Modifikace algoritmu FEKM

Marta Žambochová

Katedra matematiky a informatiky
Fakulta sociálně ekonomická
Univerzita J. E. Purkyně v Ústí nad Labem

ROBUST
9.– 14. září 2012
Němčičky

Motivace

- Potřeba metod pro shlukovou analýzu dat (velmi) velkých datových souborů
- Minimalizace počtu průchodů celým datovým souborem
- Využití výběrového souboru dat, respektive speciálně vytvořeného souboru reprezentujícího původní datový soubor

Algoritmus FEKM

- Fast and Exact (Out-of-Core) K-Means
- Goswami, A., Ruoming J., Agrawal,G.
- 2004
- Algoritmus DFEKM
  - Distributed Fast and Exact (Out-of-Core) K-Means
- 2006

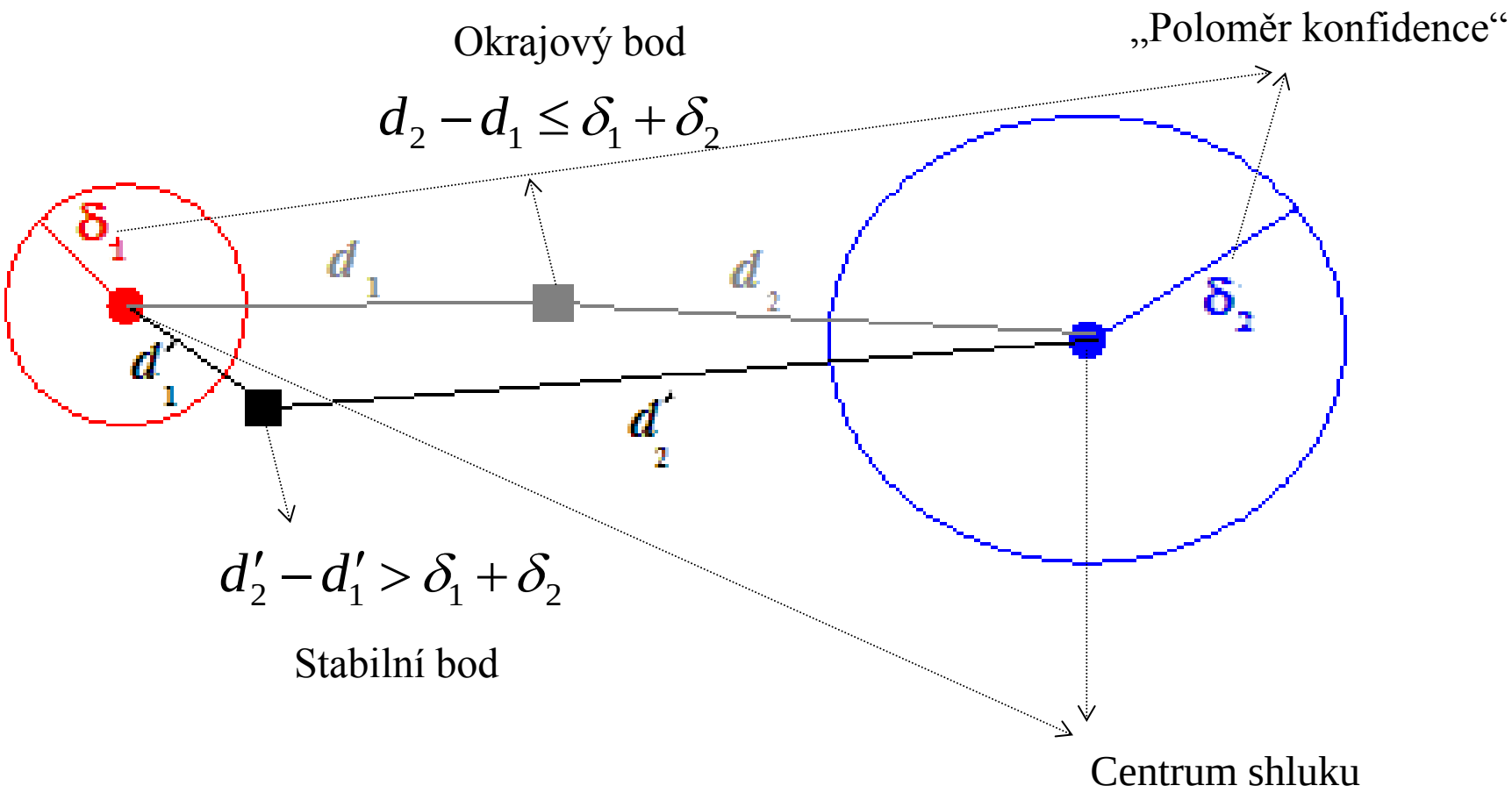
Postup zpracování – I.fáze

- Prvotní vytvoření přiměřeně velkého výběrového souboru z původního souboru dat
- V rámci tohoto výběrového souboru jsou vytvořeny shluky pomocí klasického algoritmu *k*-průměrů
- V každé iteraci se zaznamená všech *k* center a k nim popisné statistiky shluku

