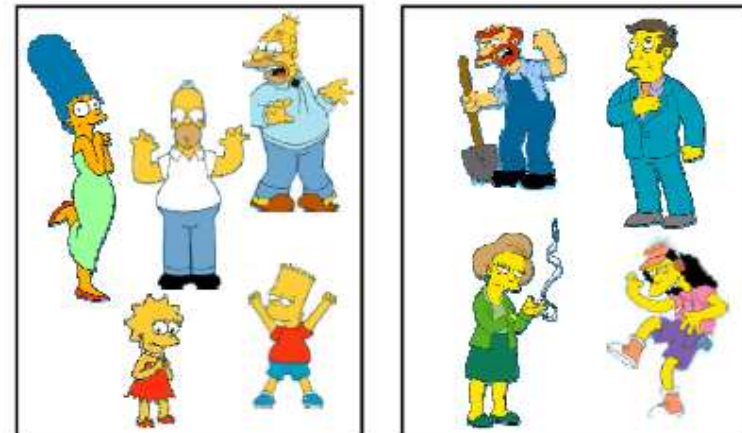
Shlukování rozkladem konstruuje různé rozklady množiny uvažovaných objektů a z těch vybírá nejvhodnější vzhledem k nějakému kritériu.

Hierarchické shlukování postupně sdružuje uvažované objekty podle zvoleného kritéria (míra podobnosti)

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 $o1$ a $o2$ přiřazuje reálné číslo $D(o1,o2)$ tak, aby pro libovolné objekty A,B a C měla funkce D tyto vlastnosti

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

Platí-li navíc ještě následující vlastnost

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

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

Nejčastější míry vzdálenosti

Minkowského míra pro objekty popsané **atributy s reálnými hodnotami**

- Necht' objekt je popsán p atributy jako reálný vektor. Uvažujme objekty $\mathbf{x} = (x_1, x_2, \dots, x_p)$ a $\mathbf{y} = (y_1, y_2, \dots, y_p)$

Minkowského metrická vzdálenost je definována

$$d(\mathbf{x}, \mathbf{y})^n = (|x_1 - y_1|^n + |x_2 - y_2|^n + \dots + |x_p - y_p|^n)$$

Kde n je parametr volený podle potřeb aplikace, např.:

- Je-li $n = 1$, mluvíme o **Manhattanské vzdálenosti**
- Je-li $n = 2$, jedná se **Eucleidovu vzdálenost**

Nejčastější míry vzdálenosti

Uvažujme **binární vektory** s p atributy, tedy objekty typu $\mathbf{x} = (x_1, x_2, \dots, x_p)$ a $\mathbf{y} = (y_1, y_2, \dots, y_p)$, kde x_i až y_p mají hodnoty 0 nebo 1.

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

Míra pro objekty s binárními atributy vychází z kontingenční tabulky. V případě, že obě hodnoty (0 i 1) mají stejnou důležitost, říkáme, že **atribut** je **symetrický**, jinak je **asymetrický**.

– Jsou-li **všechny atributy symetrické**, pak

$$d(\mathbf{x}, \mathbf{y}) = (r+s) / (q+r+s+t)$$

– Jsou-li **všechny atributy asymetrické** a výsledek 1 je významnější než 0, pak se nezapočítává shoda v hodnotách 0:

$$d(\mathbf{x}, \mathbf{y}) = (r+s) / (q+r+s)$$

Nejčastější míry vzdálenosti

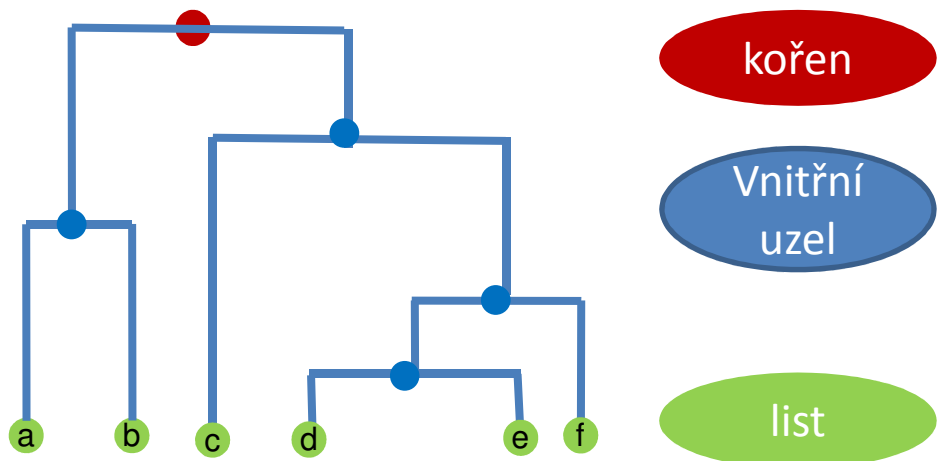
Nechť objekt je popsán jako vektor o p attributech s **nominálními hodnotami**, tj. hodnoty mají výčtový typ (např. *barvy* nebo *jména měst*). Uvažujme objekty $\mathbf{x} = (x_1, x_2, \dots, x_p)$ a $\mathbf{y} = (y_1, y_2, \dots, y_p)$, jejichž hodnoty se přesně **shodují na m pozicích**.

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

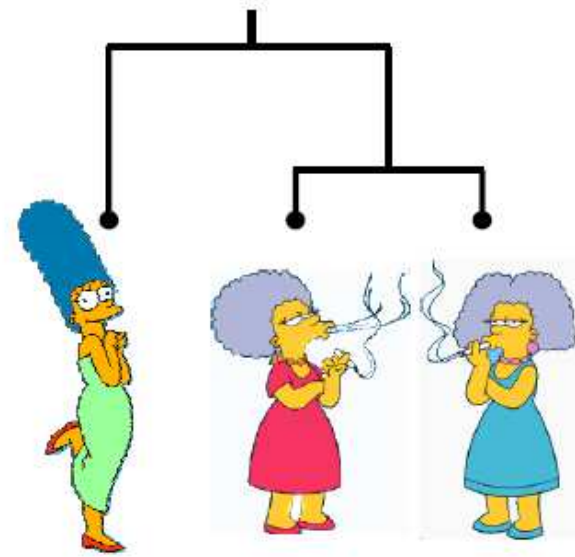
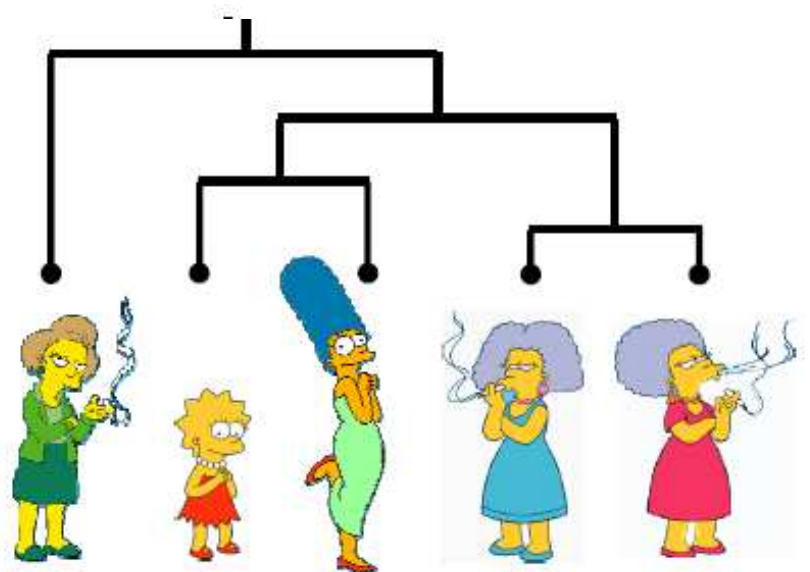
Je-li objekt popsán p atributy jako vektor s **ordinálními hodnotami**, tj. jde o konečný definiční obor, jehož hodnoty jsou uspořádány (např. *kvalitativní hodnoty* ... opatrně u *čísel měsíců* !), používá se míra podobná jako u reálných hodnot (původní hodnota je nejprve transformována do intervalu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

Pokud **různé atributy mají různé typy**, míra vzdálenosti vznikne jako **součet měr po příslušných typech** – pozor na normalizaci!

Dendrogram jako užitečný nástroj pro kompaktní znázornění vztahů podobnosti ve skupině



Vzdálenost 2 objektů a a b je v dendrogramu vyjádřena výškou (= vzdáleností od listu ke kořeni) nejmenšího podstromu, který obsahuje objekty a i b : Nejbliž k sobe jsou d a e , blízko k nim je i f . O trochu dál než skupina $\{d, e, f\}$ jsou od sebe a a b . Vzdálenost mezi c a f je větší než mezi a a b , ještě vzdálenost je však např. mezi b a f .



