Katedra Geomatiky

Strojové učení v dálkovém průzkumu Země

MZ370G24
Lekce 5: Učení bez učitele
[shlukování, odhad hustoty, snížení dimenzionality]

Přednášející: Lukáš Brodský lukas.brodsky@natur.cuni.cz, Ondřej Pešek ondrej.pesek@fsv.cvut.cz a Martin Landa martin.landa@fsv.cvut.cz

Učení bez učitele (unsupervised learning)

- Algoritmy se snaží naučiti vzory (patterns) obsažené v datech
- Proč?
  - Řešíme jím problémy, ve kterých nejsou data označkována (labelled)
- Kvalita model
  - Absence referenčních dat pro posouzení kvality modelu
  - Nutno vyhodnocovati vizuálně

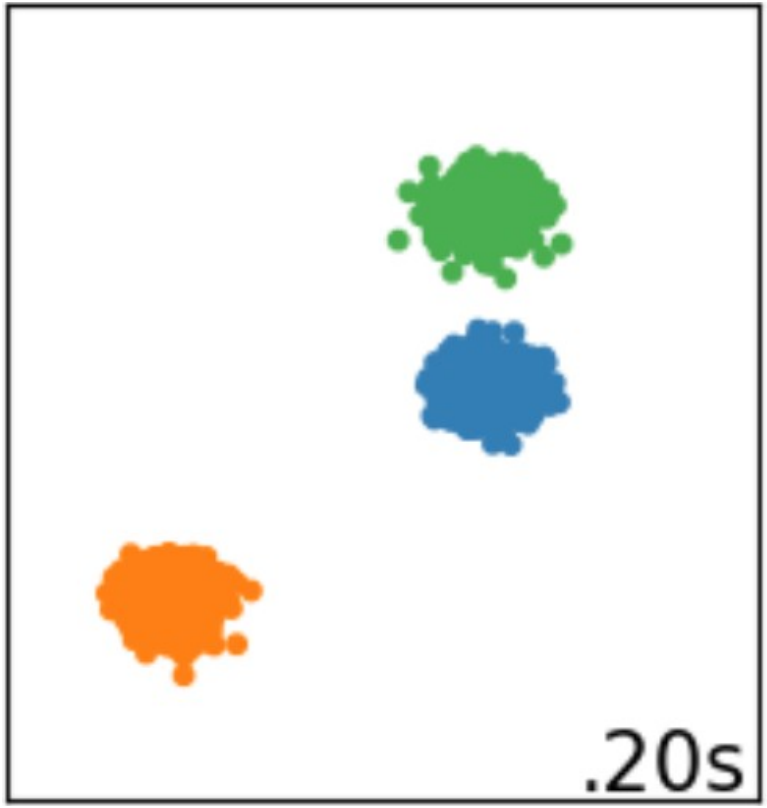
Algoritmy učení bez učitele

- "Učení bez učitele označuje pokusy o extrakci informací z rozdělení bez potřeby lidského značkování" (Goodfellow, 2015)
  - Pokus o zjednodušení reprezentace dat
- Typické příklady:
  - Shlukování
  - Odhad hustoty
  - Snížení dimenzionality

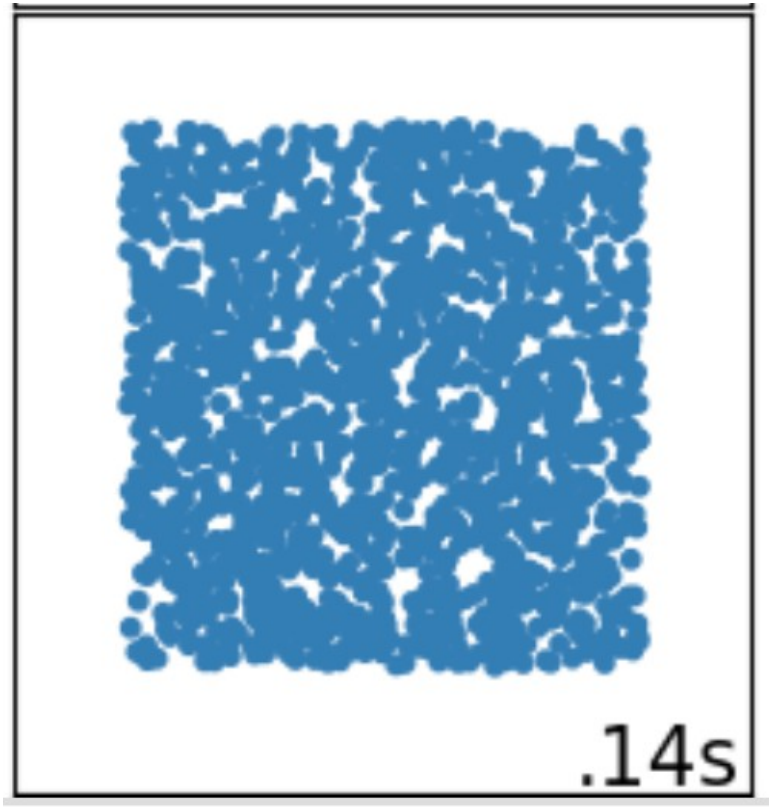
Shlukování (clustering)

Shlukování

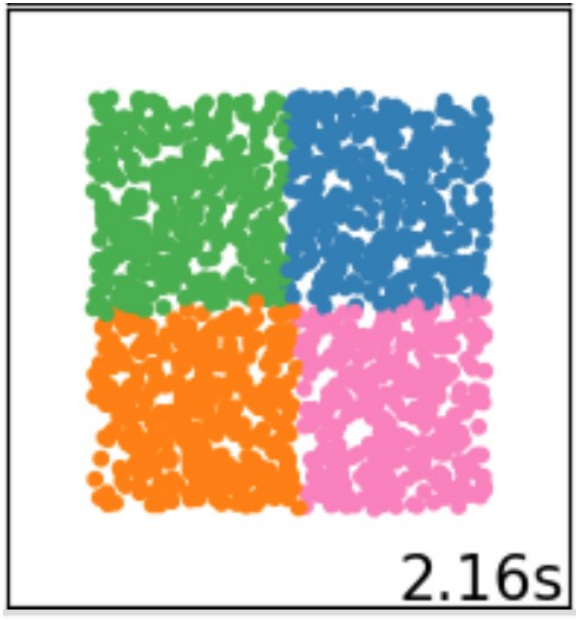
- Třídění dat takovým způsobem, aby objekty v jedné skupině (shluku) byly podobnější objektům v téže skupině než ve skupině odlišné

