Slovenská technická univerzita

Fakulta informatiky a informačných technológií

Ilkovičova 3, 842 19 Bratislava 4

Sakalosh Oleksandr

Klastrovanie

Študijný program: Informatika

Ročník: 3.

Predmet: **Umelá inteligencia**

Ak. rok: 2021/2022

Zadanie

Vašou úlohou je naprogramovať zhlukovač pre 2D priestor, ktorý zanalyzuje 2D priestor so všetkými jeho bodmi a rozdelí tento priestor na k zhlukov (klastrov). Implementujte rôzne verzie zhlukovača, konkrétne týmito algoritmami:

- k-means, kde stred je centroid
- k-means, kde stred je medoid
- aglomeratívne zhlukovanie, kde stred je centroid
- divízne zhlukovanie, kde stred je centroid

Vyhodnocujte úspešnosť/chybovosť vášho zhlukovača. Za úspešný zhlukovač považujeme taký, v ktorom žiaden klaster nemá priemernú vzdialenosť bodov od stredu viac ako 500.

Vizualizácia: pre každý z týchto experimentov vykreslite výslednú 2D plochu tak, že označkujete (napr. vyfarbíte, očíslujete, zakrúžkujete) výsledné klastre.

Algoritmy

K-means

Pri k-means ako centroidy vyberiem 20 náhodných bodov. Následne priradím každý bod najbližšiemu klastru, čim si vytvorím prvé ohodnotenie bodov. Po ohodnotení bodov vyrátam nový stred klastra. Ak má byť stred centroid, tak ho vyrátam ako priemernú vzdialenosť všetkých x-ových súradníc, čím dostanem x-ovú súradnicu centroidu a rovnako vyrátam priemer všetkých y-ových súradníc, a tak získam y-ovú súradnicu centroidu. Centroid je iba fiktívny stred, tento bod reálne vykreslený nie je. Naopak ak má byť stred medoid, tak toto je už reálny bod, ktorý môžeme povedať, že to je najbližší bod k všetkým bodom, teda bod s najmenším súčtom Manhattan vzdialeností k všetkým ostatným bodom. Keď už mám aktualizovaný stred klastra, tak znova priradím všetky body k danému klastru.

Divízne zhlukovanie

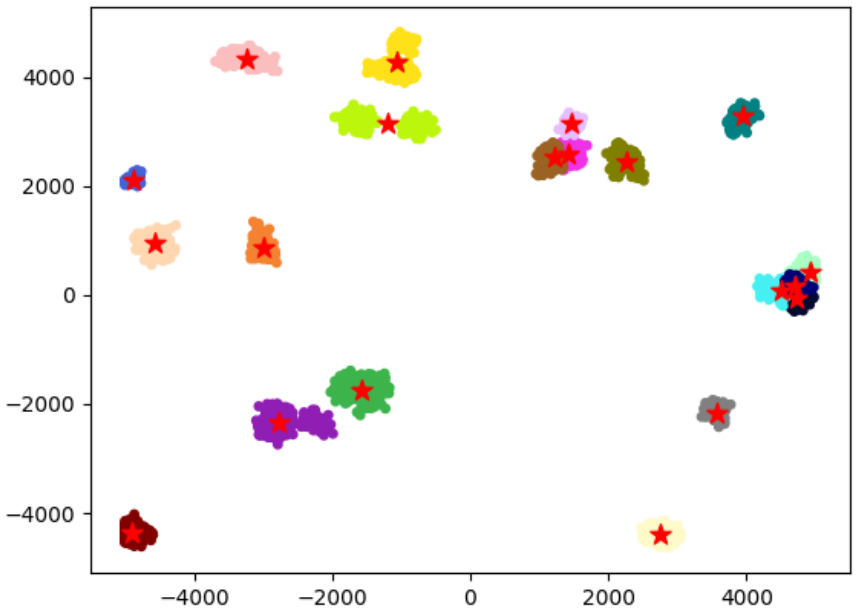
V divíznom zhlukovaní využívam k-means algoritmus so stredom ako centroid. Na začiatku mám 1 klaster, ktorý rozdelím na 2 klastre. Následne ak ešte nemám požadovaný počet klastrov, tak vyberiem klaster, ktorý má najväčší priemer vzdialeností v rámci klastra a ten znova rozdelím na 2 klastre. Tento cyklus sa opakuje až pokiaľ nedosiahnem zadaný počet klastrov.

