



TECHNICKÁ UNIVERZITA V LIBERCI
Fakulta mechatroniky, informatiky
a mezioborových studií ■

Shlukování

Petr Červa, František Kynych
18. 11. 2021 | MVD





TECHNICKÁ UNIVERZITA V LIBERCI
Fakulta mechatroniky, informatiky
a mezioborových studií ■

Část I.: Úvod do problematiky



Učení s učitelem (supervised learning)

- Algoritmy se učí na základě dat, u kterých jsou člověkem připraveny příslušné správné značky ve formě
 - Příslušnosti ke třídě – úloha klasifikace
 - Hodnoty závislé veličiny – úloha regrese

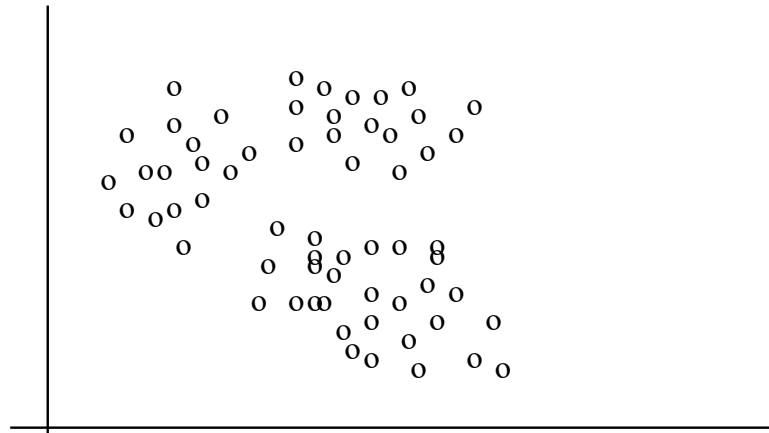
Učení bez učitele (unsupervised learning)

- Algoritmy se učí na základě dat, u kterých nejsou člověkem připraveny žádné značky
 - Neučí se klasifikovat
 - Neučí se ani predikovat
- Algoritmy bez učitele místo toho hledají vnitřní strukturu dat
 - Ani tato struktura není ale předem označkována

Shluková analýza ~ hledání vnitřní struktury

- Cílem je rozdělit data do několika skupin, resp. shluků (angl. clusters)
- Musí přitom platit, že:
 - Data uvnitř jednoho shluku jsou si vzájemně podobná
 - Data uvnitř jednoho shluku se liší od dat ve všech ostatních shlucích

Lze v obrázku najít nějaké shluky a kolik?



Co je obecně třeba ke shlukování?

- Metriky pro měření vzdálenosti mezi daty
- Metriky pro měření vzdálenosti (podobnosti) mezi shluky
- Metriky pro vyhodnocování úspěšnosti shlukování

Metriky pro měření vzdálenosti mezi daty

Typicky používané vzdálenostní funkce pro P dimenzí:

- Euklidovská vzdálenost (Euclidean dist.)

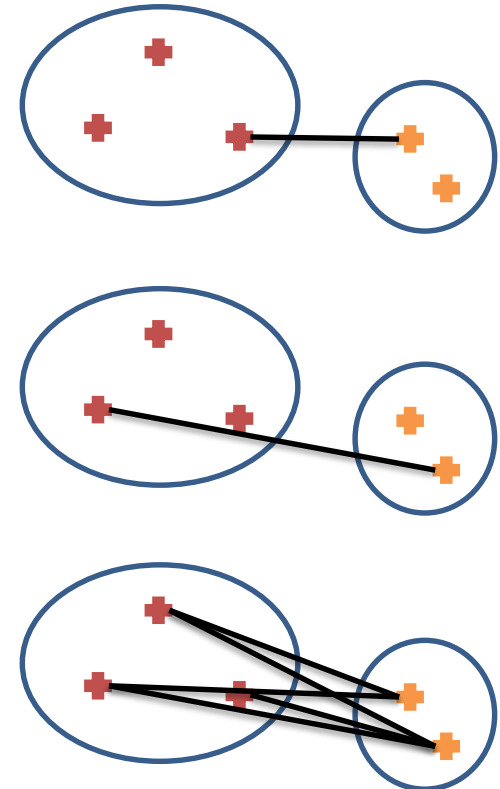
$$d(\mathbf{x}, \mathbf{z}) = \sqrt{\sum_{i=1}^P (x_i - z_i)^2}$$

- Vzdálenost v městských blocích (Manhattan dist.)

$$d(\mathbf{x}, \mathbf{z}) = \sum_{i=1}^P |x_i - z_i|$$

Metriky pro určování podobnosti shluků

- 1) **Min** – (single linkage) podobnost (vzdálenost) dvou nejvíce podobných vzorků přiřazených do shluků
 - Hrozí nebezpečí vytvoření shluků jen na základě vzájemné blízkosti dvou outlierů
- 2) **Max** – (complete linkage) podobnost dvou nejméně podobných vzorků
- 3) **Group average** – průměrná hodnota podobnosti pro všechny možné dvojice vzorků z obou shluků
- 4) **Vzdálenost centroidů**