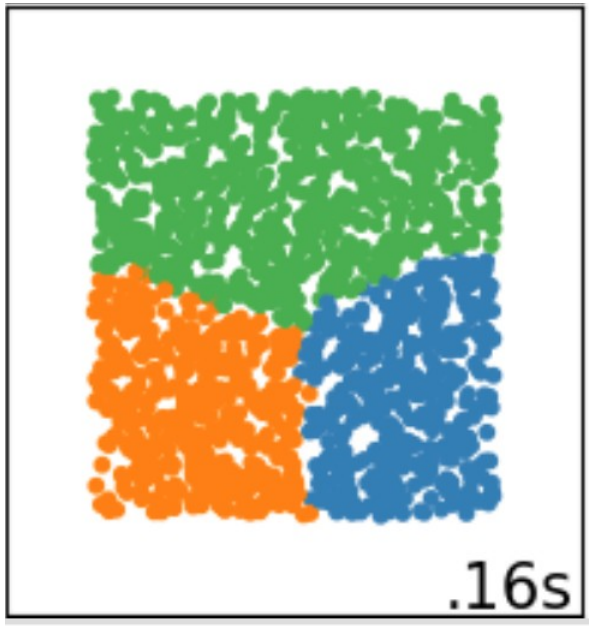


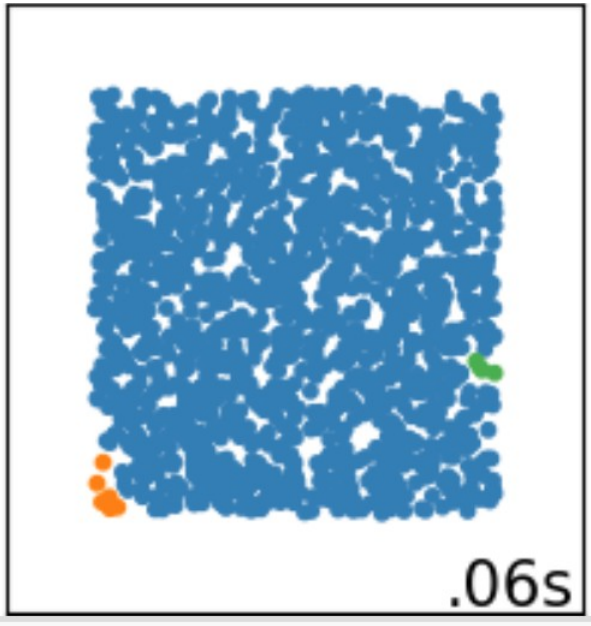
source: sklearn



source: sklearn







source: sklearn

Algoritmy shlukování

Hierarchické shlukování

- Tvorba shluků se vytváří jejich postupným dělením (divisivní metody) či spojováním (aglomerativní metody)
- Hierarchické shlukování si lze představiti ve tvaru stromu

Aglomerativní metody shlukování

---

1. přiřaď každému datu vlastní shluk
2. vypočti podobnost mezi jednotlivými shluky
3. spoj dva nejpodobnější shluky

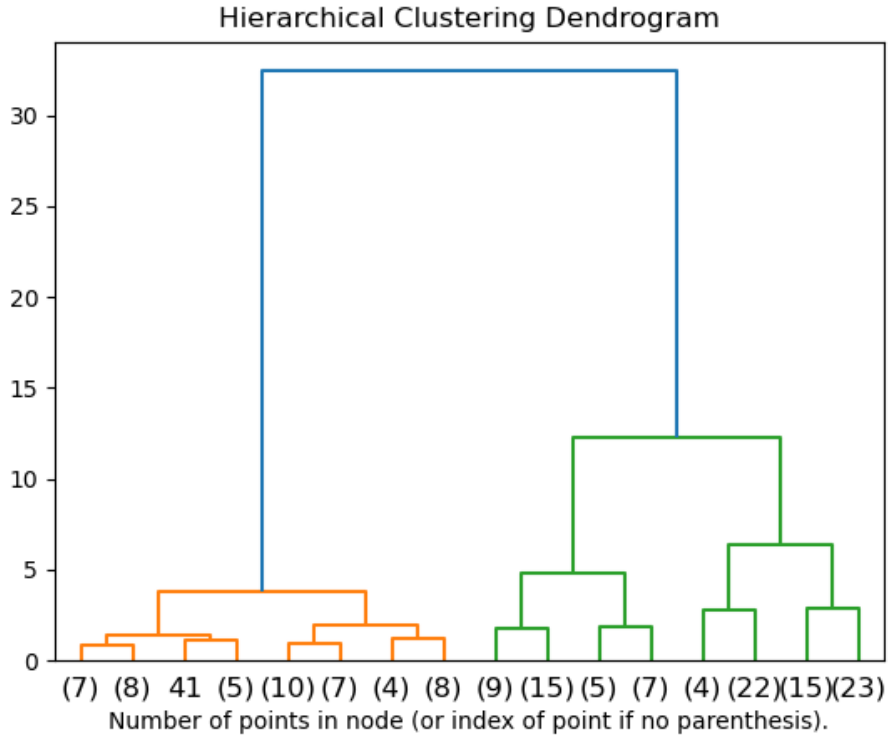
Opakuj 2 a 3:

dokud neexistuje pouze jeden shluk

Kritéria: průměrné/maximální vzdálenosti mezi daty

Sklearn API: from sklearn.cluster import AgglomerativeClustering

aglomerativní



divisivní

Divisivní shlukování

---

1. přiřaď všechna data k jednomu shluku
2. rozděl shluk na dva nejméně podobné

Opakuj rekurzivně na každém shluku:

dokud nemá každé datum přiřazený vlastní shluk

K-Means

- rozdělí data do množiny shluků
  - jednotlivé shluky jsou hyperkoule
- rozdělí *n* dat do *k* shluků, kde každé datum náleží do shluku, jehož průměr je mu neiblíže
  - $\sum_{i=0}^{n} \min_{\mu_j \in C}(||x_i - \mu_j||^2)$) nalezení k-rozměrného one-hot vektoru h reprezentujícího vstup x
- nedeterministický