Aglomeratívne zhlukovanie

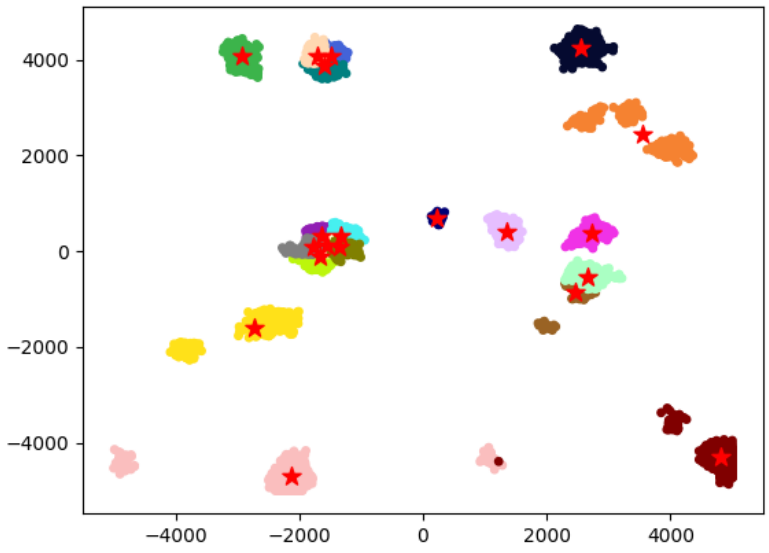
Aglomeratívne zhlukovanie funguje naopak ako divízne, keďže teraz máme na začiatku nbodov, pričom každý bod tvorí samostatný klaster. Na začiatku si teda vytvorím n-klastrov, a následne si vytvorím maticu vzdialeností medzi dvoma klastrami. Následne nájdem klastre, ktoré sú k sebe najbližšie a tie spojím do jedného. Pôvodné klastre odstránim ako z matice vzdialeností, tak aj z poľa, kde mám uložené všetky klastre. Nový klaster, ktorý vznikol spojením, pridám na miesto prveho klastra z dvoch vybraných. Tento cyklus spájania vykonávam pokiaľ nemám počet klastrov rovný požadovanému počtu klastrov.

