Vytěžování Dat
Přednáška 4 – Shluková analýza

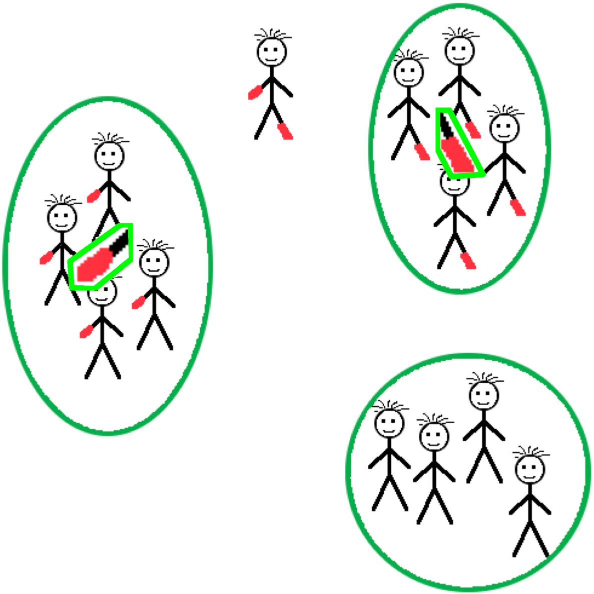
Miroslav Čepek

Fakulta Elektrotechnická, ČVUT

14.10.2011

Co to je shluková analýza

- Je jednou ze základních úloh vytěžování dat.
- Jde o seskupení objektů do skupin podle jejich vlastností. Tak aby si objekty ve skupinách byly "nějak" podobné.
- A zároveň nebyly podobné objektů v jiných skupinách.



Co to je shluková analýza (II)

- V principu jde o optimalizační problém.
- Co se musí optimalizovat?
  - Počet shluků (skupin)
  - Přiřazení instancí do shluků

Co to je shluková analýza (II)

- V principu jde o optimalizační problém.
- Co se musí optimalizovat?
  - Počet shluků (skupin)
  - Přiřazení instancí do shluků

Jak zjistit, že jsou si dva vzory podobné?

- To je obecně velmi složitá otázka.
- Protože shlukovou analýzu budou provádět hlavně počítače, musí být výsledkem nějaké číslo.
- Z matematické analýzy známe pojem metrika – což je jiné označení vzdálenosti.
- Metrika musí splňovat několik základních podmínek, aby ji bylo možné použít.
  - $d(x, y) \geq 0$
  - $d(x, y) = d(y, x)$
  - $d(x, y) = 0 \Leftrightarrow x = y$
  - $d(x, y) + d(y, z) \geq d(x, y)$

Jak zjistit, že jsou si dva vzory podobné?

- To je obecně velmi složitá otázka.
- Protože shlukovou analýzu budou provádět hlavně počítače, musí být výsledkem nějaké číslo.
- Z matematické analýzy známe pojem metrika – což je jiné označení vzdálenosti.
- Metrika musí splňovat několik základních podmínek, aby ji bylo možné použít.
  - $d(x, y) \geq 0$
  - $d(x, y) = d(y, x)$
  - $d(x, y) = 0 \Leftrightarrow x = y$
  - $d(x, y) + d(y, z) \geq d(x, y)$

Metriky

- Jaké znáte metriky?
- Eukleidovská metrika
- Manhattanská metrika
- Kosinová metrika
- Příklady dalších metrik
  - Editační vzdálenost (vzdálesnost dvou slov = počet změn, kterými můžu změnit jedno slovo na druhé)
  - Grafová metrika (počet hran, které musím v grafu projít, abych se dostal do z jednoho uzlu do druhého)
  - http://en.wikipedia.org/wiki/Metric_space

Metriky

- Jaké znáte metriky?
- Eukleidovská metrika
- Manhattanská metrika
- Kosinová metrika
- Příklady dalších metrik
  - Editační vzdálenost (vzdálesnost dvou slov = počet změn, kterými můžu změnit jedno slovo na druhé)
  - Grafová metrika (počet hran, které musím v grafu projít, abych se dostal do z jednoho uzlu do druhého)
  - http://en.wikipedia.org/wiki/Metric_space

Metriky

- Jaké znáte metriky?
- Eukleidovská metrika
- Manhattanská metrika
- Kosinová metrika
- Příklady dalších metrik
  - Editační vzdálenost (vzdálesnost dvou slov = počet změn, kterými můžu změnit jedno slovo na druhé)
  - Grafová metrika (počet hran, které musím v grafu projít, abych se dostal do z jednoho uzlu do druhého)
  - `http://en.wikipedia.org/wiki/Metric_space`

Eukleidovská metrika

- Nejpřirozenější metrika, protože se s ní běžně setkáváme.
- Jak změříme vzdálenost dvou bodů na tabuli?
- Pravítkem :)!
- A když známe souřadnice, můžeme ji spočítat. Jak?