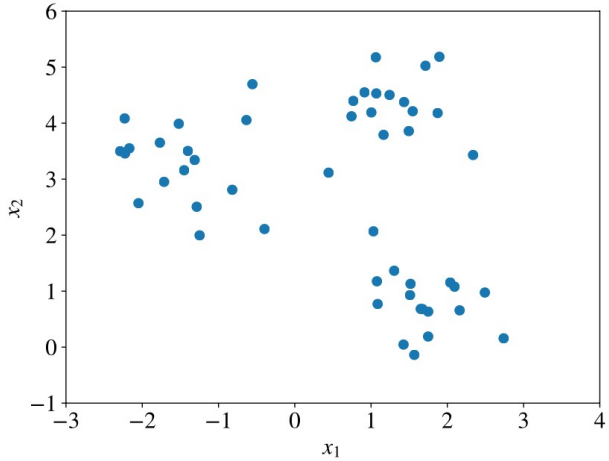
K-Means

---

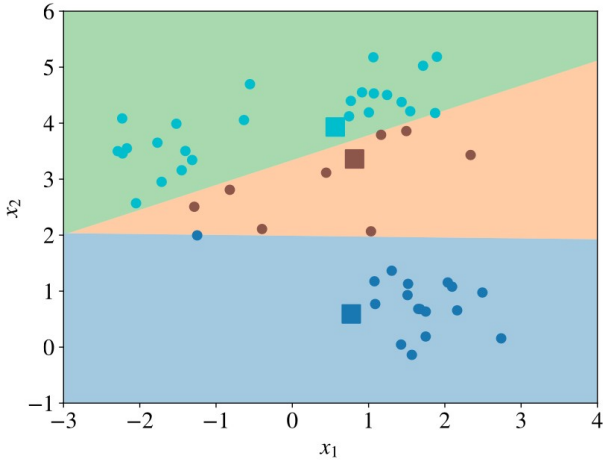
1. zvol *k* - počet shluků;
2. inicializace: náhodně rozlož *k* vektorů hodnot (centroidů) do prostoru hodnot

Opakuj:

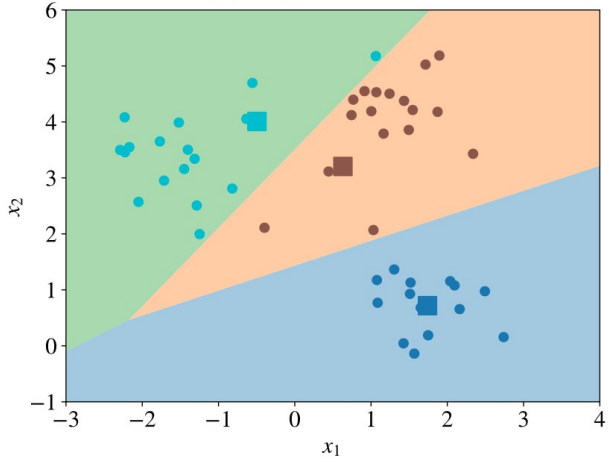
3. vypočti euklidovskou vzdálenost každého *x* od centroidu *c*;
4. každému datu *x* přiřaď nejbližší *c*;
5. pro každé *c* vypočti průměrný vektor hodnot pro data k němu přiřazená



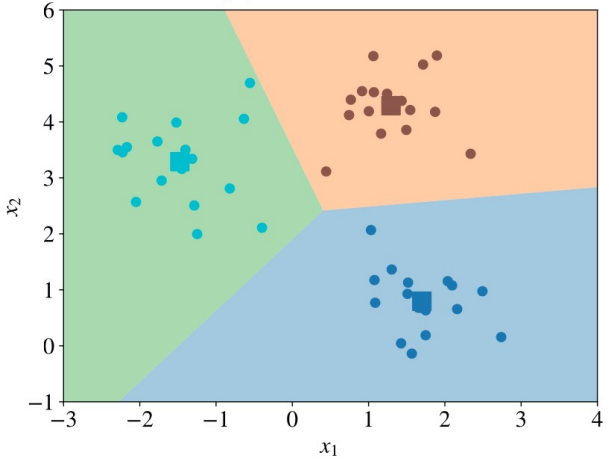
(a) original data



(b) iteration 1



(c) iteration 3



(d) iteration 5

K-Means

- pozice prvotních centroidů ovlivňuje výsledek
  - dvojí spuštění může vésti ke dvěma výsledkům
  - chybí reprodukovatelnost
- vhodná hodnota hyperparametru *k* musí býti uživatelem nalezena
  - existují “triky”, jak vhodnou hodnotu odhadnout
  - zkušenost a znalost dat pomáhají ke “kvalifikovanému odhadu”

Rozhodování o počtu shluků

Viz shlukování jako problém učení s učitelem, ve kterém musíme také odhadnout správné označkování dat *(Tibshirani and Walther, 2013)*