Vizualizácia jednotlivých typov klastrovania

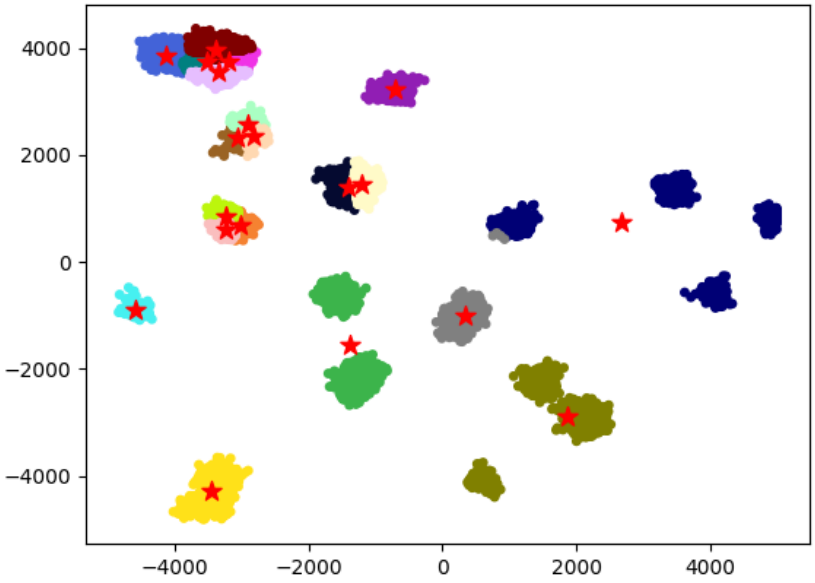
K-means, kde stred je centroid



5000 bodov

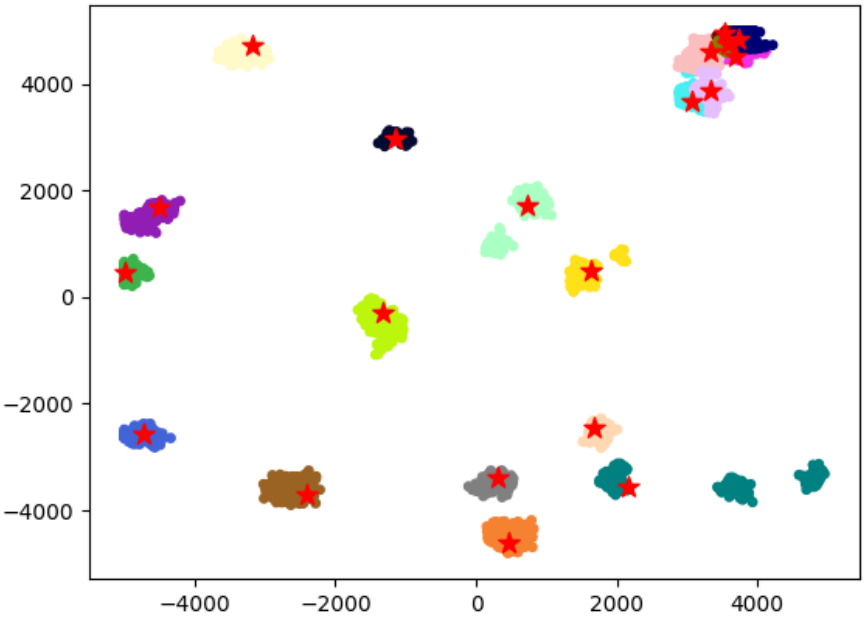


10000 bodov

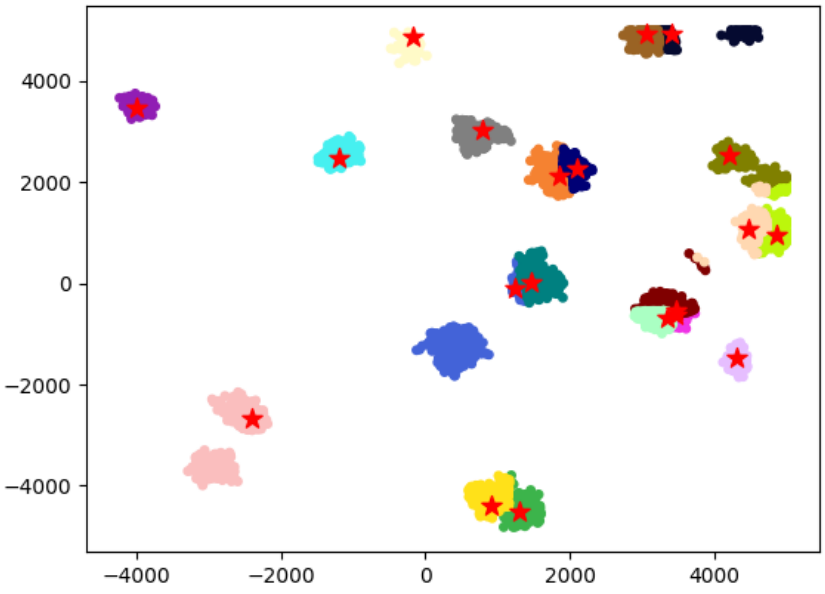


20000 bodov

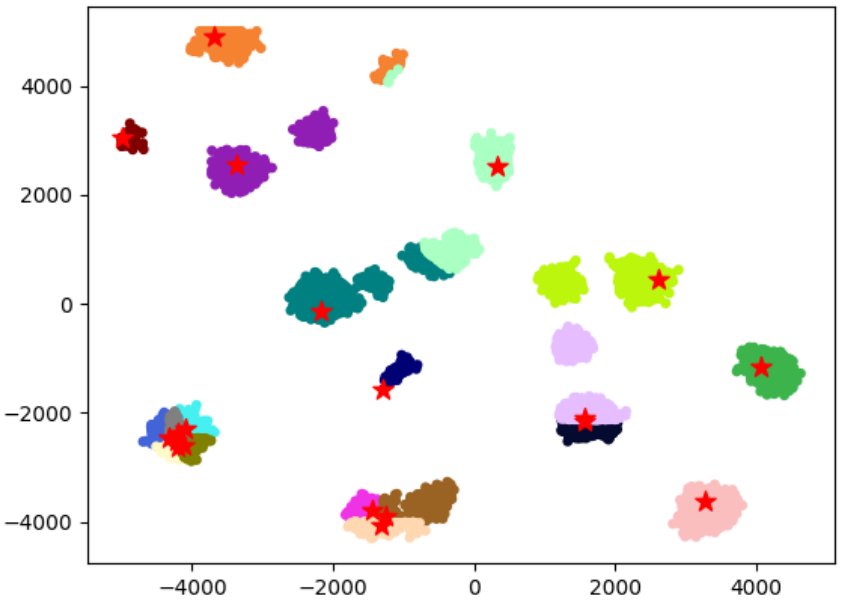