Jaké vlastnosti by měl mít algoritmus pro tvorbu shluků?

- **Škálovatelnost** vzhledem k rozsahu dat = **polynomiální** (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

Hierarchické shlukování zdola nahoru

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

$$D(\text{Mrs. Krabappel, Lisa Simpson}) = 8$$











$$D(\text{Marge Simpson, Mrs. Gribble}) = 1$$

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

Aglomerativní postup zdola-nahoru: Simpsonovi-1

$$D(\text{Lisa Simpson, Bart Simpson}) = 8$$

$$D(\text{Marge Simpson, Maggie Simpson}) = 1$$

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











Otestujeme všech $5 \cdot 4 / 2$ párů a vybereme ten, jehož objekty jsou si nejbliž!



Simpsonovi-2

$$D(\text{Marge}, \text{Lisa}) = 8$$

$$D(\text{Marge}, \text{Bart}) = 1$$

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

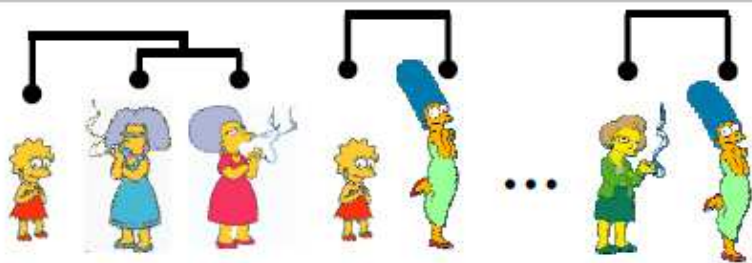
?

2

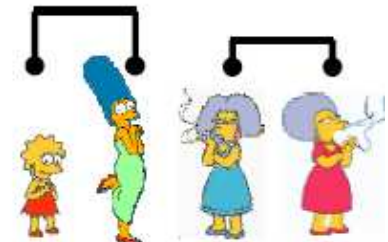
8



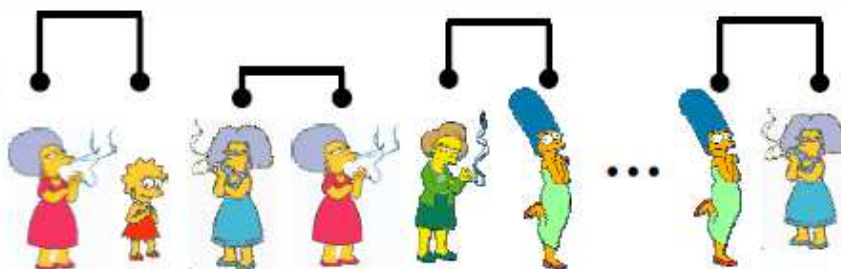
Otestuj všechny možnosti



Vyber nejlepší



Otestuj všechny možnosti

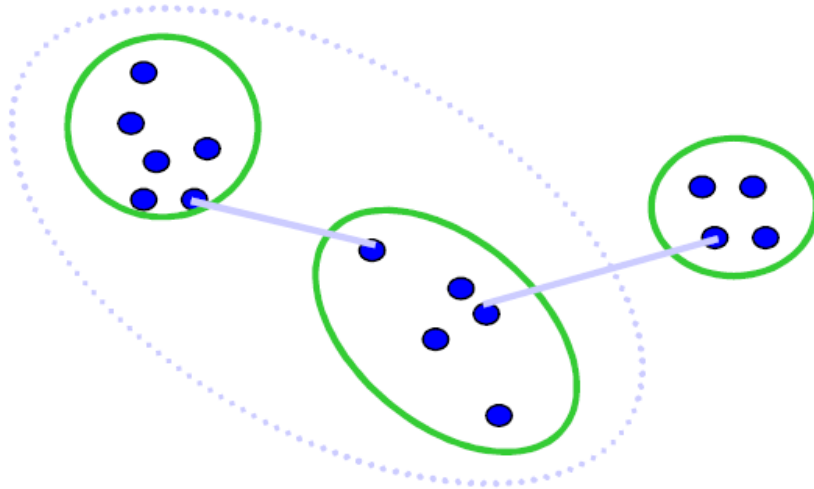


Vyber nejlepší



Určování vzdálenosti 2 shluků : míra1

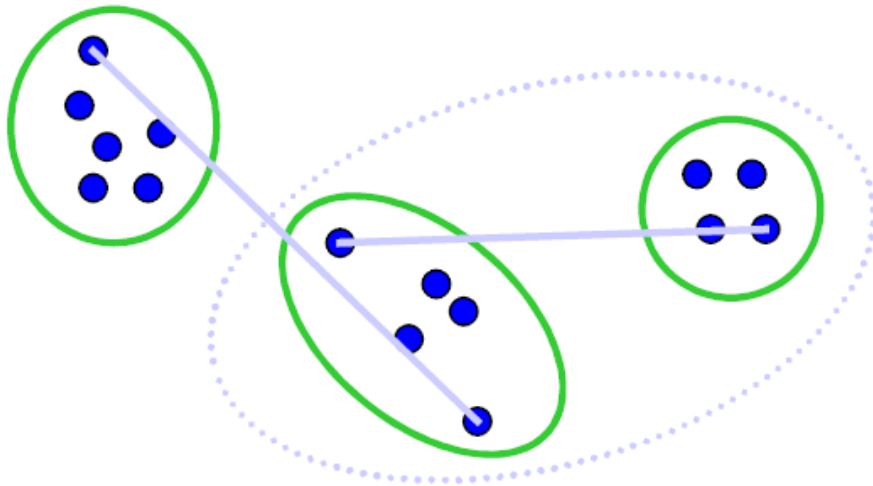
**Vzdálenost shluků (*single link*) = vzdálenost jejich 2
nejbližších prvků**



Tato míra pro vzdálenost
má tendenci tvořit řetízky
menších shluků

Určování vzdálenosti 2 shluků: míra2

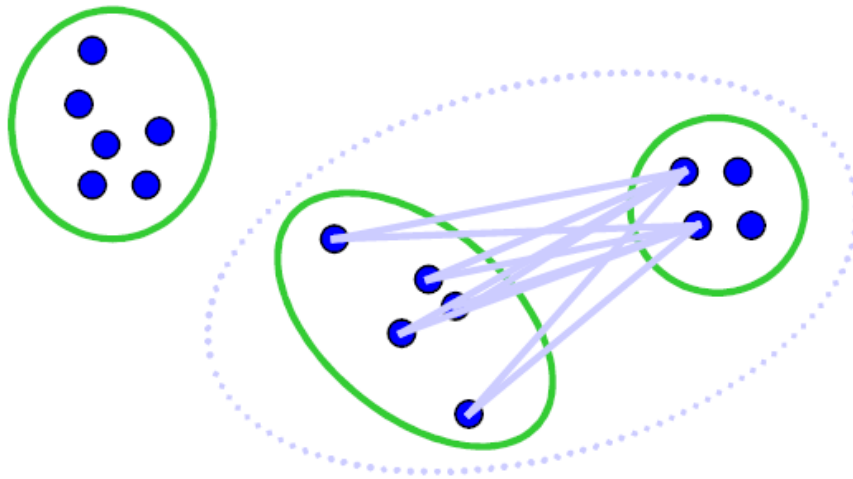
Vzdálenost shluků (complete link) = vzdálenost jejich 2 nejvzdálenějších prvků