Vyhodnocování úspěšnosti shlukování

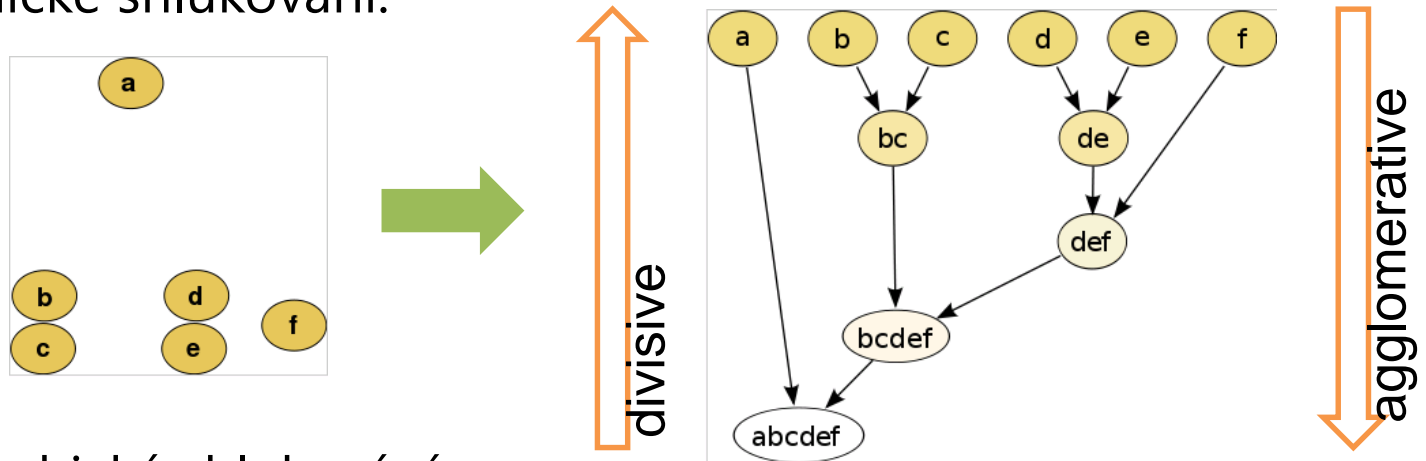
- Nejobtížnější problém
 - Data nejsou označkována (učení bez učitele)
 - Výsledek se často ohodnocuje na základě posouzení expertem
- Bere se přitom v potaz:
 - Vnitřní kompaktnost shluku
 - Jak moc jsou data vzdálena od centroidu
 - Vzájemná izolace shluků
 - Centroidy jednotlivých shluků by od sebe měly být co nejvíce vzdáleny



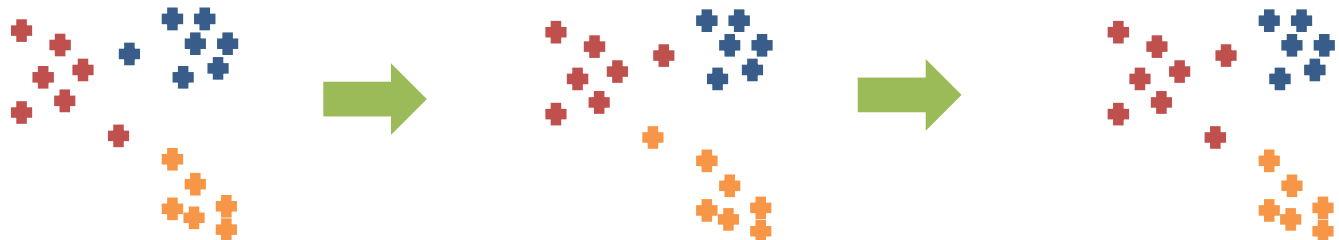
<http://www.mit.edu/~9.54/fall14/slides/Class13.pdf>

Metody shlukování

- Hierarchické shlukování:



- Nehierarchické shlukování:
 - Samooorganizující neuronové sítě
 - K-Means

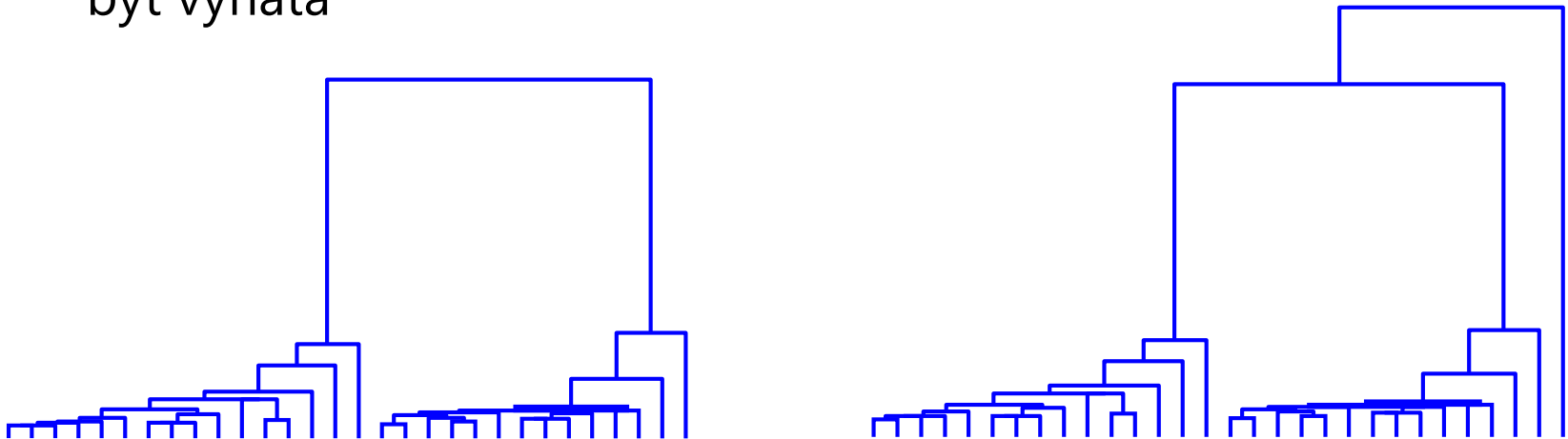




Část II.: Hierarchické shlukování

Hierarchické shlukování

- Jednotlivé kroky procesu shlukování je možné zaznamenat pomocí tzv. dendrogramu
- Dendrogram reflektuje míru podobnosti jednotlivých shluků, umožňuje tak odhadnout optimální počet shluků; příp. identifikovat tzv. outliery
- Nevýhodou je, že data jednou zařazená do shluku již nemohou být vyňata





Část III.: Nehierarchické shlukování

Algoritmus K-průměrů (K-means)

- Pro zvolené číslo K se hledá **rozklad trénovací množiny** na K podmnožin (shluků) tak, že každý j -tý shluk má svého reprezentanta (centroid) μ_j a je charakterizován součtem vzdáleností mezi ostatními prvky shluku a centroidem.
- **Kritériem** vhodnosti rozkladu je **součet všech dílčích vzdáleností** přes všechny shluky.

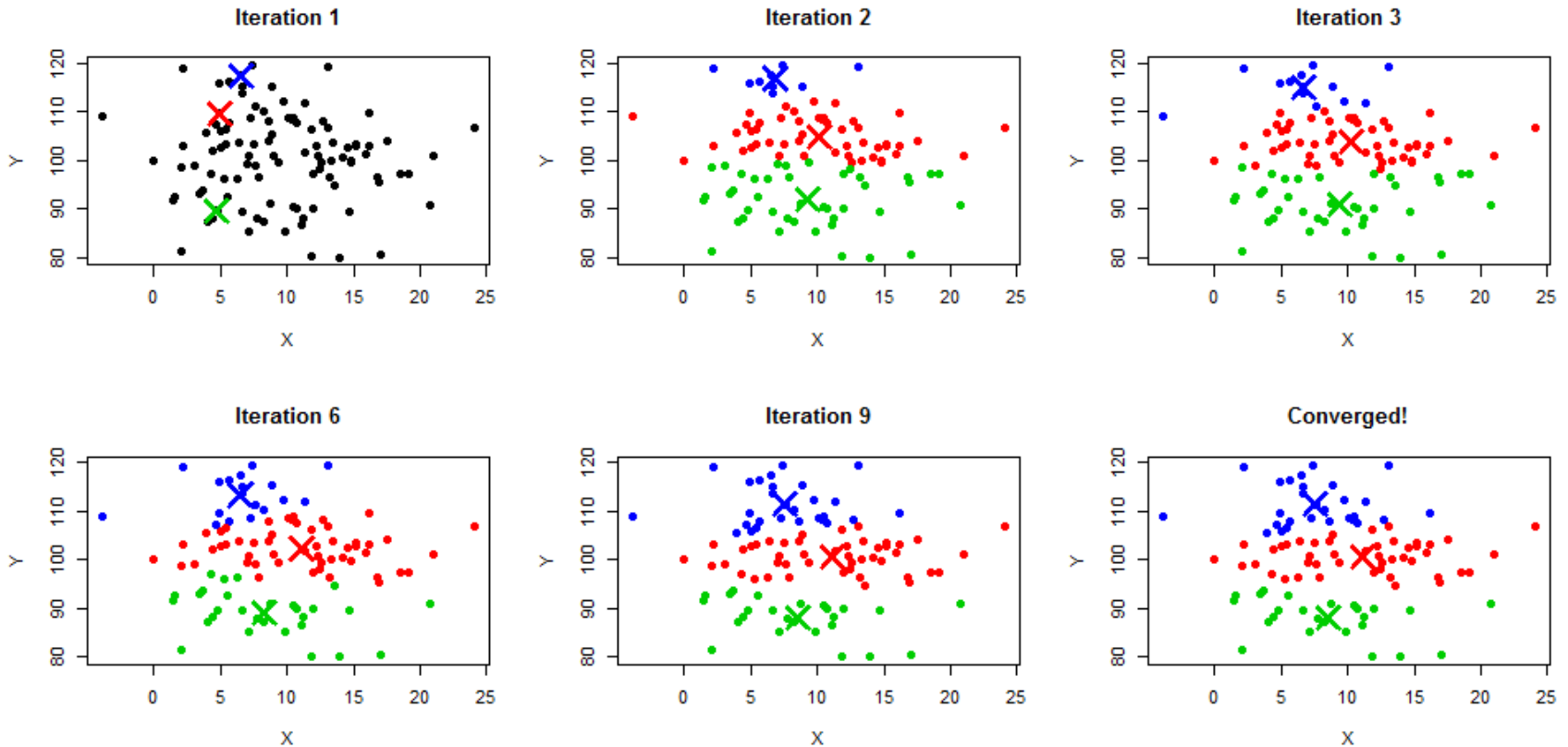
$$J = \sum_{j=1}^K \sum_{i=1}^N \|x_i^{(j)} - \mu_j\|$$