K-means, kde stred je medoid



5000 bodov

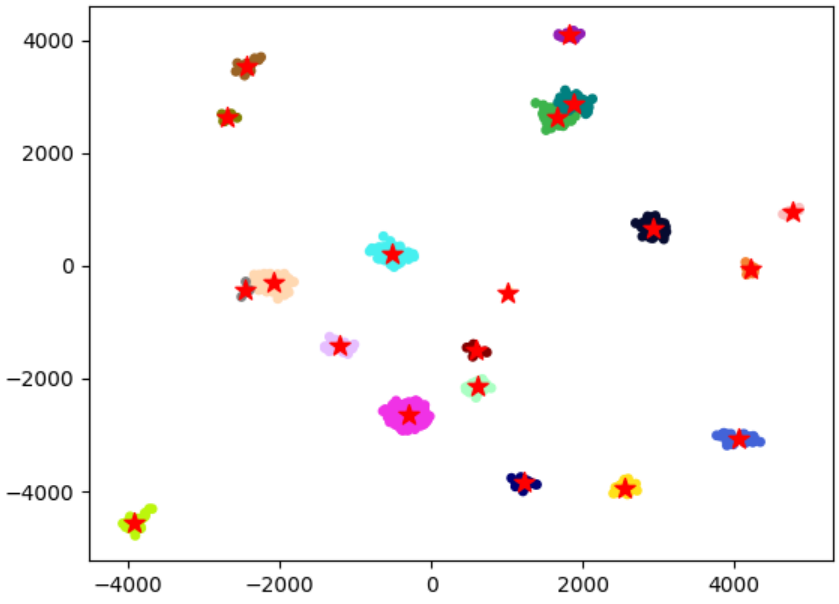


10000 bodov



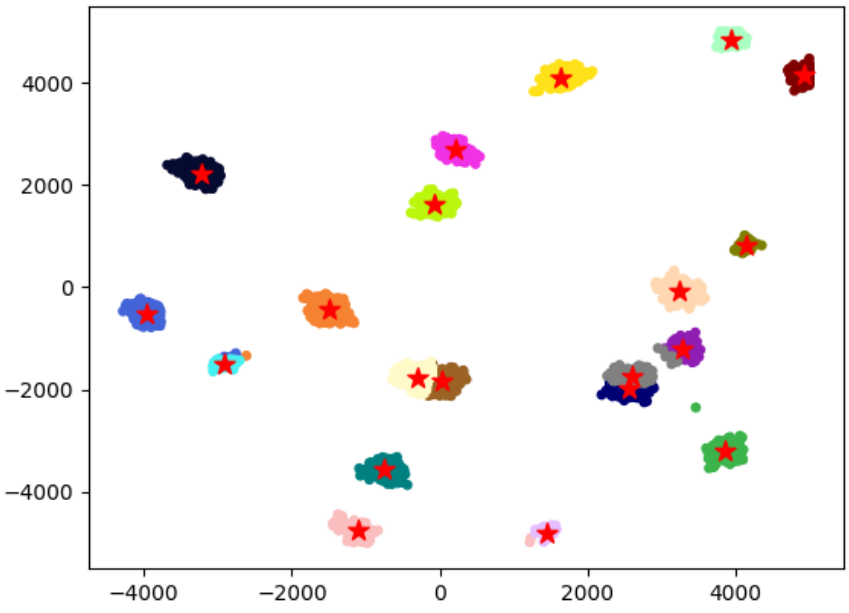
20000 bodov

Aglomeratívne zhlukovanie

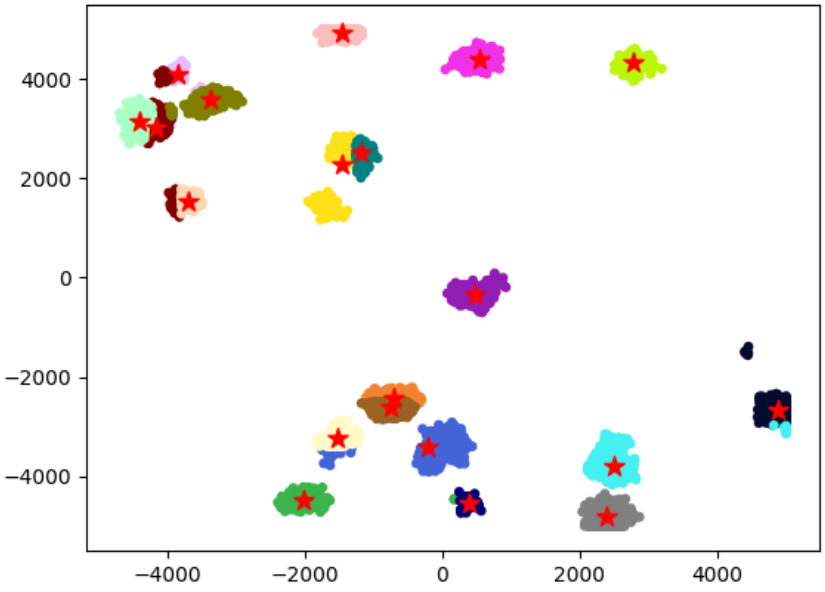


1000 bodov

