ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE FAKULTA STAVEBNÍ KATEDRA GEOMATIKY					
Název předmětu Geoinformatika					
Úloha U4	Název úlohy: Neřízená klasifikace				
akademický rok 2024/2025	semestr zimní	studijní skupina C102	vypracoval Matyáš Pokorný Tereza Černohousová	datum 13.12.2024	klasifikace

Technická zpráva

1 Zadání

Vygenerujte alespoň 3 množiny bodů/klastry (např. funkce randn) ve 2D. Vytvořte vlastní implementaci algoritmu shlukování k-means. Porovnejte svůj výsledek s výsledkem shlukování nad totožnou množinou s využitím funkce kmeans. Případné rozdíly komentujte.

2 Bonusové úlohy

Rozšiřte své řešení shlukování k-means do n-dimensionálního prostoru

3 Pracovní postup

V této technické zprávě je popsán pracovní postup implementace algoritmu k-means pro shlukování dat ve 2D a 3D prostoru. Nejprve byla generována data pro shlukování, přičemž byly vytvořeny tři shluky bodů, které byly náhodně transformovány pomocí různých metod, jako je škálování a posun. Tato data byla následně načtena a připravena pro algoritmus k-means, přičemž byly rovnoměrně rozmístěny počáteční středy shluků na základě minimálních a maximálních hodnot souřadnic. Počáteční středy byly realizovány na přímce. Poté byl implementován vlastní algoritmus k-means, který přiřadil body k nejbližším středům, následně vypočítal nové středy shluků na základě průměrů bodů přiřazených ke každému shluku, a tento proces iteroval, dokud nedošlo k dostatečné konvergenci. Podrobnější pracovní postup je krok za krokem popsaný pomocí pseudokódu

Vygenerované shluky byly clusterizovány pomocí vestavěnné funkce `kmeans` a výsledky byly porovnány.

4 Pseudokód

4.1 2D Clusterizace

Matlab kód pro `gen_points_randn2D.m`

1. **Inicializace parametrů.**
   (a) Nastavení počtu bodů $N = 50$.
   (b) Nastavení parametru transformace $a = 1.5$.
   (c) Generování náhodného posunutí pro shluky jako vektor b s náhodnými hodnotami mezi 0 a 10.
2. **Generování bodů.**
   (a) Vytvoření matice A obsahující N bodů, které jsou vygenerovány s normálním rozdělením (pro každý bod se generují dvě náhodné hodnoty s normálním rozdělením).

(b) Transformace bodů matice A pomocí parametru a a přidání posunutí b, což dává novou matici bodů B.

(c) Další transformace bodů matice A pomocí vektoru b a přičtení záporné hodnoty pro druhou souřadnici, což dává matici bodů C.

3. **Vizualizace generovaných dat.**

   (a) Zobrazení bodů z matice A na grafu (rozptýlené body, označené jako černé).

   (b) Zobrazení bodů z matice B na stejném grafu (označené jinou barvou, např. modré).

   (c) Zobrazení bodů z matice C na stejném grafu (označené jinou barvou, např. červené).

   (d) Použití funkce `grid on` pro zobrazení mřížky na grafu.

4. **Sjednocení bodů do jedné matice.**

   (a) Sjednocení všech tří matic A, B a C do jedné matice M.

5. **Uložení dat.**

   (a) Uložení matice M obsahující všechny body do souboru s názvem `data.mat`.

Matlab kód pro `k_means2D.m`

1. **Načtení dat.**

   (a) Načtení dat z uloženého souboru.

   (b) Převod dat do matice pro získání souřadnic x a y.

2. **Definice parametrů.**

   (a) Zadání počtu shluků n uživatelem.

   (b) Stanovení počátečního umístění středů shluků pomocí minimálních a maximálních hodnot souřadnic.

3. **Výpočet diagonální přímky.**

   (a) Určení minimálního bodu $[min(x), min(y)]$.

   (b) Určení maximálního bodu $[max(x), max(y)]$.

4. **Umístění počátečních středů.**

   (a) Vytvoření počátečních středů rovnoměrným rozdělením mezi minimálními a maximálními hodnotami souřadnic pro x a y.

5. **Zobrazení počátečních středů.**

   (a) Zobrazení datových bodů.

(b) Zobrazení diagonální přímky.

(c) Zobrazení počátečních středů jako zelené kříže na grafu.

6. **Clusterizace - Přiřazení bodů ke shlukům.**

   (a) Inicializace tolerance pro změnu středů.

   (b) Inicializace proměnné pro změnu středů.

   (c) Inicializace maximálního počtu iterací.

   (d) Inicializace počtu iterací na 0.