Eukleidovská metrika

- Nejpřirozenější metrika, protože se s ní běžně setkáváme.
- Jak změříme vzdálenost dvou bodů na tabuli?
- Pravítkem :)!
- A když známe souřadnice, můžeme ji spočítat. Jak?

Eukleidovská metrika (II)

- Pythagorova věta! $c = \sqrt{a^2 + b^2}$
- A Pythagorovu větu můžeme zobecnit pro $\Re^n$

$$\vec{x} = (x_1, x_2, \ldots, x_n), \vec{y} = (y_1, y_2, \ldots, y_n)$$

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

Eukleidovská metrika (II)

- Pythagorova věta! $c = \sqrt{a^2 + b^2}$
- A Pythagorovu větu můžeme zobecnit pro $\Re^n$

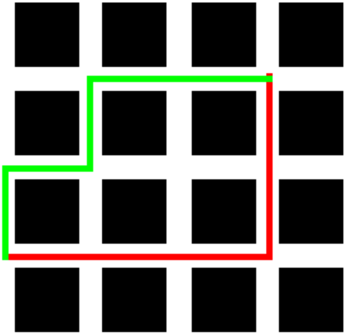
$$\vec{x} = (x_1, x_2, \ldots, x_n), \vec{y} = (y_1, y_2, \ldots, y_n)$$

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

Manhattanská metrika (City-block distance)

- Základní myšlenka: Kolik bloků ve městě musím obejít, abych se dostal z jednoho místa na druhé?
- Nebo také – kolik tahů králem musím udělat abych se dostal z jednoho místa šachovnice na druhé?

Manhattanská metrika (City-block distance) (II)



- Pokud znám souřadnice, vzdálenost spočítam takto:

$$dist(\vec{x}, \vec{y}) = |x_1 - y_1| + |x_2 - y_2| + \ldots + |x_n - y_n|$$

Kosinová vzdálenost

- Vzdálenost dvou vektorů je úhel, který svírají.

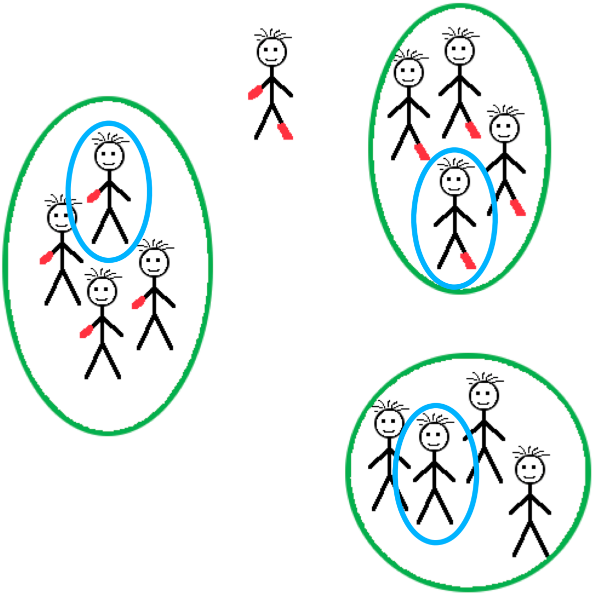
$$similarity(\vec{x}, \vec{y}) = \frac{\sum_{i=1}^{n}(x_i * y_i)}{\sqrt{\sum_{i=1}^{n}(x_i^2) * \sum_{i=1}^{n}(y_i^2)}}$$

- Výsledky této funkce jsou v rozmezí -1 ... +1. -1 znamená úplný opak, 0 nezávyslost a +1 naprostou shodu.
- Aby výsledky vyhovovali definici metriky je potřeba podobnost odečíst od jedné.

$$dist(\vec{x}, \vec{y}) = 1 - similarity(\vec{x}, \vec{y})$$

Shlukování pomocí KMeans

- Jednostlivé shluky budou zastoupeny jedním reprezentantem, který ponese vlastnosti typické pro danou skupinu/shluk.
- Každá instance (vzor) v datech bude reprezentována reprezentantem, který je jí nejpodobnější .
- Jinými slovy – který ji bude nejblíž (v dané metrice).