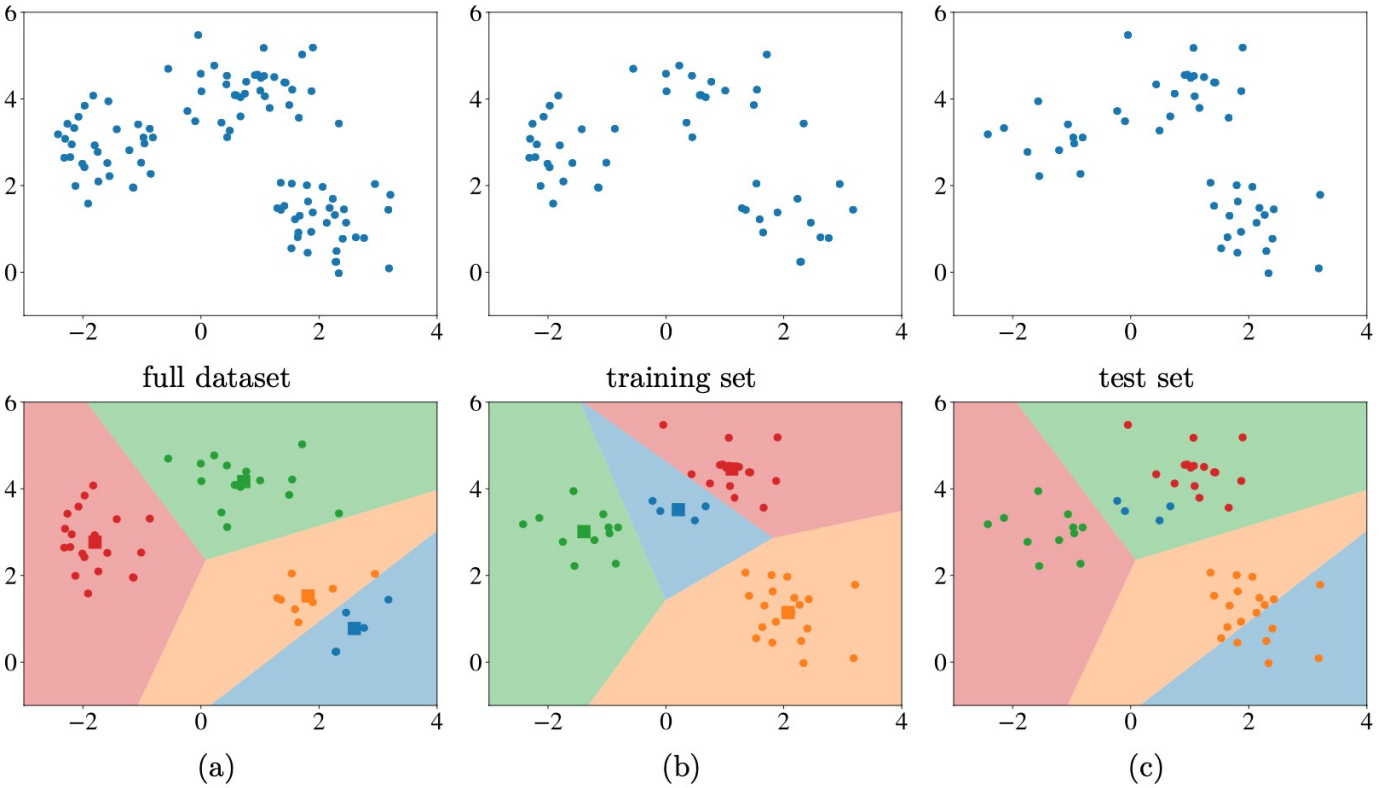
- rozdělme data do trénovacího (S_{tr}) a testovacího (S_{te}) datasetu s N_{tr} a N_{te} prvky;
- zvolme počet shluků k;
- spusťme shlukovací algoritmus C na S_{tr} a S_{te} – získáme výsledky shlukování $C(S_{tr}, k)$ a $C(S_{te}, k)$;
- Vypočtěmě [N_{tr} x N_{te}] matici společného členství (co-membership matrix)

  $D[A, S_{te}]^{(i, i')} = 1$ … pokud příklad x_i a $x_{i'}$ náleží ke stejnému shluku

  a = 0 jinak;

- pokud k dává smysl, $C(S_{te}, k)$ nejspíše náleží k týmž shlukům jako $C(S_{tr}, k)$

Rozhodování o počtu shluků



Rozhodování o počtu shluků

Síla predikce (prediction strength)

$$\mathrm{ps}(k) = \min_{1 \le j \le k} \frac{1}{n_{kj}(n_{kj}-1)} \sum_{i \neq i' \in A_{kj}} D[C(X_{\mathrm{tr}}, k), X_{\mathrm{te}}]_{ii'}.$$