Tato míra pro vzdálenost obvykle tvoří poměrně kompaktní shluky

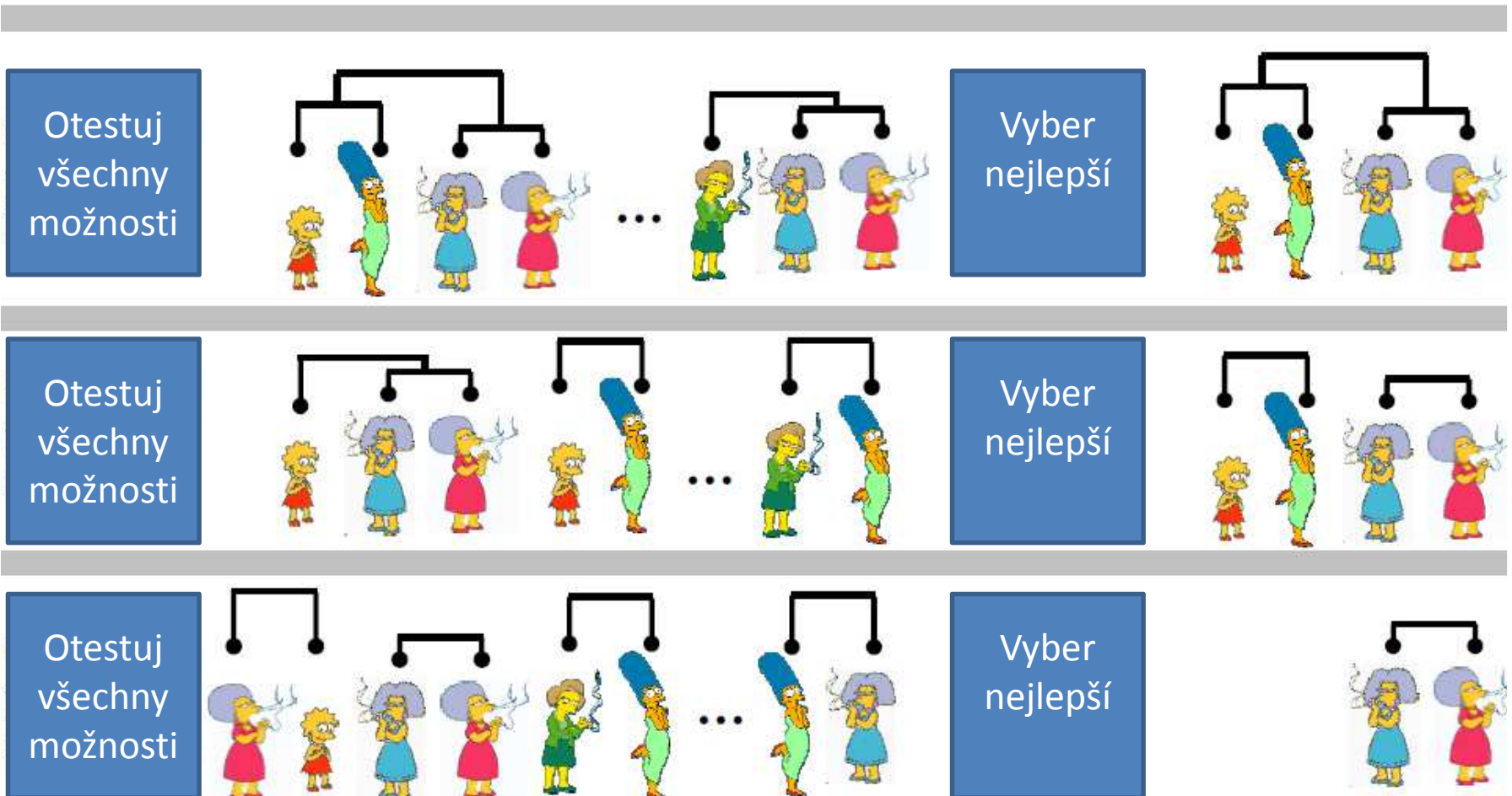
Určování vzdálenosti 2 shluků : míra3

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

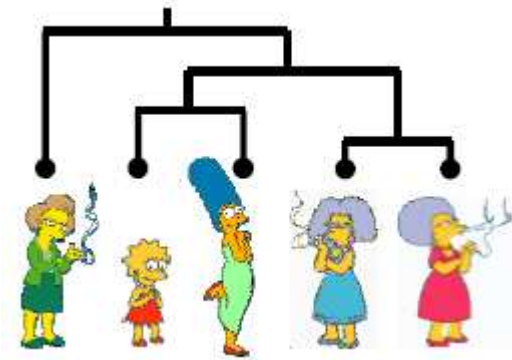


- Nejčastěji používaná míra pro vzdálenost dvou shluků.
- Robustní vůči šumu!

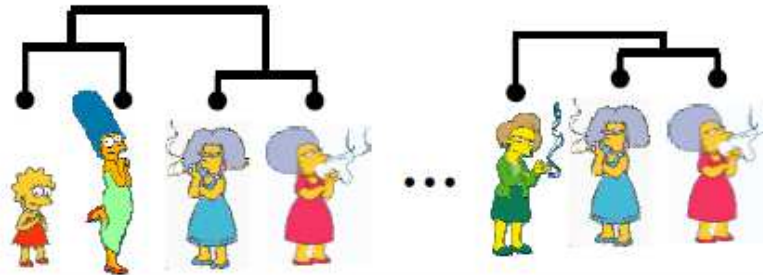
Simpsonovi-3



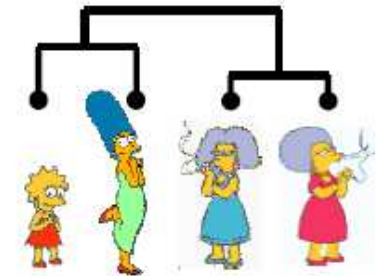
Simpsonovi-4



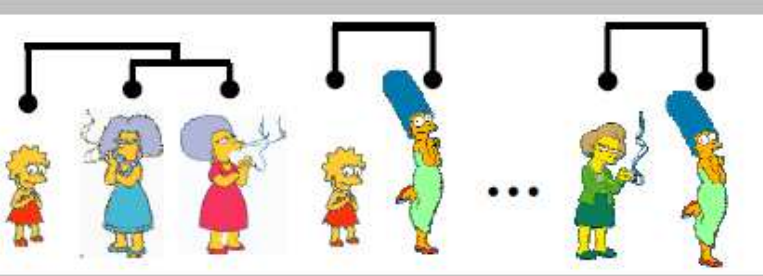
Otestuj
všechny
možnosti



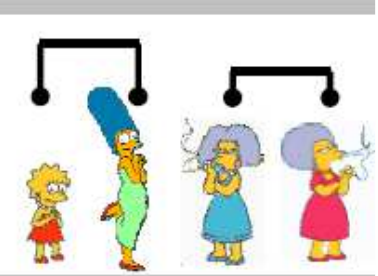
Vyber
nejlepší



Otestuj
všechny
možnosti



Vyber
nejlepší



Otestuj
všechny
možnosti



Vyber
nejlepší

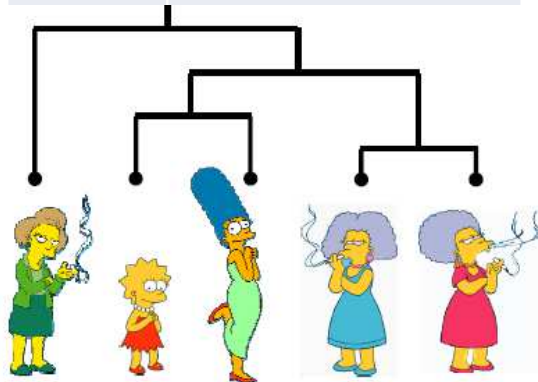


Hierarchické shlukování: nejdůležitější vlastnosti

$d(m)$: počet dendrogramů s m listy

$$d(m+1) = (m+1) * m * d(m) / 2$$

# listů	# Dendrogramů
2	1
3	3
4	18
5	180
...	...
10	Asi 40 000 000



$d(m)$ má složitost vyšší než $\mathcal{O}(m!)$. Prohlédnutí všech kandidátů je NP úloha → k řešení je nutná heuristika (např. „aglomerativní postup vede k cíli“)

Postup zdola-nahoru (aglomerativní):

Začátek: každý objekt tvoří vlastní shluk.

Najdeme 2 nejbližší shluky, které sloučíme.