7. **Iterace pro přiřazení bodů k shlukům a aktualizaci středů.**

   (a) **While cyklus:**

      i. Pokračování, dokud změna středů není menší než tolerance a není dosažen maximální počet iterací:

         A. Vytvoření vektoru L, který bude obsahovat přiřazení bodů k shlukům.

         B. Pro každý bod:

         C. Vypočítat vzdálenosti bodu od všech středů shluků.

         D. Přiřadit bod k shluku s nejbližším středem.

         E. Po přiřazení bodů k shlukům, přepočítat nové středy:

         F. Pro každý shluk vypočítat nový střed jako průměr bodů přiřazených ke shluku.

         G. Pokud je shluk prázdný, zachovat původní střed.

         H. Vypočítat změnu mezi starými a novými středy (distanční změnu).

         I. Pokud je změna středů větší než tolerance, pokračovat v iteraci.

8. **Použití funkce `kmeans` pro kontrolu výsledků**

   (a) Použití vestavěné funkce `kmeans` pro clusterizaci.

9. **Vizualizace výsledků.**

   (a) Zobrazení výsledků clusterizace:

   (b) Zobrazení výsledků pomocí vestavěné funkce `kmeans`:

4.2 3D Clusterizace

Matlab kód pro `gen_points_randn3D.m`

1. **Inicializace parametrů.**

   (a) Nastavení počtu bodů $N = 100$ v každém shluku.

(b) Nastavení parametru pro škálování druhého shluku $a = 1.25$.

(c) Generování náhodného vektoru b pro posunutí třetího shluku, kde b obsahuje tři náhodné hodnoty mezi 0 a 10.

2. **Generování dat pro shluky.**

(a) Vytvoření matice A obsahující $N = 100$ bodů generovaných s normálním rozdělením v 3D prostoru.

(b) Generování druhého shluku B, kde každý bod v matici A je vynásoben koeficientem $a = 1.25$ a poté je přičten náhodný vektor b.

(c) Generování třetího shluku C, kde body v matici A jsou posunuty o vektor b v záporném směru pro souřadnici x a posunuty v ose y a z.

3. **Vizualizace generovaných shluků.**

4. **Spojení dat do jedné matice.**

(a) Sjednocení všech bodů z matic A, B a C do jedné matice M.

5. **Uložení dat do souboru.**

(a) Uložení matice M obsahující všechny body do souboru s názvem `data3D.mat`.

Matlab kód pro `k_means3D.m`

1. **Načtení dat.**

(a) Načtou se data z uloženého souboru `data3D.mat`, který obsahuje shluky bodů v 3D prostoru.

(b) Data se extrahují a uloží do proměnných x, y a z, které obsahují jednotlivé souřadnice bodů.

2. **Zadání počtu shluků.**

(a) Uživatel zadá požadovaný počet shluků n.

3. **Výpočet diagonální přímky.**

(a) Vypočítají se minimální a maximální hodnoty souřadnic pro x, y a z, což definuje rozsah dat.

4. **Rozmístění počátečních středů.**

(a) Počáteční středy shluků jsou rovnoměrně rozmístěny mezi minimálními a maximálními hodnotami souřadnic pro každou osu (x, y, z).

(b) Počáteční středy jsou uloženy do matice S.

5. **Vizualizace počátečních středů.**

(a) Vykreslí se 3D scatter plot pro všechny body, kde jsou body zobrazeny podle jejich souřadnic (x, y, z).

(b) Na grafu je vykreslena diagonální přímka a počáteční středy shluků jsou zobrazeny jako červené hvězdičky.

6. **Algoritmus k-means.**

(a) Inicializují se parametry pro algoritmus k-means:

- Tolerance $podminka = 0.01$ pro zastavení algoritmu.
- Maximální počet iterací $maxIter = 100$.

(b) Pro každou iteraci se provádí následující kroky:

- Každý bod je přiřazen k nejbližšímu středu (na základě Euklidovské vzdálenosti).
- Pro každý shluk se přepočítá nový střed jako průměr bodů přiřazených k tomuto shluku.
- Spočítá se vzdálenost posunu středů a pokud je posun menší než stanovená tolerance, algoritmus se zastaví.

7. **Porovnání s k-means funkcí.**

(a) Výsledek algoritmu je porovnán s výsledky funkce `kmeans` z MATLABu.

(b) Funkce `kmeans` je použita pro stejné množství shluků a vrací indexy přiřazení bodů a nové středy.