- Toto kritérium se snažíme **minimalizovat**

Algoritmus K-průměrů (K-means)

- Algoritmus je založen na **iteračním postupu**:
 1. Zvolíme K prvků TrM jako prvotní odhady centroidů.
 2. Zařadíme každý prvek TrM do jedné z K skupin na základě nejmenší vzdálenosti k centroidu.
 3. Pro každou skupinu vypočteme nový centroid tak, aby měl nejmenší vzdálenost ke všem prvkům skupiny.
 4. Vyhodnotíme celkové kritérium a v případě, že se jeho hodnota liší od předchozí hodnoty o méně než ϵ , iterační proces zastavíme, jinak návrat na krok 2.

Ilustrace průběhu algoritmu K-průměrů



<http://www.learnbymarketing.com/methods/k-means-clustering/>

K-means – silné stránky

- Jednoduchý na implementaci a pochopení
- Výpočetní náročnost je $O(TKN)$, kde
 - N je počet dat
 - K je počet shluků
 - T je počet iterací
 - Protože K a T je obvykle malé, je K-means považován za lineární
- Nejpopulárnější algoritmus
 - Výsledky se obtížně porovnávají, nelze určit nejlepší algoritmus

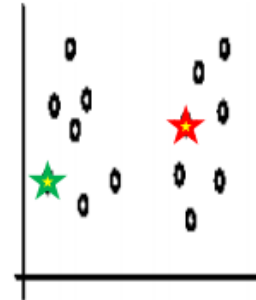
K-means – slabé stránky

- Není zajištěno dosažení globálního minima kritériální funkce
- Výsledné centroidy mohou záviset na volbě počátečních centroidů
 - Proces lze opakovat s různými počát. centroidy a vybrat pak to řešení, které má minimální hodnotu kritériální funkce
- Náchylný na přítomnost outlierů
 - Lze částečně eliminovat – viz dále
- Nefunguje pro všechna rozložení příznaků
- Na shlukovaných datech je nutné umět spočítat střední hodnotu
- Algoritmus neřeší otázku nejvhodnějšího čísla K

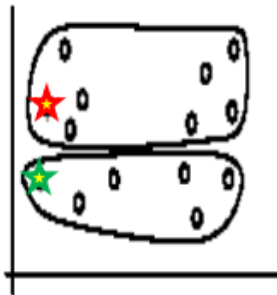
K-means – závislost na inicializaci



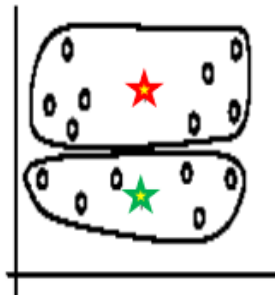
Random selection of seeds (centroids)



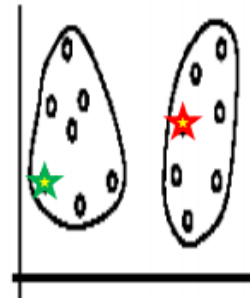
Random selection of seeds (centroids)