D .. matice společného členství

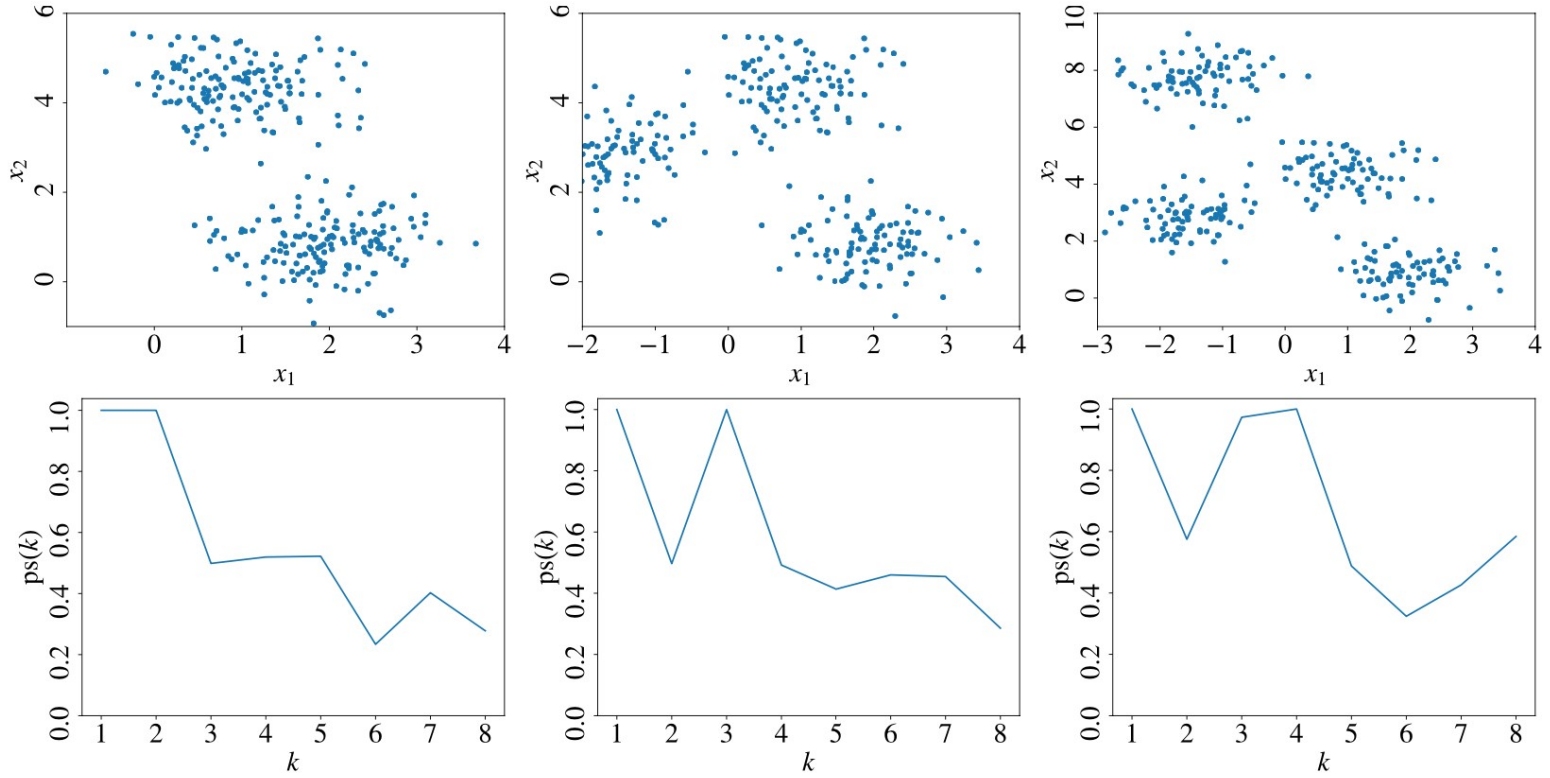
k .. počet shluků,

n_{kj} .. množství dat ve shlucích,

A_{kj} .. indexy příkladů (observací) v testovacích shlucích 1, 2…k.

Empirie: ps(k) > 0.8 je dostatečné

Rozhodování o počtu shluků



DBSCAN

- k-means a podobné algoritmy jsou založené na centroidech, DBSCAN je založen na hustotě rozdělení
- namísto odhadu počtu shluků definujeme dva hyperparametry: *epsilon* a *ni*

DBSCAN

---

1. i = 1
2. vyber náhodné datum z *x* a přiřaď jej ke shluku *i*;
3. spočti množství dat ve vzdálenosti nanejvýš *epsilon* od vybraného data;
4. pokud je počet >= *ni*, přiřaď tato data ke stejnému shluku;
5. opakuj body 3-4 pro každého nového člena shluku *i*;
6. pokud je ve shluku *i* pouze jeden člen, označ jej jako odlehlou hodnotu;
7. pokud nelze do shluku *i* přidat další členy a existují data nepřiřazená ke shluku, *i* += 1 a vrať se na bod 2;

DBSCAN

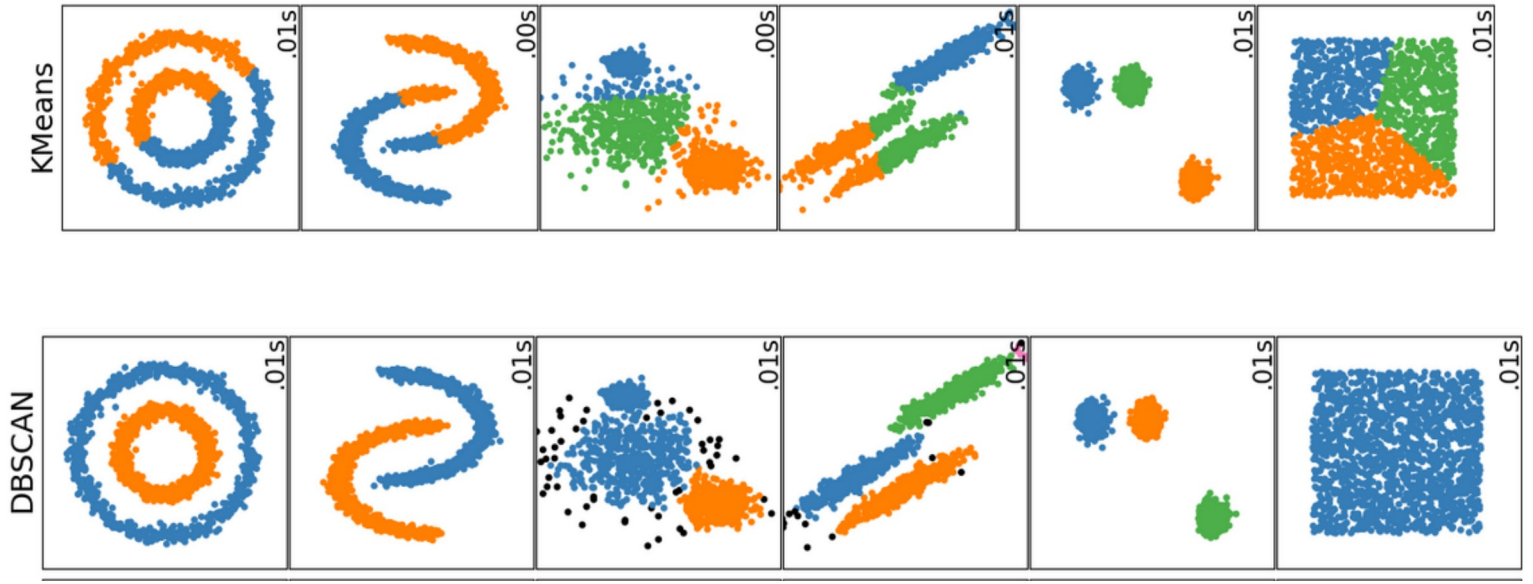
Výhody:

- tvary shluků jsou flexibilnější

Nevýhody:

- arbitrární volba *epsilon* a *ni*
- pokud existují shluky rozdílných hustot, algoritmus může data rozdělit do příliš mnoha shluků
    - řešení: HDBSCAN (volí se pouze parametr *ni*)

Porovnání K-Means a DBSCAN



Hodnocení kvality shlukování

Hodnocení kvality shlukování

- Metrika podobnosti či nepodobnosti
  - Např. vzdálenost mezi body uvnitř shluku
- Příklady:
  - Koeficient siluety (silhouette coefficient)
  - Dunnův index (Dunn’s index)

Hodnocení kvality shlukování - Koeficient siluety

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

počítáme průměr koeficientu siluety pro každý bod

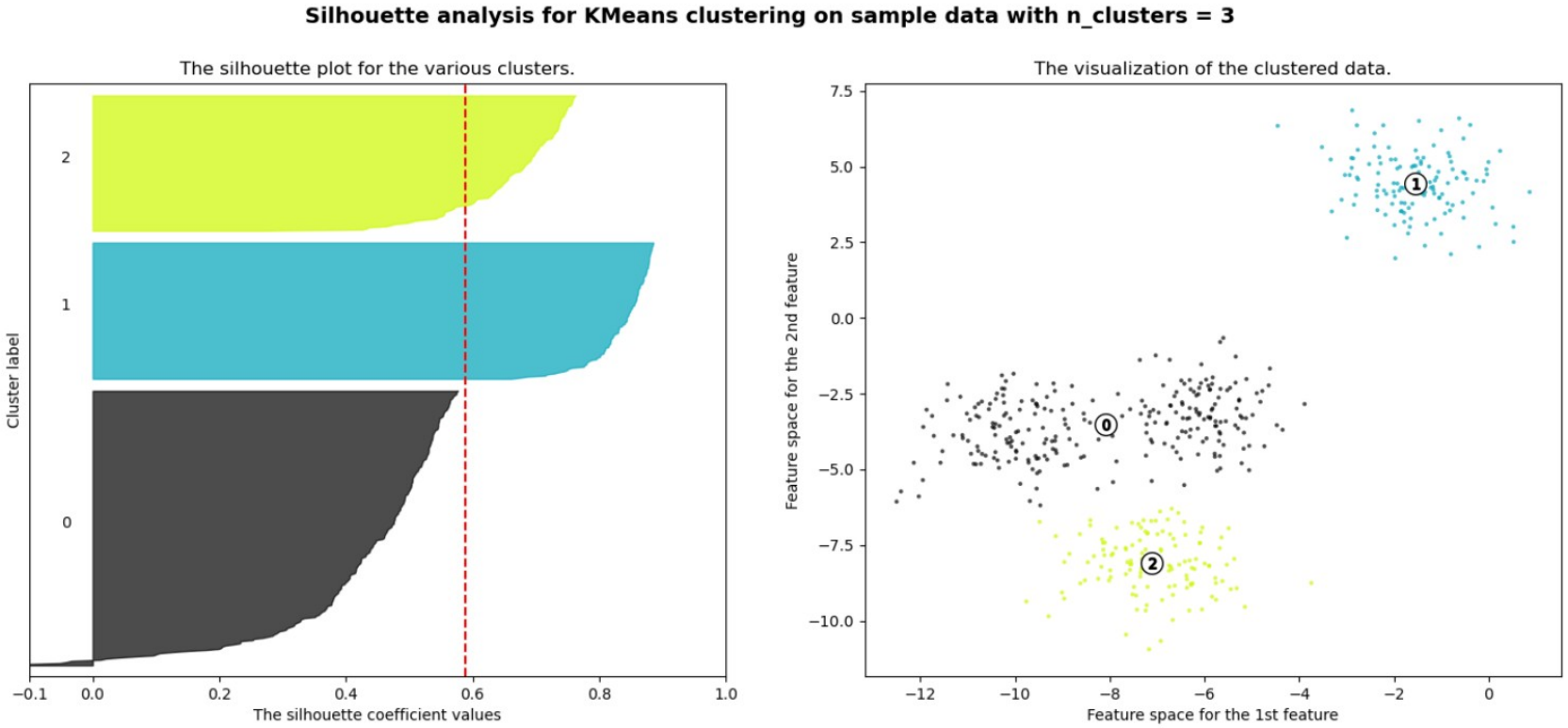
a .. průměrná vzdálenost mezi vybraným bodem a všemi dalšími body v tomtéž shluku

b .. průměrná vzdálenost mezi vybraným bodem a všemi dalšími body v nejbližším shluku