Postup zpracování – II.fáze

- V druhé fázi algoritmus prochází celý datový soubor
- Každý datový objekt se pro každou iteraci přiřadí do určitého shluku (k nejbližšímu centru dané iterace)
- Problém chybného zařazení do shluku se týká především objektů ležících na okraji shluků

Okrajové body



Postup zpracování – III.fáze

- Ve třetí fázi se algoritmus zabývá podezřelými okrajovými body, které odhalila a uložila předchozí fáze
- Provádí se přepočet s využitím uložených statistik popisujících každý jednotlivý shluk a podezřelých okrajových objektů.
- Pokud existuje přepočtené centrum, které je od původního více vzdálené, než předem zadaná kritická hodnota, vrací se algoritmus do druhé fáze a probíhá opětovný průchod celým datovým souborem

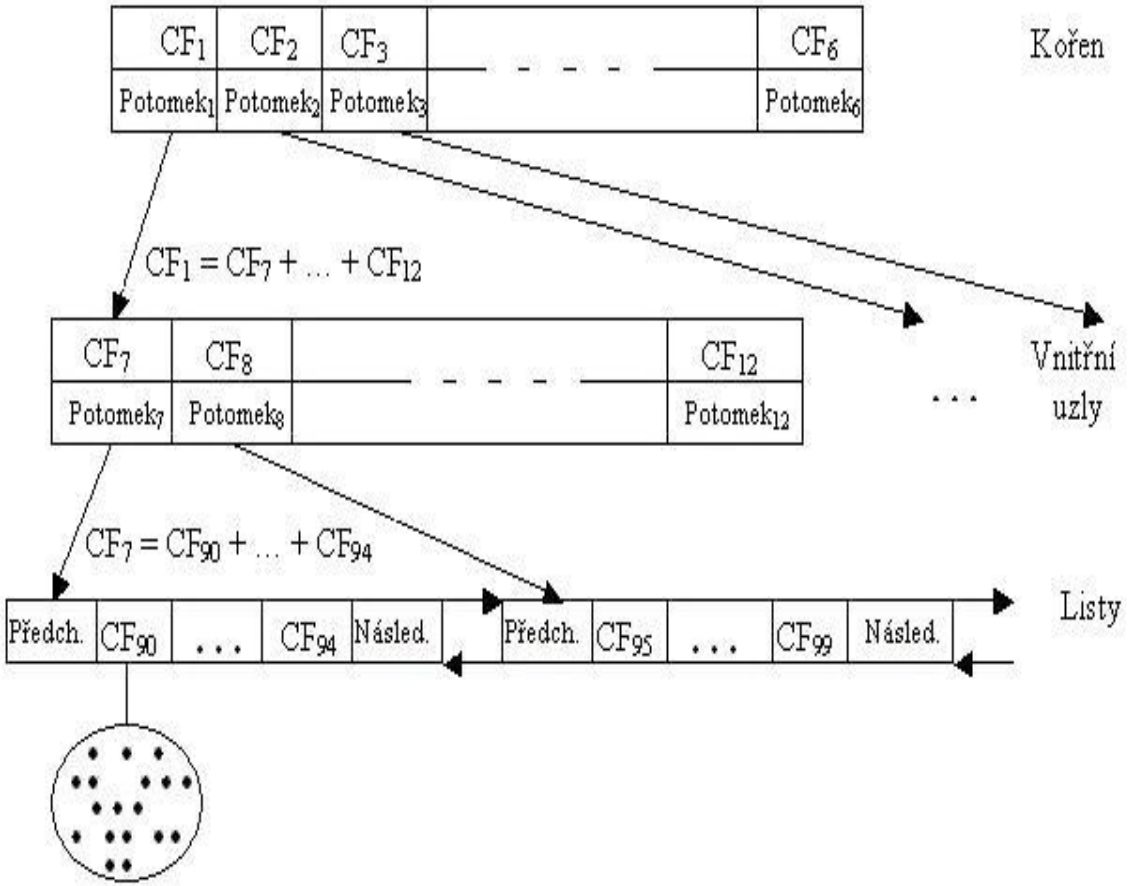
Metody BIRCH

(Balanced Iterative Reducing and Clustering using Hierarchies)

- algoritmus BIRCH
  - Zhang, T., Ramakrishnan, R., Livny, M. (1996)
  - CF-stromy
  - CF-charakteristika shluku (Clustering Feature) je uspořádaná trojice $CF = (N, LS, SS)$, kde N je počet objektů ve shluku, LS je vektorovým součtem souřadnic všech objektů ve shluku a SS je vektorovým součtem druhých mocnin souřadnic těchto objektů
- algoritmus BIRCH *k*-průměrů
  - Bradley, P., Fayyad, U., Reina, C. (1998)
  - uspořádaná trojice údajů (m, q, b), kde m je velikost daného shluku, q je kvalita daného shluku (součet druhých mocnin vzdáleností centroidu od všech objektů ve shluku) a b je centroid shluku
- z trojic (N, LS, SS) a (m, q, b) můžeme získat identické informace