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

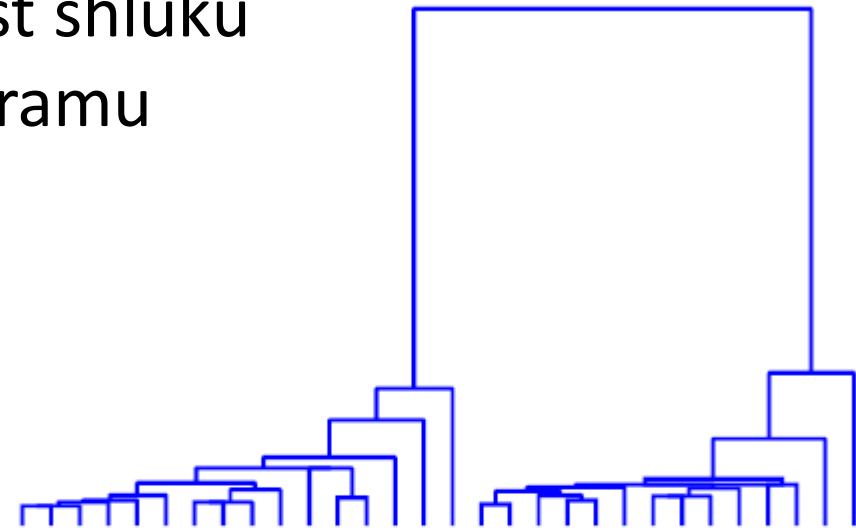
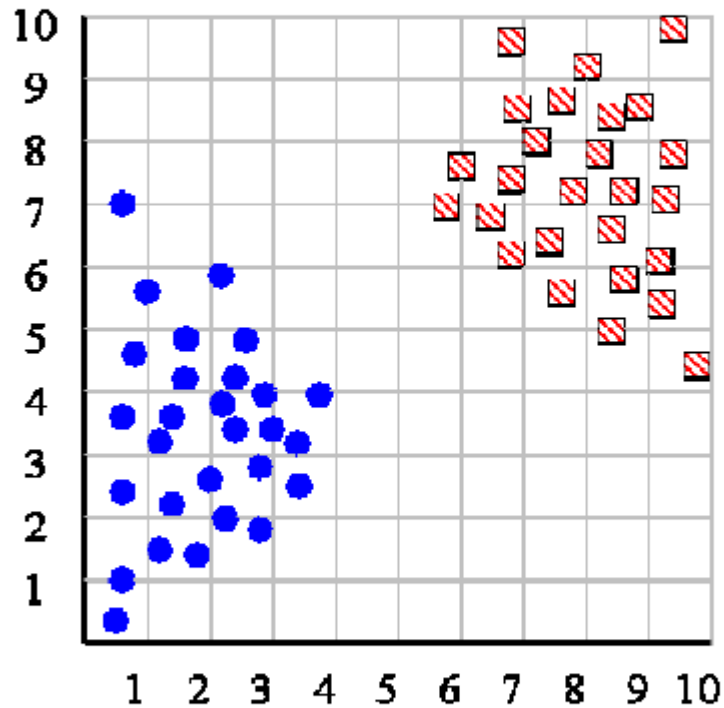
Postup shora-dolů (postupné dělení):

Začátek: jediný shluk tvořený všemi daty.

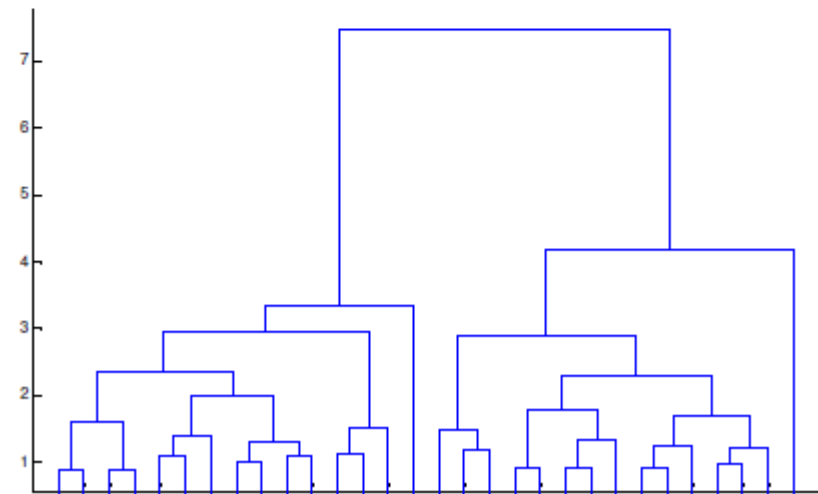
Otestujeme všechny možnosti, jak shluk rozdělit na 2 disjunktní části a vybereme nejlepší variantu.

Postup: Rekurzivně pokračujeme na obou vzniklých podmnožinách.

Vliv volby míry pro vzdálenost shluků na výsledný tvar dendrogramu



Vzdálenost: nejbližší sousedi (míra 1)



Vzdálenost: průměr všech (míra 3)

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 $O(m^3)$, kde m je mohutnost shlukované množiny (počet jejích prvků)
- **Nebezpečí** uvíznutí v lokálním optimu

1.Krok	m^2
2.Krok	$(m - 1)^2$
3.Krok	$(m - 2)^2$
4.Krok	$(m - 3)^2$
$m-1$.Krok	$(m - (m - 1))^2$

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, \dots, 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 $\mathbf{x} \in C_i$
- Pro každý shluk i vypočteme odchylku od těžiště uvnitř shluku (*squared error*)

$$SE_i = \sum_{\mathbf{x} \in C_i} d(\mathbf{x} - \mu_i)$$

$$SEE = \sum_{i \leq K} SE_i$$

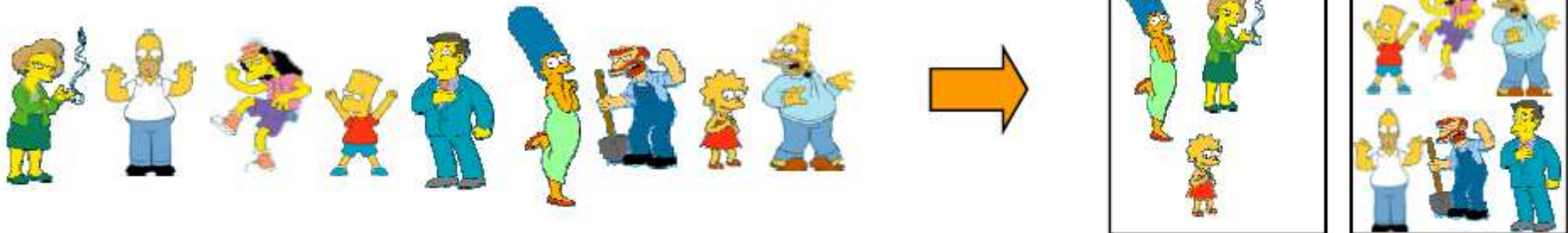
Vnější míra kvality srovnává to, jak navržený rozklad odpovídá nějaké klasifikaci, která již existuje pro uvažované vybrané instance objektů.