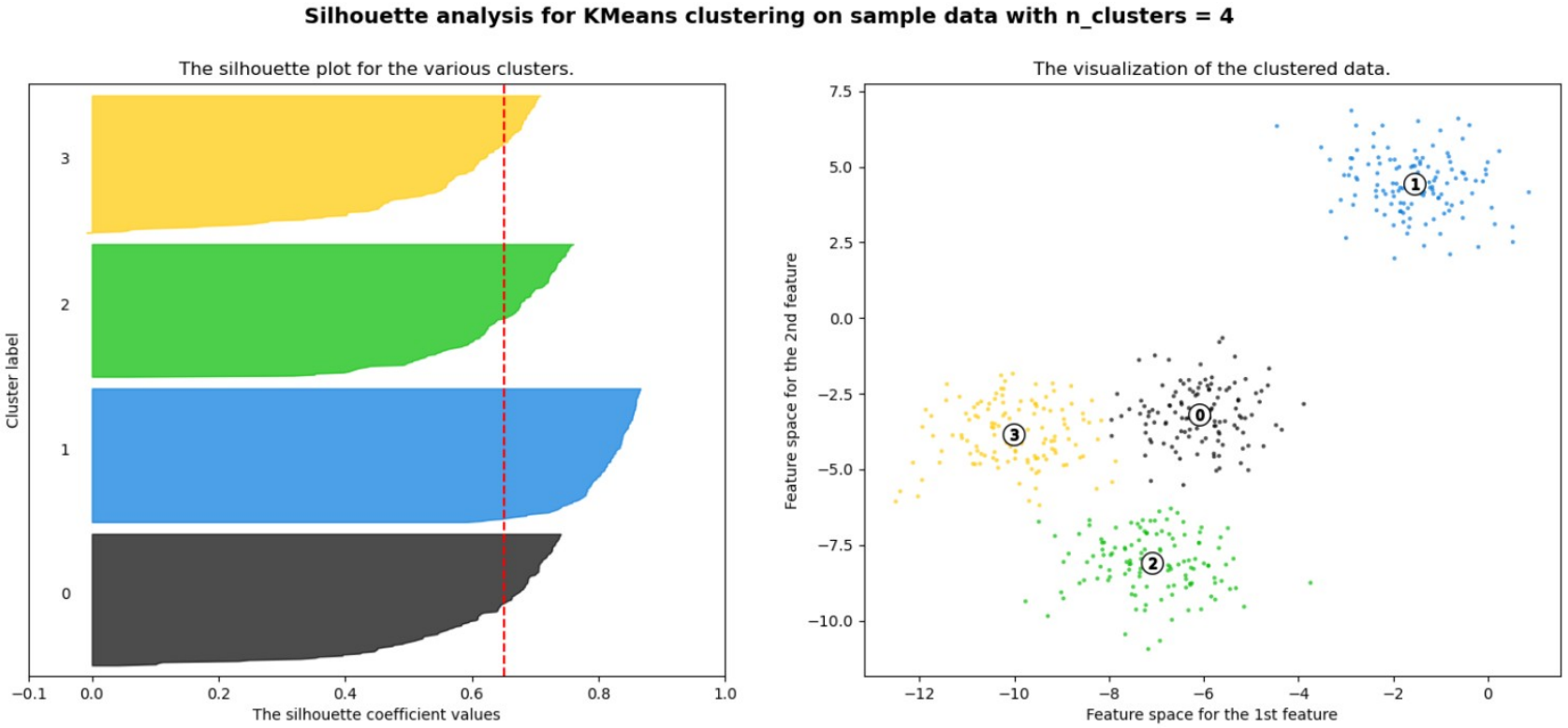
Hodnocení kvality shlukování - Koeficient siluety

- Rozmezí hodnot <-1; 1>
- Hodnoty blížící se 1 – bod je daleko od sousedícího shluku
- Hodnoty blížící se 0 – bod je blízko rozhodovací hranici mezi dvěma sousedícími shluky
- Hodnoty blížící se -1 – bod byl přiřazen k nesprávnému shluku

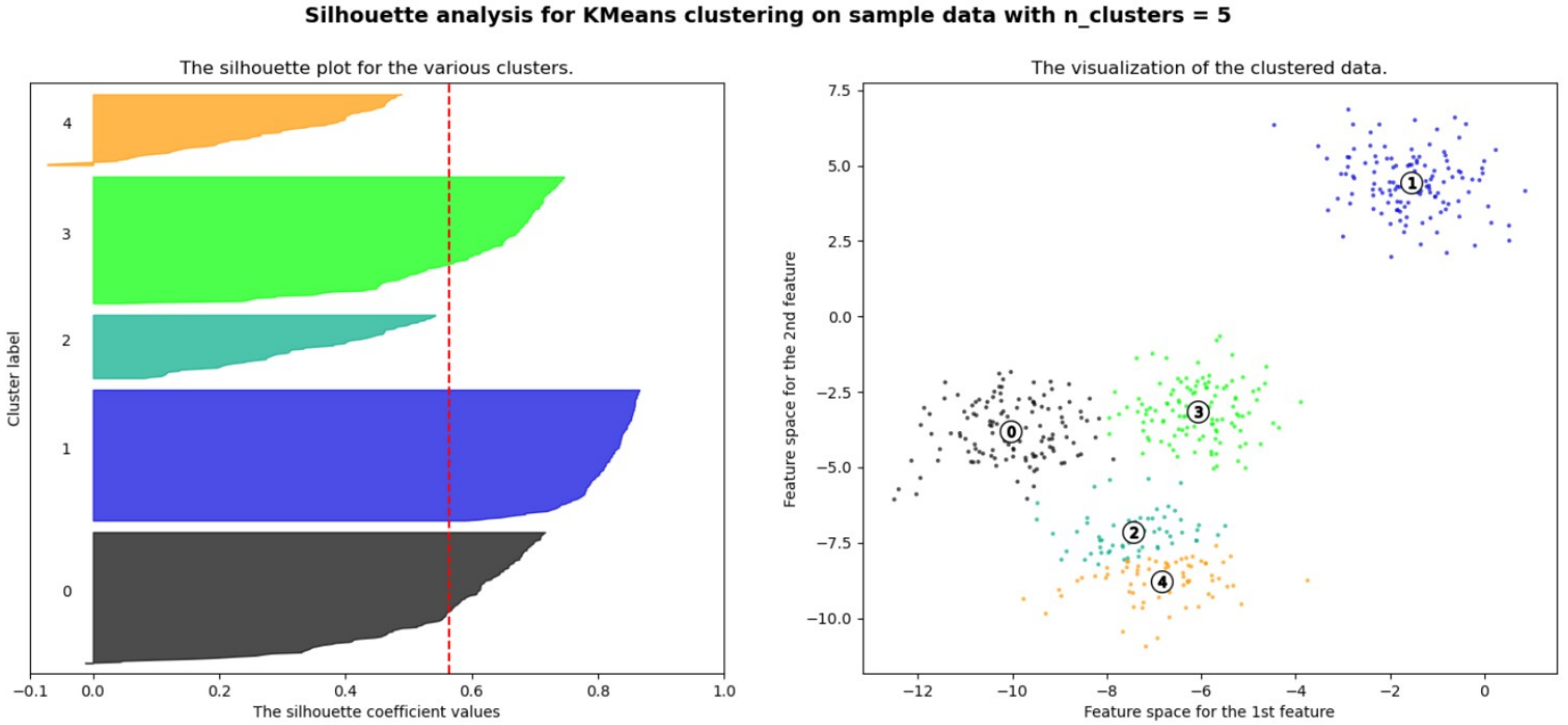
Hodnocení kvality shlukování - Koeficient siluet