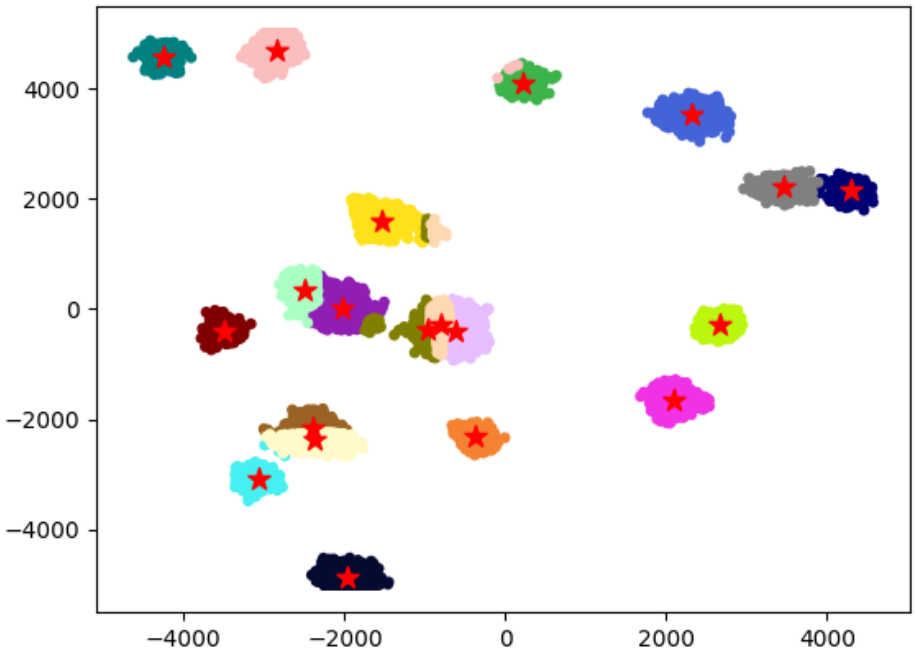
Divízne zhlukovanie



5000 bodov



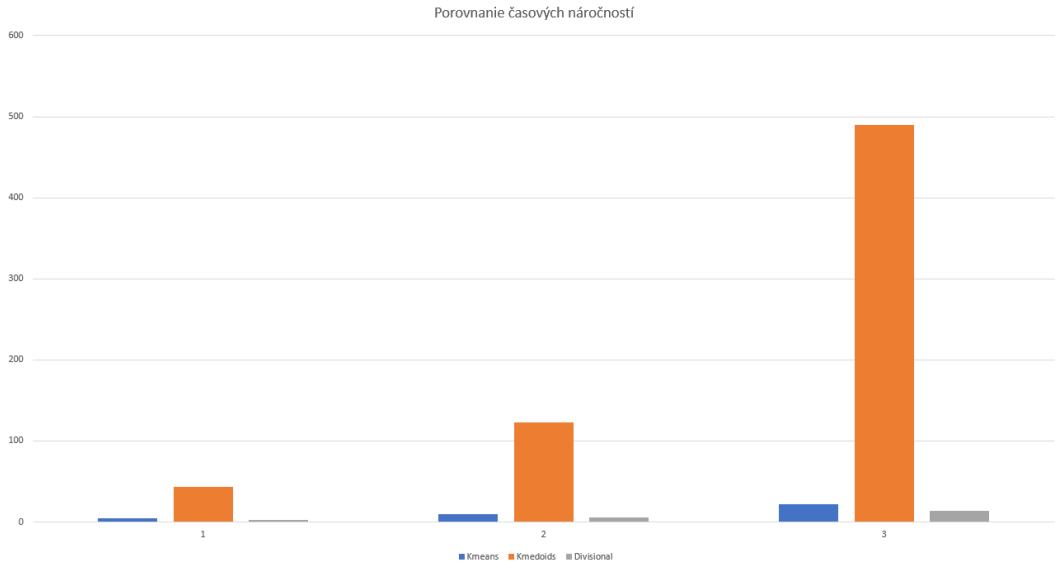
10000 bodov

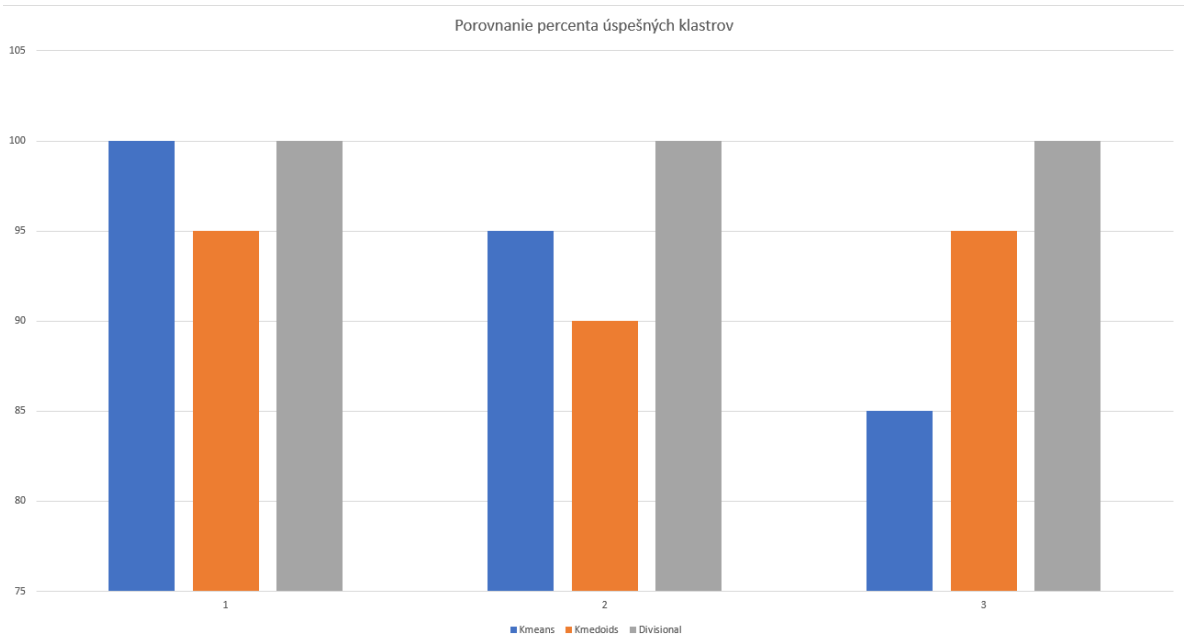


20000 bodov

Zhodnotenie

Z implementovaných typov zhlukovačov je najefektívnejší zhlukovač divízivnou metódou pomocou k-means. Výpočet samotného centroidu je veľmi rýchly. K-means je tiež rýchly. Implementácia medoidu je časovo najviac náročná posle aglomeracií.





Metóda aglomerácie bola najdlhšia a trvala mi 579 sekúnd pre 1000 bodov. Ale úspešnosť bola 100%.