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

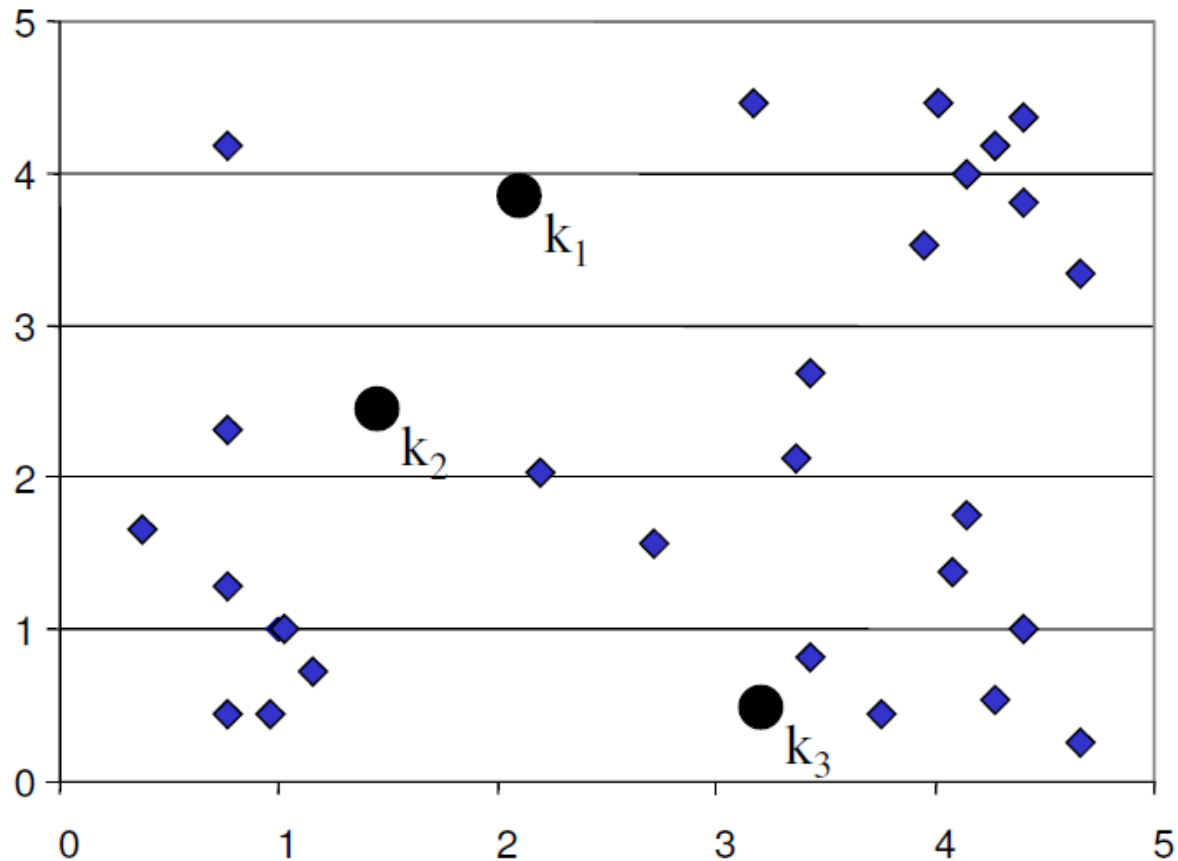
Shlukování rozkladem

- **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



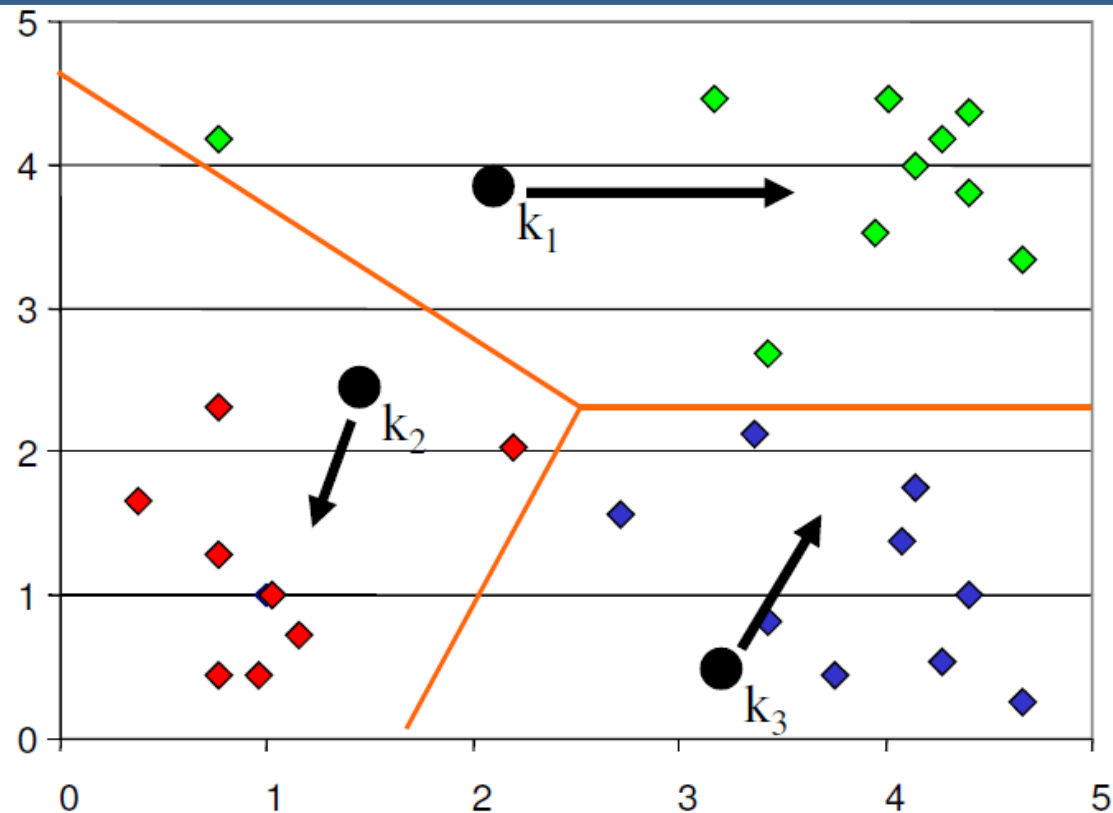
Shlukování metodou K-jader (means): krok 0 - inicializace

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



K-means shlukování: iterační krok 1

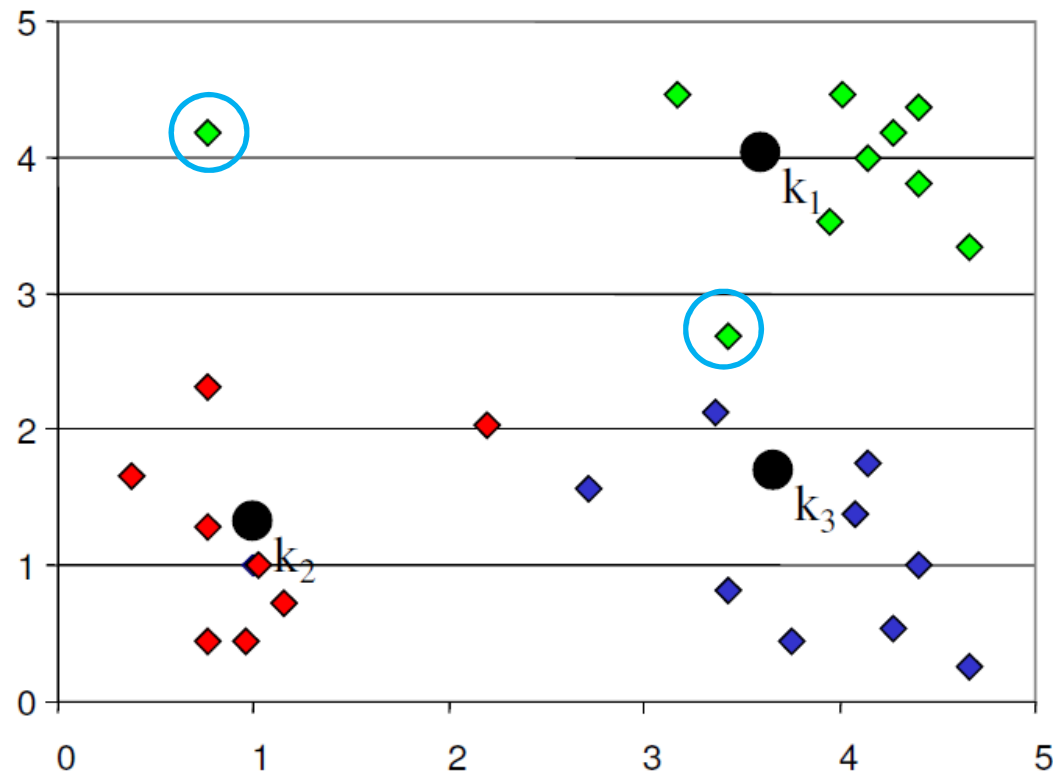
- Přiřaď každý bod výchozí množiny k **nejbližšímu** z vybraných k jader.
- V každé ze vzniklých k množin bodů **nadefinuj nové jádro** jako „průměr“ všech prvků této množiny.



K-means shlukování: iterační krok 2a

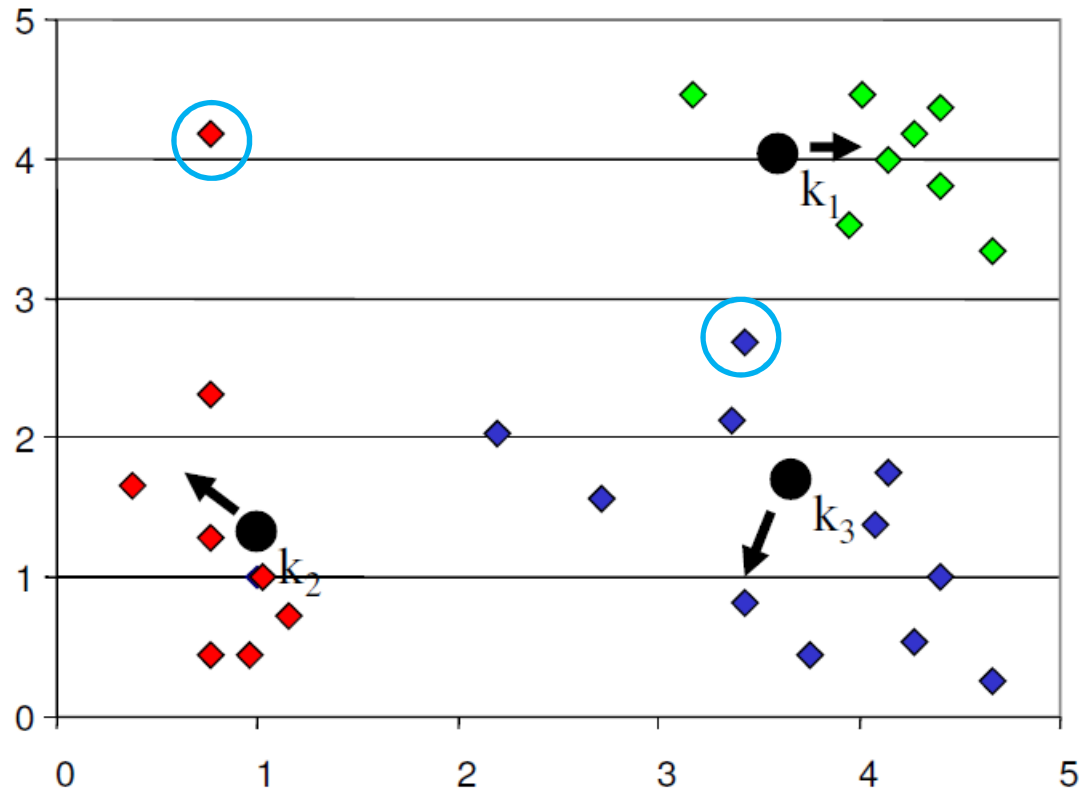
[A] Pro každý bod původní množiny proved'