Shlukování pomocí KMeans

- Jak určit, kde je správné místo pro reprezentanty?
- Chceme, aby vzdálenost mezi reprezentanty a instancemi byla co nejmenší.
- Snažíme se vlastně minimalizovat součet všech vzdáleností mezi instancemi a jejich reprezentanty $\Rightarrow$ jde o optimalizační problém.
- Taková optimalizace se dá řešit mnoha způsoby, ale jeden z nejjednodušších je iterační.

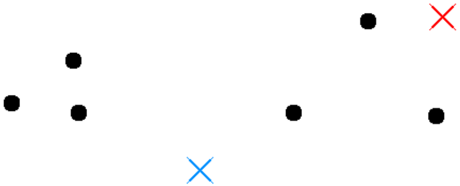
Algoritmus KMeans – značení

- Máme množinu n vstupních vzorů/instancí (vektorů) x_k. Jednotlivé složky vektoru budeme označovat $x_k(s)$.
- A máme množinu K reprezentantů. $means_i^t$ je i-tý reprezentant v kroku t.

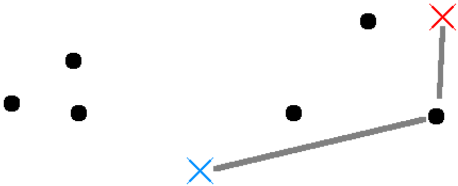
Algoritmus KMeans

1. Nastav reprezentanty $means_i^0$ do náhodných počátečních bodů.
2. Najdi a přiřaď každé instanci jeho nejbližšího reprezentanta.
   - $\forall x$ najdi j tak, aby $dist(x, means_j^t) \leq dist(x, means_i^t) \forall i$
   - a pro každého reprezentanta $means_i^t$ vytvoř množinu $nearest_i^t$ instancí, ke kterým je nejblíž.
3. Přesuň reprezentanta tak aby ležel "uprostřed" své množiny nejbližších instancí.
   - $means_i^{t+1}(s) = \frac{1}{\|nearest_i^t\|} \sum_{x_k \in nearest_i^t} x_k(s)$
4. Pokud se změnila poloha alespoň jednoho preprezentanta, vrať se na bod 2. Jinak skonči.

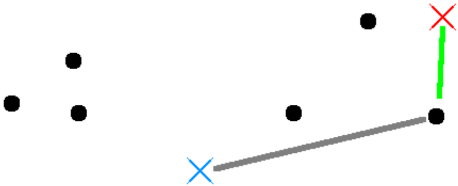
Ilustrace KMeans



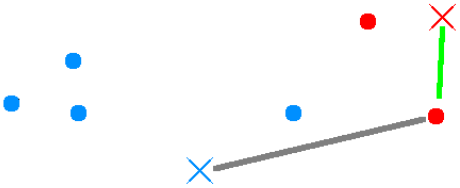
Ilustrace KMeans (II)



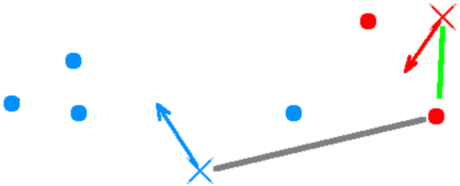
Ilustrace KMeans (III)



Ilustrace KMeans (IV)



Ilustrace KMeans (V)



Pohádka o Algoritmu KMeans :)

- Once there was a land with N houses.
- One day K kings arrived to this land.
- Each house was taken by the nearest king.
- But the community wanted their king to be at the center of the village, so the throne was moved there
- Then the kings realized that some houses were closer to them now, so they took those houses, but they lost some. This went on and on... (2-3-4)
- Until one day they couldn't move anymore, so they settled down and lived happily ever after in their village...