Hodnocení kvality shlukování - Koeficient siluet



Hodnocení kvality shlukování - Koeficient siluet



Hodnocení kvality shlukování - Koeficient siluet

Sklearn API:

```
from sklearn.metrics import silhouette_samples, silhouette_score


# The silhouette_score gives the average value for all the samples.
silhouette_avg = silhouette_score(X, cluster_labels)


# Compute the silhouette scores for each sample
sample_silhouette_values = silhouette_samples(X, cluster_labels)
```

Hodnocení kvality shlukování - Dunnův index

$$DI_m = \frac{\min\limits_{1 \leqslant i < j \leqslant m} \delta(C_i, C_j)}{\max\limits_{1 \leqslant k \leqslant m} \Delta_k}$$

- snažíme se maximalizovat minimum vzdáleností uvnitř shluku dělené velikostí největšího shluku
- větší vzdálenosti uvnitř shluku -> lepší separace dat
- menší velikost shluků -> kompaktnější shluky

Není v sklearn:

```
from jqmcvi import base

base.dunn(cluster_list)
```

Odhad hustoty

Odhad hustoty

- Problém: modelování hustoty pravděpodobnosti (probability density function, pdf) neznámého rozdělení pravděpodobnosti, z kterého dataset pochází
- Využití: detekce novot a odlehlých měření
- Příklad: systém pro odhalení průniku (intrusion detection system, IDS)

Odhad hustoty

- Nechť $\{x_i\}^N$ jest jednorozměrný dataset,
- Data v *x* pocházejí z neznámého rozdělení pravděpodobnosti *R*
- Naším úkolem je modelovat tvar rozdělení pravděpodobnosti *f* pro jádrový model *k*:

$$\hat{f}_b(x) = \frac{1}{Nb} \sum_{i=1}^{N} k\left(\frac{x - x_i}{b}\right),$$

kde *b* je hyperparametr kontrolující výměnu mezi bias a variancí

Odhad hustoty

- Stejně jako pro SVM můžeme použít Gaussovské jádro:

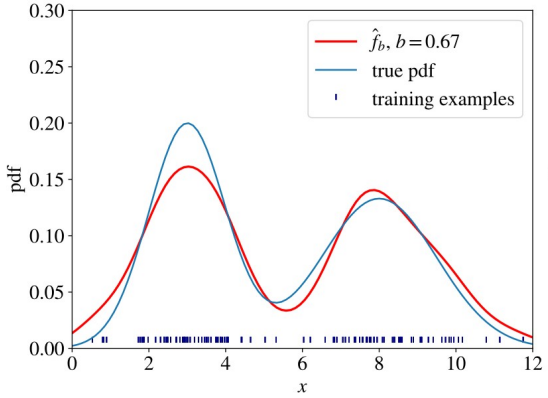
$$k(z) = \frac{1}{\sqrt{2\pi}} \exp\left(\frac{-z^2}{2}\right).$$

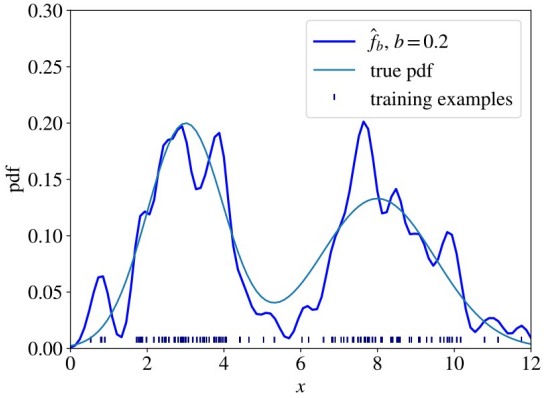
- Hledáme takové *b*, které minimalizuje rozdíl mezi skutečným tvarem *f* a tvarem modelu f_b.
- Rozumnou volbou metriky je střední integrovaná kvadratická chyba (mean integrated squared error - MISE):

$$\mathrm{MISE}(b) = \mathbb{E}\left[\int_{\mathbb{R}} (\hat{f}_b(x) - f(x))^2 \, dx\right]$$

Odhad hustoty

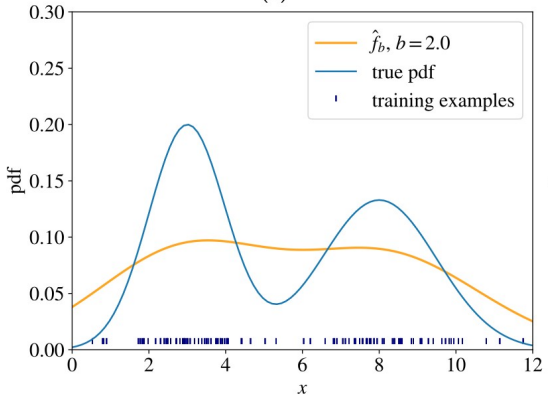
good fit

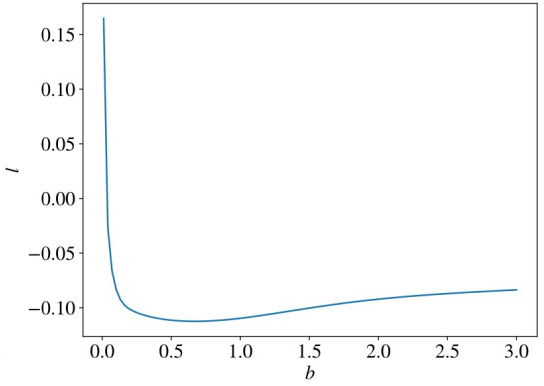




overfitted

under-fitted





grid search

Redukce dimenzionality

Analýza hlavních komponent

viz přednáška 2 - Lineární algebra a numerický Python

Outliers detection:

https://scikit-learn.org/stable/modules/outlier_detection.html