BIRCH

- CF - stromy



BIRCH

- vytvoření CF-stromu postupným zařazením datových objektů
- (kondenzace vytvořeného CF-stromu a optimalizace jeho velikosti)
- shlukování listových vrcholů pomocí aglomerativního hierarchického algoritmu shlukování
- (přerozdělení objektů k jejich nejbližším centrům, a tím získání nového složení shluků)

BIRCH *k*-průměrů

- dva parametry
  - povolený počet objektů v libovolném shluku
  - hranice variability libovolného shluku („poloměr“ shluku)
- upravená varianta algoritmu BIRCH
- nevytváří CF-strom
- postup:
  - první fáze
    - v cyklu se vždy vybere objekt z množiny objektů a nalezne se „nejbližší“ shluk z množiny shluků, ve kterém se přidáním objektu nepřekročí hranice pro počet prvků ve shluku ani hranice variability shluku
    - pokud takovýto shluk neexistuje, vytvoří se pro objekt nový shluk, který obsahuje pouze tento objekt
    - po zařazení objektu do shluku se objekt vymaže z množiny všech objektů
    - cyklus se provádí do vyprázdnění množiny objektů
  - druhá fáze
    - shlukování centroidů všech shluků, které vznikly ve fázi první
  - třetí fáze
    - všechny původní objekty jsou rozřazeny do shluků, každý objekt je přiřazen k nejbližšímu z centroidů vzniklých ve druhé fázi
- algoritmus zatím není implementován v žádném dostupném programovém systému

Data I.

- soubor IRIS
- http://archive.ics.uci.edu/ml/datasets/
- 150 objektů
- čtyři numerické proměnné (jednotlivé rozměry kališních a korunních lístků květů)
- tři shluky - tři různé druhy z rodu iris, z každého druhu 50 zástupců
- z průzkumové analýzy je zřejmé, že jeden z druhů se výrazně odlišuje v popsaných atributech, zbývající dva se odlišují nevýrazně, dokonce dochází k prolínání

Data II.

- soubor VOWEL
- 528 objektů
- 10 numerických proměnných
- jedenáct shluků - 11 jednoslabičných slov
- Obsah souboru se týká výslovnosti samohlásek v britské angličtině. Osm mluvčích (čtyři muži a čtyři ženy) přečetlo šestkrát 11 jednoslabičných slov, která se lišila výslovností samohlásky (heed, hid, head, had, hard, hud, hod, hoard, hood, who`d, heard). Každé slovo tedy bylo proneseno 48 krát. V každém z uvedených případů byl zaznamenán zvuk a převeden do deseti numerických hodnot. Každý záznam je jedním objektem výsledného souboru.

Data III.

- soubor GENER
- generovaný soubor
- 1 000 000 objektů
- dvě numerické proměnné
- 20 shluků
- Souřadnice objektů jednotlivých shluků byly generovány jako náhodné hodnoty náležící normálnímu rozdělení daných parametrů. Parametry rozdělení v jednotlivých shlucích byly generovány opět náhodně jako hodnoty rovnoměrného rozdělení; střední hodnota μ náhodně z intervalu (0, 10) a rozptyl σ^2 z intervalu (0, 3).

Shrnutí I.

- Algoritmus FEKM přináší výsledky stejné kvality jako originální algoritmus *k*-průměrů
- Zvyšování efektivity snižováním počtu průchodů datovým souborem
- Nevýhoda algoritmu FEKM =
  velmi malý počet průchodů celým souborem pouze ve výjimečných případech, závisí na prvotním vzorku dat
- Algoritmus BIRCH vytváří pomocí jednoho průchodu celým datovým souborem výsledky poněkud horší kvality než algoritmus *k*-průměrů

Shrnutí II.

- Předřazením I.fáze algoritmu BIRCH *k*-průměrů před vlastní zpracování algoritmu FEKM vznikla metoda spojující výhody obou algoritmů
- Centroidy shluků vzniklých v I.fázi algoritmu BIRCH byly vzaty jako vzorek datového souboru
- Oproti vlastnímu algoritmu FEKM došlo ke stabilizování nutného počtu průchodů celým datovým souborem
  - Z podstaty modifikace vyplývá, že minimální počet průchodů je dva, v originálním algoritmu existují případy, kdy postačil pouze jediný průchod
  - V originálním algoritmu se vyskytly případy, kdy byl vzorek dat vygenerován „nevhodně“ a došlo k výcečetnému průchodu datovým souborem
- Oproti algoritmu BIRCH *k*-průměrů došlo ke zlepšení kvality, ovšem prodloužení doby zpracování (průměrně na 2-3 násobek)

Děkuji za pozornost