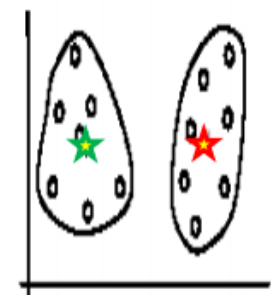
Iteration 1



Iteration 2



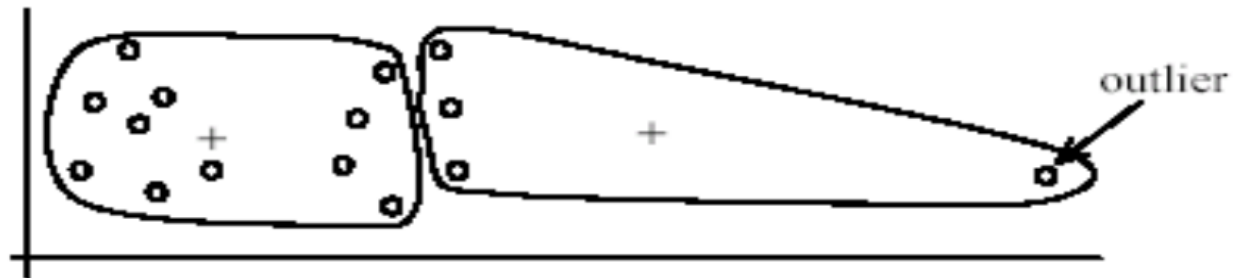
Iteration 1



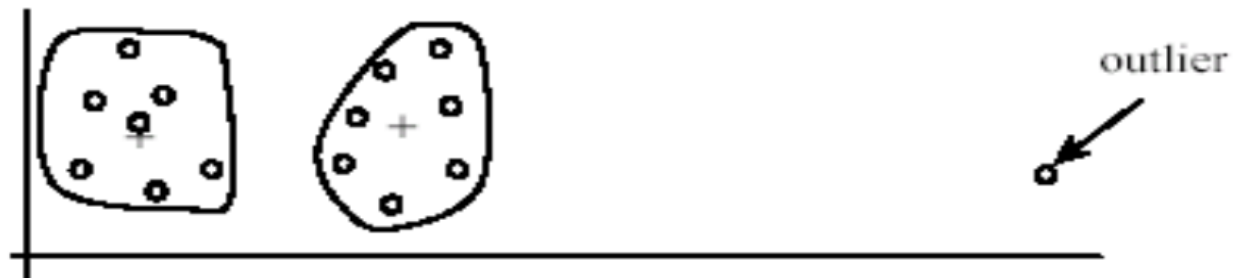
Iteration 2

<http://www.mit.edu/~9.54/fall14/slides/Class13.pdf>

K-means – outliery



(A): Undesirable clusters



(B): Ideal clusters

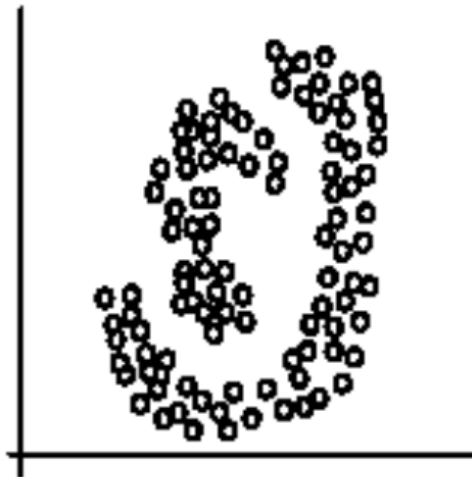
<http://www.mit.edu/~9.54/fall14/slides/Class13.pdf>

K-means – potlačení vlivu outlierů

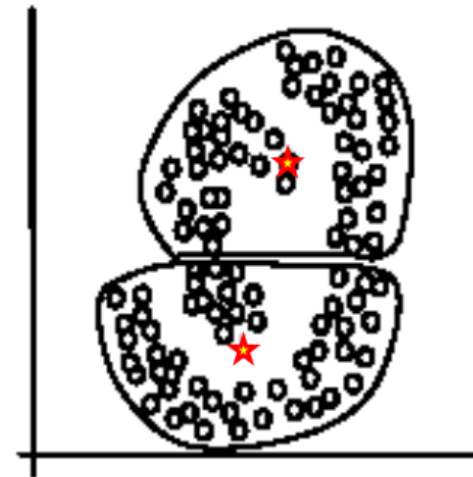
- Je možné odstranit data, která jsou nejvíce vzdálená od centroidů
 - Odstranění je vhodné provádět pouze pokud jsou body vzdáleny ve více iteracích po sobě.
- Je možné data náhodně navzorkovat – vybrat pro výpočet polohy centroidu jen menší množství bodů
 - Outlier pak nemusí být vybrán a nemusí negativně ovlivnit výsledek

K-means – nevhodná rozložení dat

- K-means je vhodný pro data, která jsou rozložena uvnitř vícerozměrné koule nebo elipsoidu
- Jinak může selhat:



(A): Two natural clusters



(B): k -means clusters

<http://www.mit.edu/~9.54/fall14/slides/Class13.pdf>

Algoritmus LBG (Linde, Buzo, Gray)

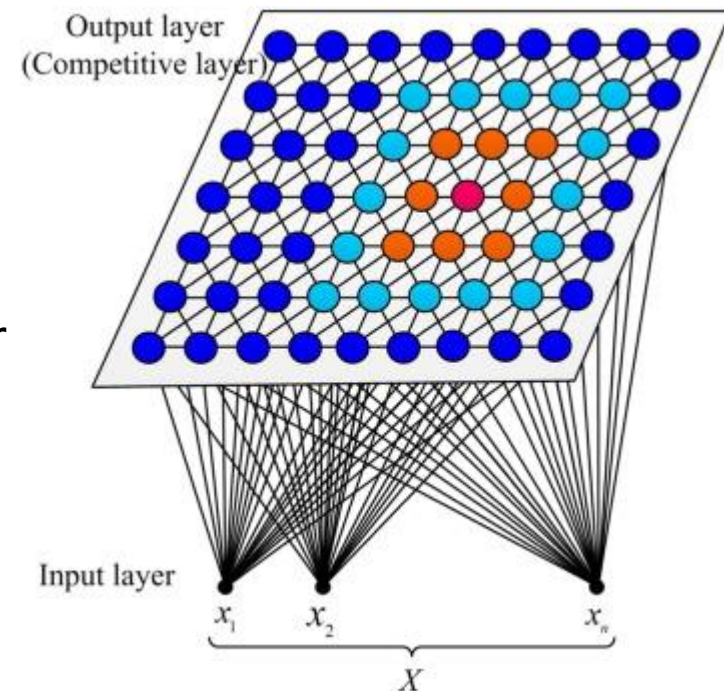
Na rozdíl od K-means řeší též nalezení čísla K

- Dvojnásobně iterační procedura (pro K , i pro určení centroidů)
 1. **Inicializace:** Nastav $K = 1$. Najdi centroid.
 2. **Rozdělení** ($K=2K$): Pro každou dosavadní skupinu urči dva nové počáteční centroidy.
 3. **Nalezení nových centroidů:** S celou TrM a s číslem K proved' K-means alg. Urči hodnotu kritériální funkce.
 4. **Ukončení:** Je-li dosaženo cílové číslo K nebo se hodnota kritériální funkce již významně nemění, skonči, jinak zpět na krok 2.

Pozn. V kroku 2 je také možné rozdělit pouze největší shluk. Pak $K=K+1$

Samoorganizující neuronové sítě (SOM)

- Název pochází z angl. Self Organizing Network
- SOM mají vstupní a výstupní vrstvu
- Neurony ve výstupní vrstvě bývají uspořádány do vícerozměrné struktury
- Ke každému neuronu je přiřazen vektor vah w_i
 - **Určuje polohu neuronu v prostoru**
- Okolo každého neuronu je definován jeho blízké okolí (region) R



SOM - učení

- Trénovací data se zpracovávají vzorek po vzorku
- Pro i -tý neuron a j -tý vzorek se vypočte vzdálenost $\|\mathbf{w}_i - \mathbf{x}_j\|$
- Váhy neuronu, který je vzorku nejbližší, se aktualizují podle vztahu