- Přiřazení k nejbližšímu z **nových** jader (viz předchozí krok).
- Porovnání příslušnosti mezi původním a novým jádrem.



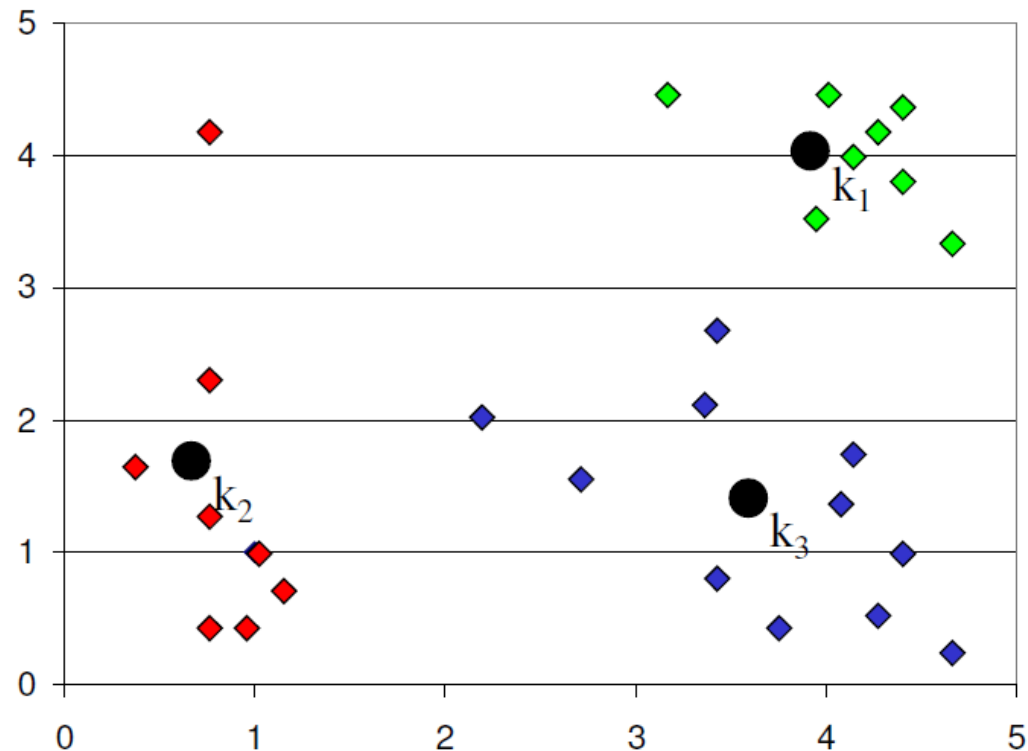
K-means shlukování: iterační krok 2b

- Pokud** se zařazení do shluků u žádného z bodů celé vstupní množiny nezměnilo, **KONEC**.
- Jinak** v každé ze vzniklých k množin bodů nadefinuj nové jádro jako „průměr“ všech prvků této množiny a SKOK na $[A]$.



K-means shlukování: kritérium pro ukončení

Opakuje se 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 jméno (souřadnice) toho jádra, kterému je tento objekt nejbliž.
4. Předefinuj pozice jader všech k shluků tak, že bude použit průměr hodnot prvků v daném shluku (a tuto záměnu proved' i v záznamech provedených v bodě 3).
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).

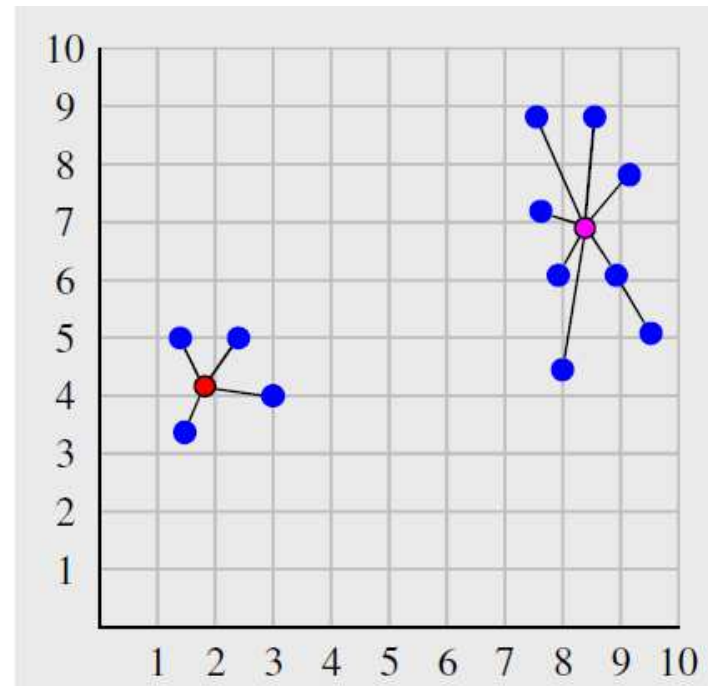
Proč algoritmus *k*-means funguje?

- **Předpoklad:** Dobré shlukování zajišťuje vysokou podobnost uvnitř shluku.
- Nechť n_k je mohutnost (počet prvků) shluku k . *K*-means minimalizuje průměrnou vzdálenost mezi prvky téhož shluku vypočtenou jako

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

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

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



$$\begin{aligned} \sum_{i \leq n_k} \sum_{j \leq n_k} (\mathbf{x}_i - \mathbf{x}_j)^2 &= \\ &= \sum_{i \leq n_k} \sum_{j \leq n_k} ((\mathbf{x}_i - \boldsymbol{\mu}_k) - (\mathbf{x}_j - \boldsymbol{\mu}_k))^2 = \\ &= \sum_{i \leq n_k} \sum_{j \leq n_k} ((\mathbf{x}_i - \boldsymbol{\mu}_k)^2 + (\mathbf{x}_j - \boldsymbol{\mu}_k)^2 - 2(\mathbf{x}_i - \boldsymbol{\mu}_k) * (\mathbf{x}_j - \boldsymbol{\mu}_k)) = \\ &= \sum_{i \leq n_k} (n_k (\mathbf{x}_i - \boldsymbol{\mu}_k)^2 + \sum_{j \leq n_k} (\mathbf{x}_j - \boldsymbol{\mu}_k)^2) - 2(\mathbf{x}_i - \boldsymbol{\mu}_k) * (\sum_{j \leq n_k} (\mathbf{x}_j - \boldsymbol{\mu}_k)) = \\ &= 2n_k \sum_{j \leq n_k} (\mathbf{x}_j - \boldsymbol{\mu}_k)^2 - 2(\sum_{j \leq n_k} (\mathbf{x}_j - \boldsymbol{\mu}_k)) * (\sum_{j \leq n_k} (\mathbf{x}_j - \boldsymbol{\mu}_k)) \end{aligned}$$