Problémy a stabilita shlukování pomocí KMeans

- Dopadne shlukování pomocí KMeans pokaždé stejně?
- Jak určit správný počet středů (shluků)?
- Jak vyhodnotit jestli shlukování dopadlo dobře a jestli jsme zvolili přiměřené K?

Problémy a stabilita shlukování pomocí KMeans

- Dopadne shlukování pomocí KMeans pokaždé stejně?
- Jak určit správný počet středů (shluků)?
- Jak vyhodnotit jestli shlukování dopadlo dobře a jestli jsme zvolili přiměřené K?

Problémy a stabilita shlukování pomocí KMeans

- Dopadne shlukování pomocí KMeans pokaždé stejně?
- Jak určit správný počet středů (shluků)?
- Jak vyhodnotit jestli shlukování dopadlo dobře a jestli jsme zvolili přiměřené K?

Vyhodnocení shluků vytvořených KMeans algoritmem

- Jednou z možných metod je tzv. silueta.
- Silueta pro každou vstupní instanci spočítá jistotu zařazení instance do daného shluku.

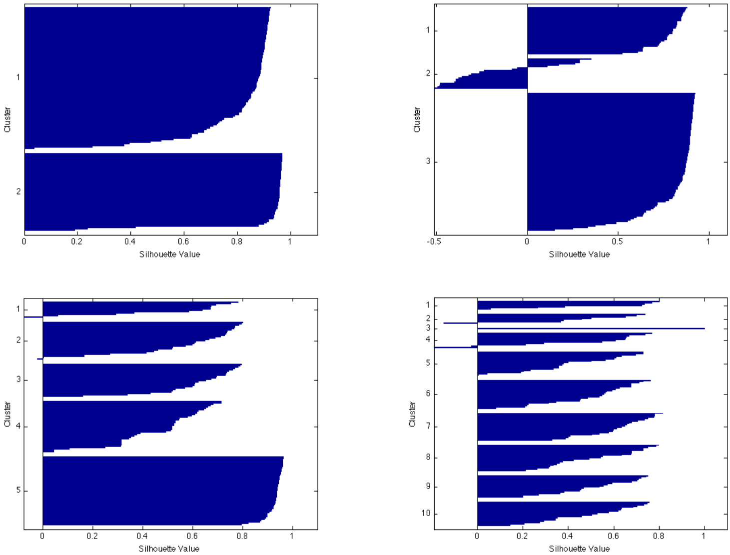
$$s(x_k) = \frac{b(x_k) - a(x_k)}{\max(a(x_k), b(x_k))}$$

- $a(x_k)$ je průměrná vzdálenost x_k od ostatních instancí shluku, ke kterému je přiřazena.
- $b(x_k)$ je průměrná vzdálenost x_k od instancí v nejbližším dalším shluku.
- Výsledné hodnoty jsou mezi -1 (x_k do shluku úplně nepatří) a +1 (úplně patří)
- `ftp://ftp.win.ua.ac.be/pub/preprints/87/Silgra87.pdf`

Vyhodnocení shluků vytvořených KMeans algoritmem (II)

- Pokud vypočítáte siletu pro všechny instance a vykreslíte ji do grafu, můžete si udělat představu, jak shlukování dopadlo.

Ukázka Siluety – shluky Kosatců



- Které shlukování dopadlo lépe?
- Co třeba průměrná silueta přes všechny instance (ideálně přes testovací data)?

Stabilita shluků

- Jak zkusit, že shluky opravdu v datech jsou a výsledné shluky nejsou náhoda?
- Náhodným smazáním např. 10% různých instancí vygenerovat M podmnožin dat a spustit shlukování na každé podmnožině.

- Existuje několik ukázkových apletů/aplikací, kde si můžete zkusit, jak algoritmus funguje.
- `http://home.dei.polimi.it/matteucc/Clustering/tutorial_html/AppletKM.html`

Hierarchické shlukování – úvod

- KMeans, jak jsme viděli, má některé mouchy.
  - Kolik je v datech shluků?
  - Závislost výsledků na počátečních podmínkách.
- Šlo by shlukování dělat i jinak?
- Šlo :). Jednou z možností je Hierarchické shlukování.
- Základní myslenka je, že vytvoříme hierarchii shluků. Vždy spojíme dva nejpodobnější shluky do jednoho většího.
- A takto budeme pokračovat, dokud nevytvoříme jeden mega-shluk.