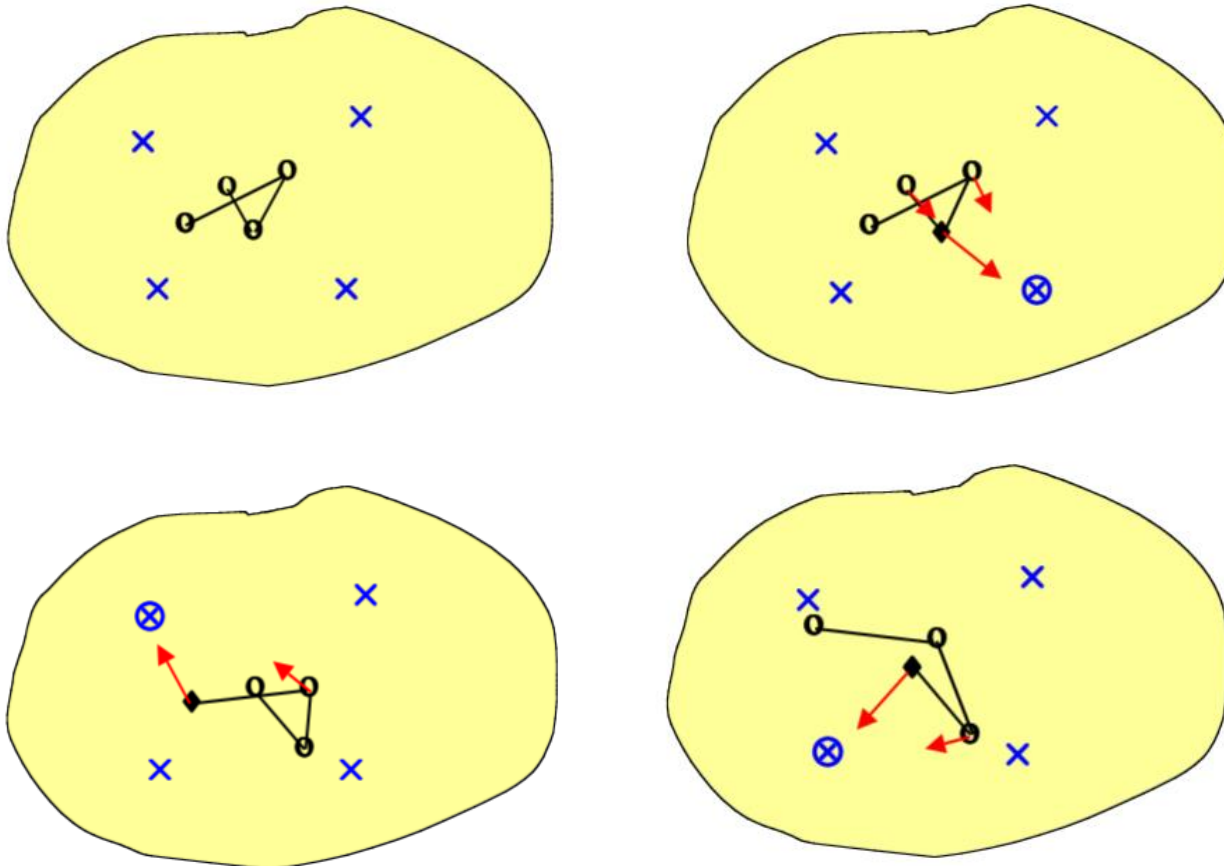
$$\mathbf{w}_{t+1,best} = \mathbf{w}_{t,best} + \alpha(\mathbf{x}_j - \mathbf{w}_{t,best})$$

- Vektor \mathbf{w}_{best} se tedy posune směrem k vektoru \mathbf{x}_j
- Váhy neuronů v rámci regionu nejbližšího neuronu se posunou méně:

$$\mathbf{w}_{t+1,r} = \mathbf{w}_{t,r} + \beta(\mathbf{x}_j - \mathbf{w}_{t,r}), \quad \beta < \alpha$$

=> **Poloha nejbližších neuronů se posune směrem k vektoru \mathbf{x}_j**

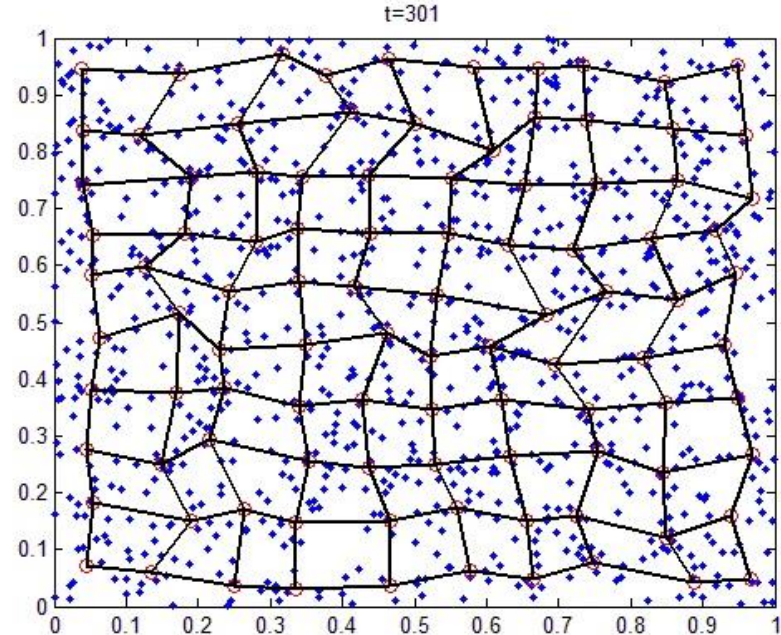
SOM – ilustrace procesu učení



<http://www.mit.edu/~9.54/fall14/slides/Class13.pdf>

SOM – výsledek učení

- Během učení se neurony posouvají směrem k trén. vektorům
- Sít' se podle polohy těchto vektorů sama organizuje
- Ve výsledku neurony zaujmou polohu centroidů shluků přítomných v datech



<https://www.mathworks.com/matlabcentral/mlc-downloads/downloads/submissions/46481/versions/1/screenshot.jpg>