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**

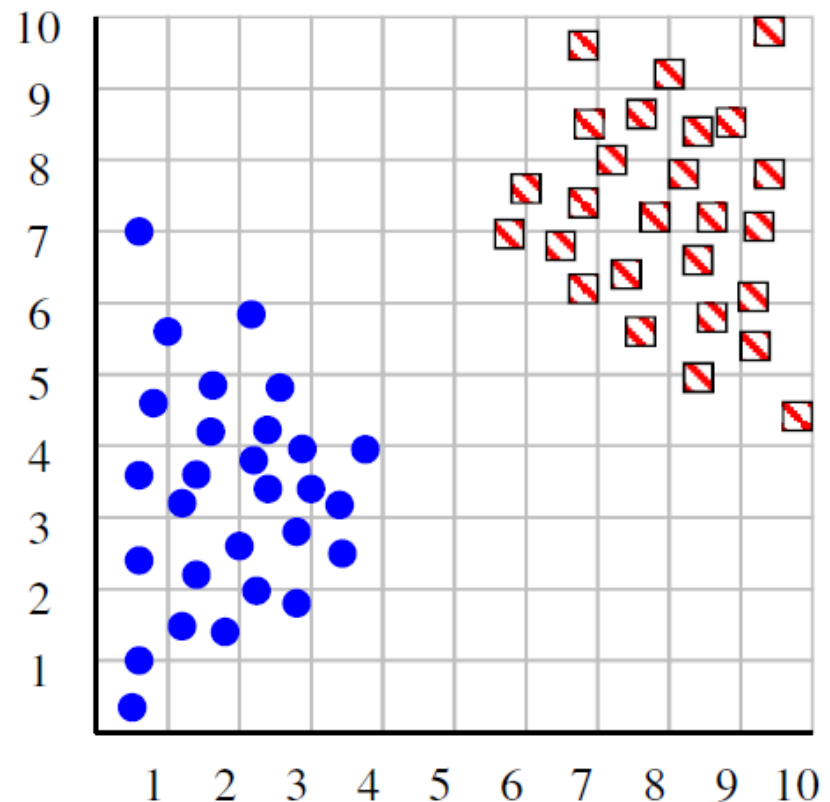
- Použitelné, jen tam, kde *umíme spočítat průměr*. Co kategorická data?
- Velmi záleží na inicializaci – nebezpečí uvíznutí v *lokálním minimu*.
- Požaduje se znalost *počtu shluků*.
- Nevhodné pro zašuměná data s *výjimkami* (outliers).
- **Nevhodné pro situace, kdy shluky nemají konvexní tvar**.

Osnova přednášky

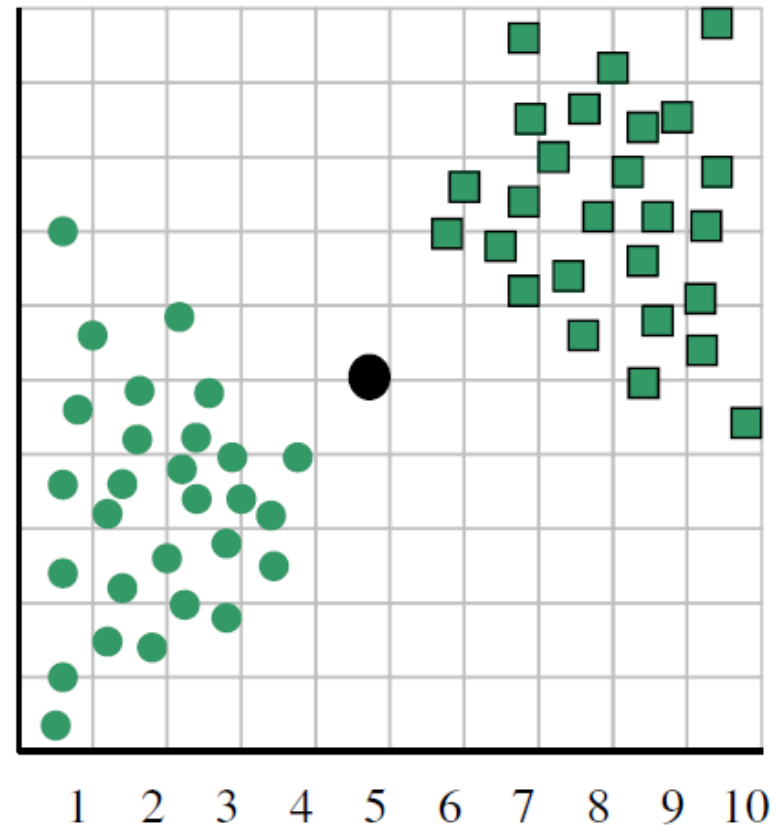
- Motivace
- Míra vzdálenosti
- Hierarchické shlukování
- Hodnocení kvality rozkladu
- Shlukování rozkladem
 - k-means (k-středů)
 - Odhad počtu shluků
 - * EM (expectation maximization) algoritmus, Gaussovská směs

Jak se rozhodnout pro správný počet shluků?

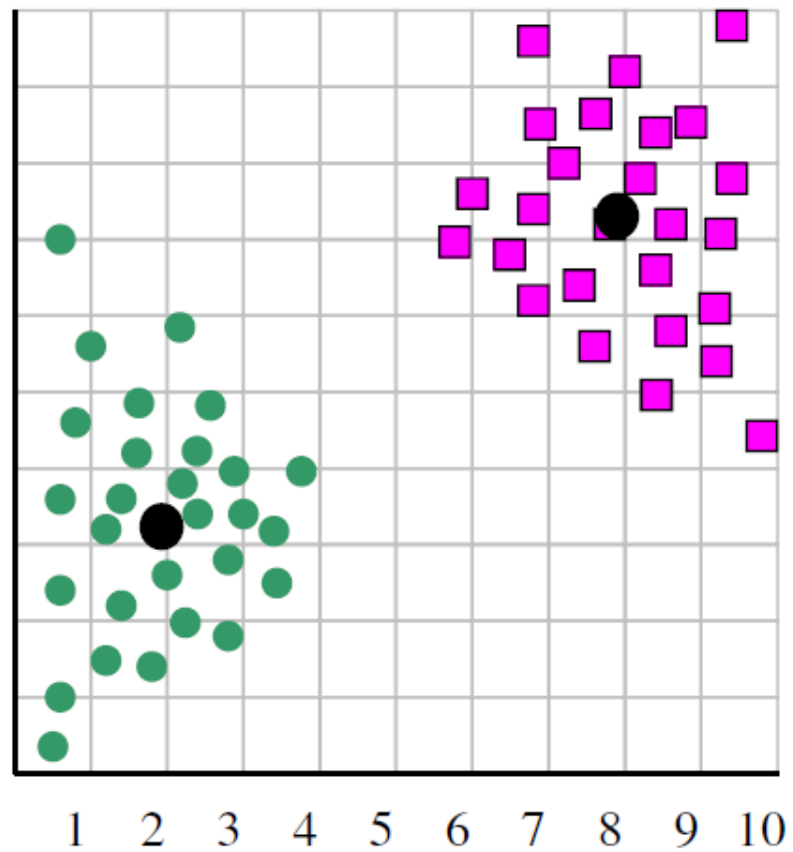
- Obecně jde o otevřený problém.
- Používá se řada **heuristik**
- Jedna z nich vychází ze srovnání **hodnoty objektivní funkce** (=celkový součet vzdáleností uvnitř shluků) pro různé volby počtu shluků



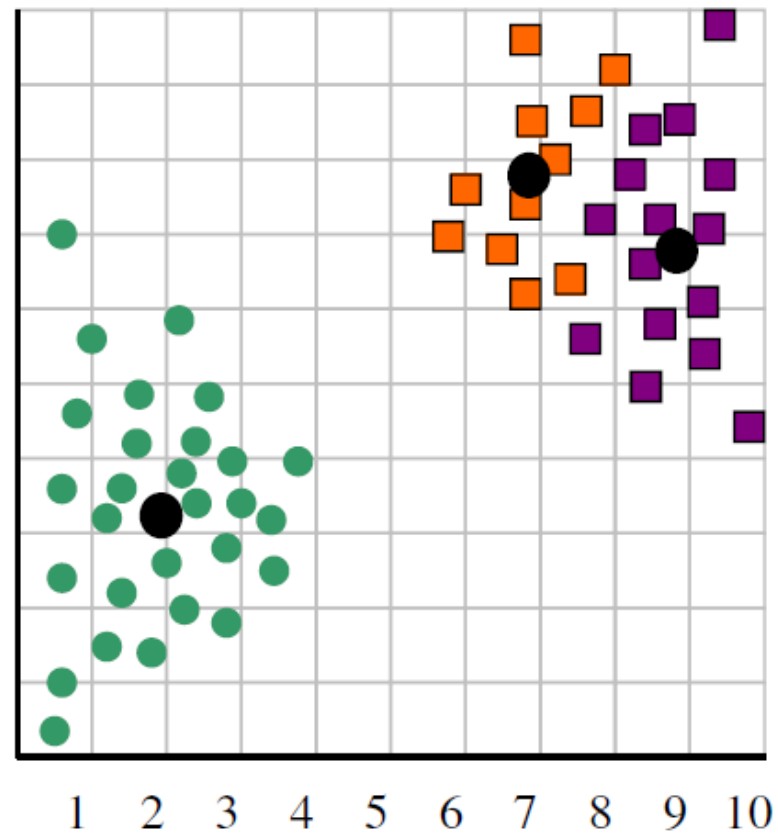
- Je-li $k = 1$, je
výsledná hodnota
objektivní funkce
873