Hierarchické shlukování – úvod

- KMeans, jak jsme viděli, má některé mouchy.
  - Kolik je v datech shluků?
  - Závislost výsledků na počátečních podmínkách.
- Šlo by shlukování dělat i jinak?
- Šlo :). Jednou z možností je Hierarchické shlukování.
- Základní myslenka je, že vytvoříme hierarchii shluků. Vždy spojíme dva nejpodobnější shluky do jednoho většího.
- A takto budeme pokračovat, dokud nevytvoříme jeden mega-shluk.

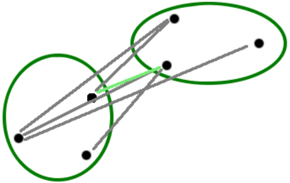
Hierarchické shlukování

1. Začne ze stavu, kdy každá instance je jedním shlukem.
2. Najdi dva nejbližší shluky.
3. Spoj je do jednoho.
4. Zůstávají nějaké shluky, které lze spojit? Pokud ano, vrať se na bod 2.

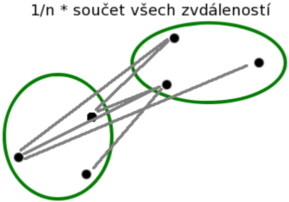
Nejbližší shluky

- Jak zjistím vzdálenost dvou shluků?
- Dokud shluky obsahují jen jednu instanci, je spočítání vzdálenosti jednoduché. Ale pak?
- Vzdálenost shluků je určena
  - Nejbližší sousedé – vzdáleností nejbližších instancí ve shluku.
  - Nejvzdálenější sousedé – vzdáleností nejvzdálenějších instancí ve shluku.
  - Vzdálenost středů – vzdáleností center (středů) shluků.
  - Průměrná vzdálenost – průměrná vzdálenost mezi všemi instancemi v obo shlucích

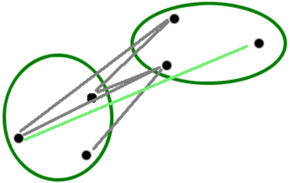
Vzdálenost shluků – ilustrace



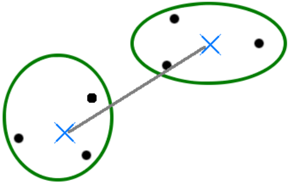
Nejkratší vzdálenost



Průměrná vzdálenost



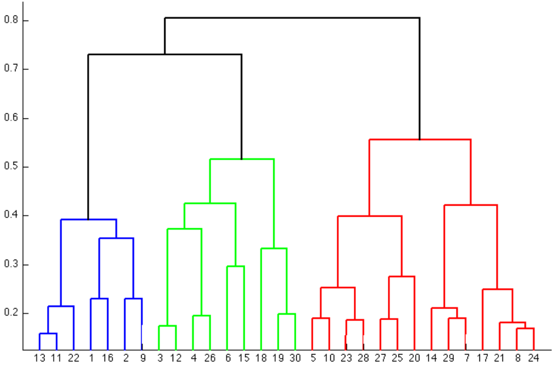
Největší vzdálenost



Vzdálenost mezi reprezentanty

Dendrogram

- Když zkusíme vizualizovat postup shlukování – tj. které shluky se spojují, získáme strom – dendrogram.
- Jak nalezneme počet shluků? Výběrem :), podle toho, kolik shluků potřebujeme nebo kolik vyjde jako nejvhodnější.



Vyhodnocení hierarchického shlukování

- Můžeme opět použít siluetu, stejně jak jsme ji používali v K-Means.
- Druhou možností je vypočítat CPCC (Cophenetic Correlation Coeffitient).
- CPCC je normovaná kovariance vzdáleností v původním prostoru a v dendrogramu.
- Pokud je hodnota CPCC menší než cca 0.8, všechny instance patří do jediného velkého shluku.
- Obecně platí, že čím vyšší je kofenetický koeficient korelace, tím nižší je ztráta informací, vznikající v procesu slučování objektů do shluků.

Další informace a zdroje

- http://cmp.felk.cvut.cz/cmp/courses/recognition/zapis_prednasky/zapis_02/13/shlukovani.pdf
- http://www.fit.vutbr.cz/study/courses/ZZD/public/seminar0304/hlukovani2.pdf