TECHNICKÁ UNIVERZITA V LIBERCI
Fakulta mechatroniky, informatiky
a mezioborových studií ■

Část IV.: Využití shlukování



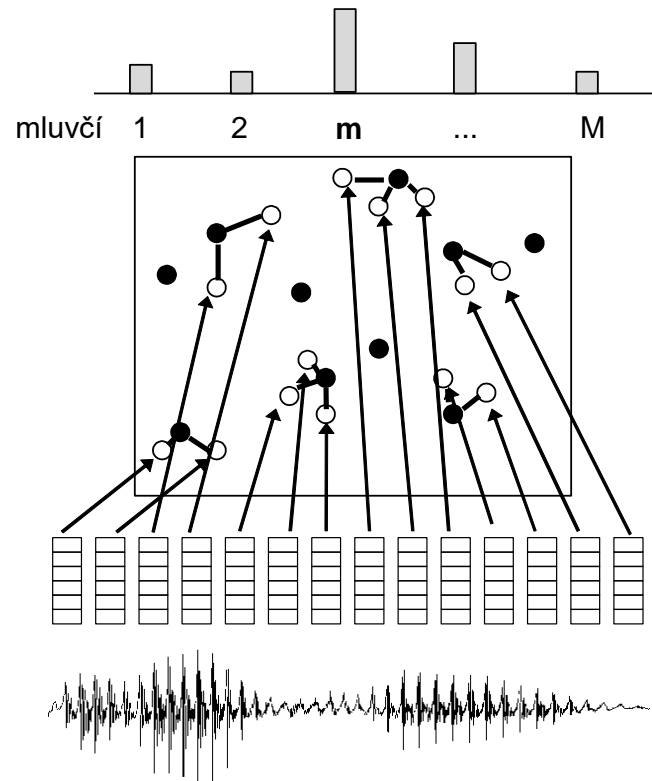
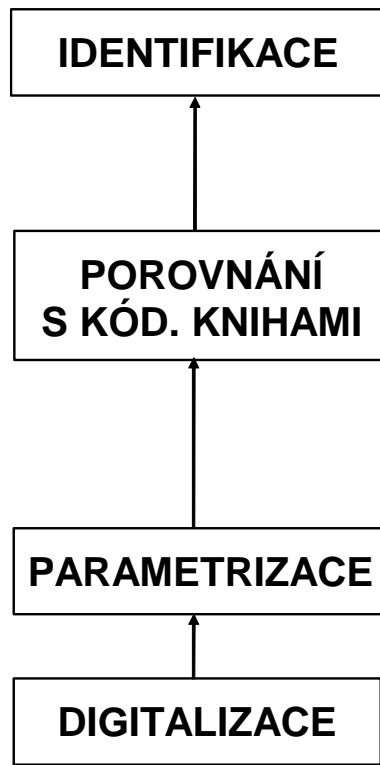
Rozpoznávání osob podle hlasu

- **Fáze trénování**

- Každá osoba v trénovací množině namluví několik minut řeči.
- Nahrávky se zparametrizují
- Vypočtou se příznakové vektory pro každých 10 ms řeči.
- Příznakové vektory vytvoří v obrazovém prostoru shluky, které se identifikují a reprezentují svými etalony.
- Tento proces se nazývá také jako vektorová kvantizace
- Pozice etalonů pak vytvářejí tzv. **kódovou knihu**, která charakterizuje každou osobu.

Rozpoznávání osob podle hlasu

- Fáze testování:



vzdálenosti od kód. knih

Kódová kniha mluvčího m
plnými kolečky jsou
vyznačeny centroidy
prázdnými kolečky pozice
vektorů
tlustou čarou vzdálenosti od
centroidů (kvantizační chyba)

vyhledávání
nejbližších centroidů

vektory příznaků

vzorkovaný signál
(mluvčí m)

Rozpoznávání osob podle hlasu

- **Fáze trénování**

- Každá osoba v trénovací množině namluví několik minut řeči.
- Nahrávky se zparametrizují
- Vypočtou se příznakové vektory pro každých 10 ms řeči.
- Příznakové vektory vytvoří v obrazovém prostoru shluky, které se identifikují a reprezentují svými etalony.
- Tento proces se nazývá také jako vektorová kvantizace
- Pozice etalonů pak vytvářejí tzv. **kódovou knihu**, která charakterizuje každou osobu.

Další aplikace

- Shlukování dokumentů
 - Doporučení podobných, shlukování do kategorií
- Segmentace zákazníků
 - Vytvoření skupin na základě jejich nákupů
- Detekce pojistných podvodů
 - Včasná detekce podvodných vzorů
- Shlukování akcií
 - Vytvoření shluků vysoce korelovaných akcií
 - Pomůže diversifikovat portfolio
- Komprese obrazu



Užitečná literatura / kurzy

- [Unsupervised learning – Clustering](#)