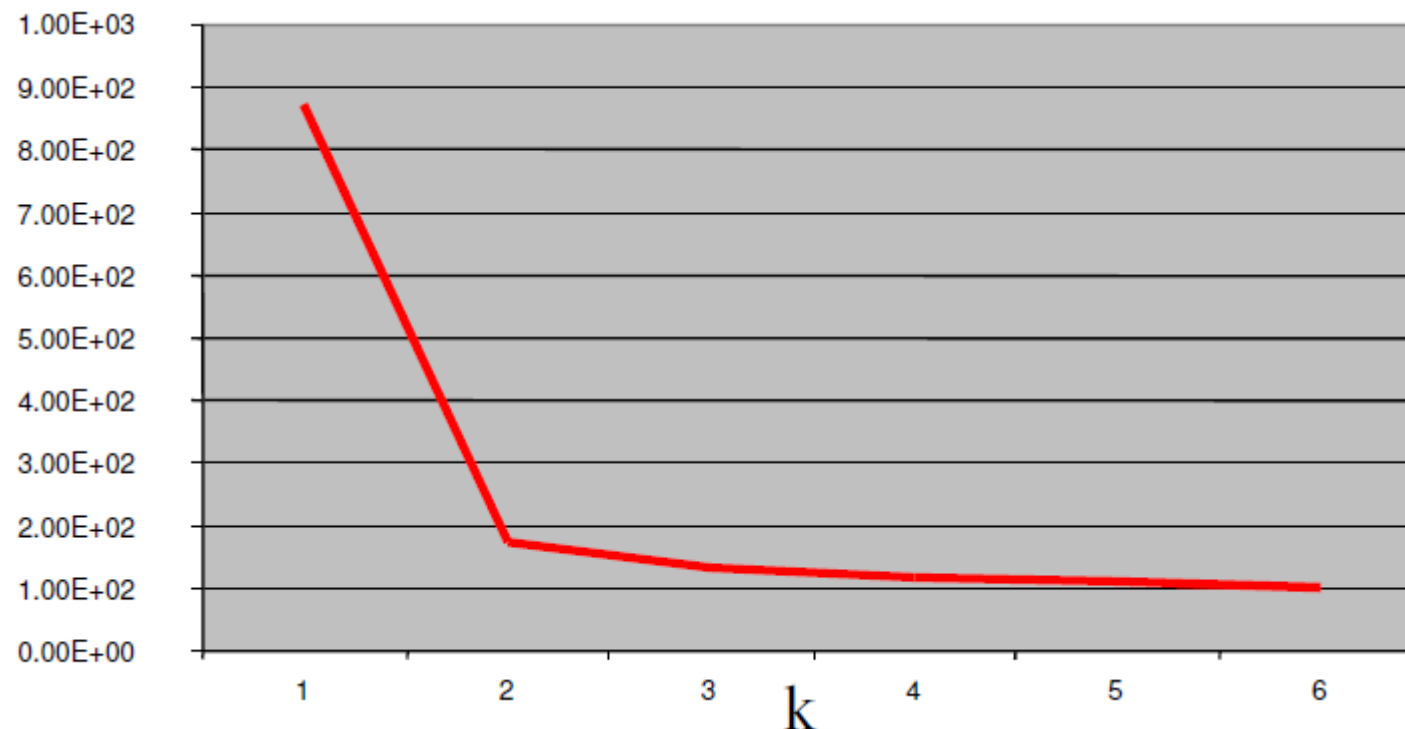
- Je-li $k = 2$, je výsledná hodnota obj. funkce **173,1**



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



- Sledujme hodnotu objektivní funkce pro $k= 1,2,\dots,6$
- 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.

Osnova přednášky

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

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

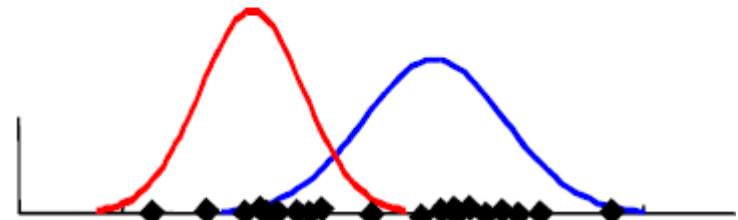
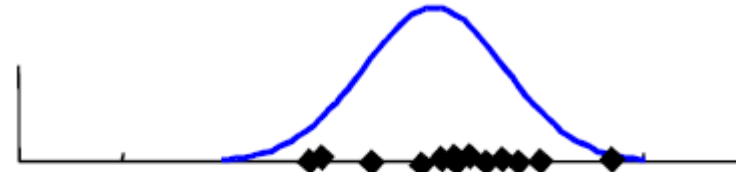
- Gaussian

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

– např. x je výška populace

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

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

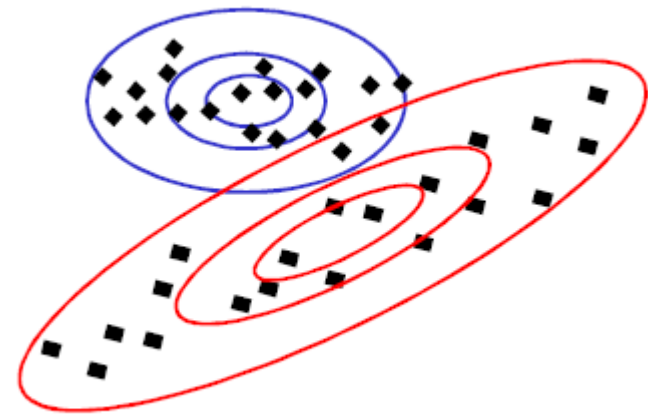


- Gaussovská směs GMM
(Gaus.Mixture Model)

* Vícerozměrný model typu **GMM** „Gaussovská směs“
se iterativně upřesňuje

- Směs vícerozměrných
Gausiánů

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



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

* 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!
- $O(m^3)$

* **GMM + EM** (Expectation Maximization algorithm) = „Soft k-means“

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

$$p_{ij} = P(C = i | \mathbf{x}_j) = \alpha P(\mathbf{x}_j | C = i) P(C = i)$$

$$p_i = \sum_j p_{ij}$$

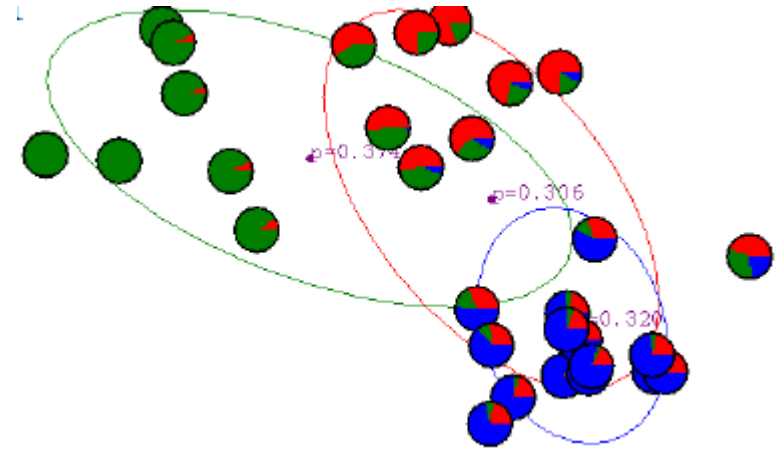
4. **M-krok:** proved' nové odhady parametrů s využitím právě vypočtených hodnot. / T

$$\mu_i \leftarrow \sum_j p_{ij} \mathbf{x}_j / p_i$$

$$\Sigma_j \leftarrow \sum_j p_{ij} \mathbf{x}_j \mathbf{x}_j^T / p_i$$

$$\omega_i \leftarrow p_i$$

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



Srovnání metod shlukování

	Hierarchické shlukování	Shlukování rozkladem	
		K-means	GMM
Časové nároky	$\mathcal{O}(m^3)$ m je počet objektů	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