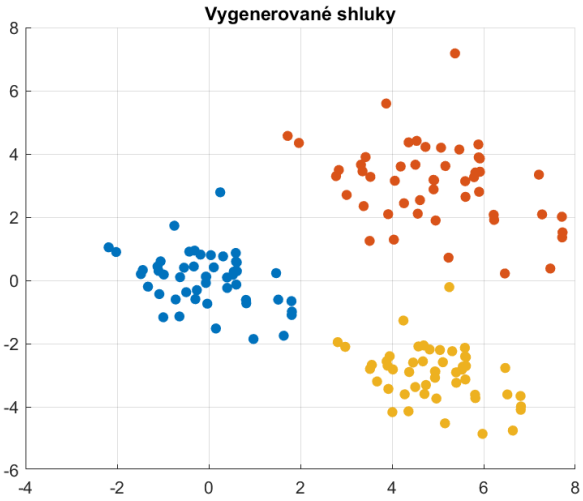
8. **Vizualizace výsledků.**

(a) Na prvním grafu jsou vykresleny body podle přiřazení k shlukům z vlastního algoritmu k-means a střední hodnoty shluků jsou zobrazeny červenými hvězdičkami.

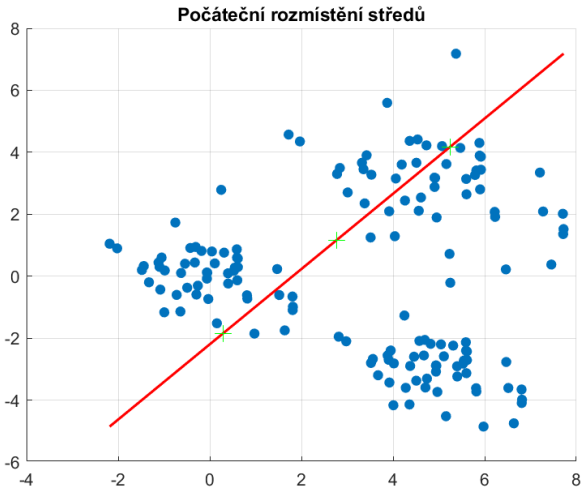
(b) Na druhém grafu jsou vykresleny výsledky funkce `kmeans`, kde jsou přiřazeny různé barvy pro jednotlivé shluky a středové body jsou opět zobrazeny jako červené hvězdičky.

5 Výsledky

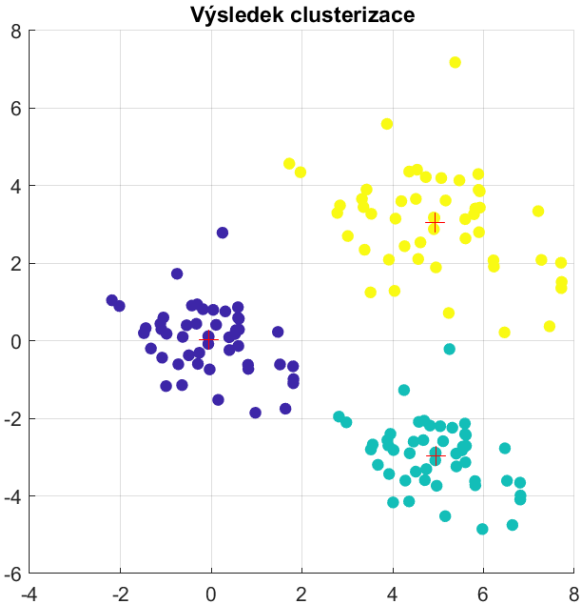
5.1 2D



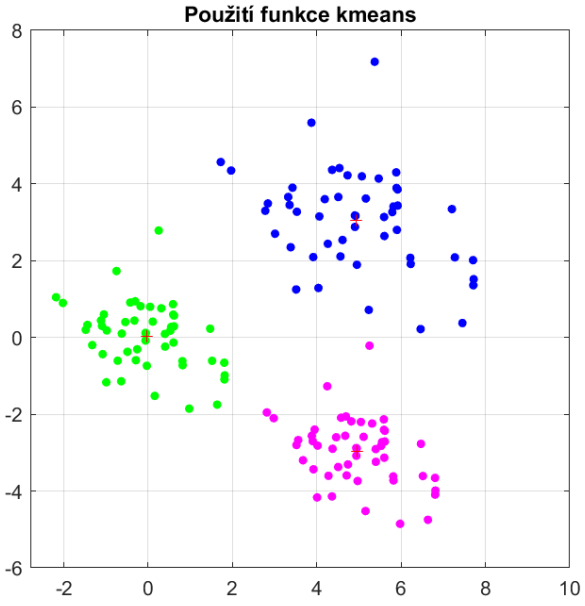
Obrázek 1: Vygenerované 2D shluky



Obrázek 2: Počáteční rozmístění středů

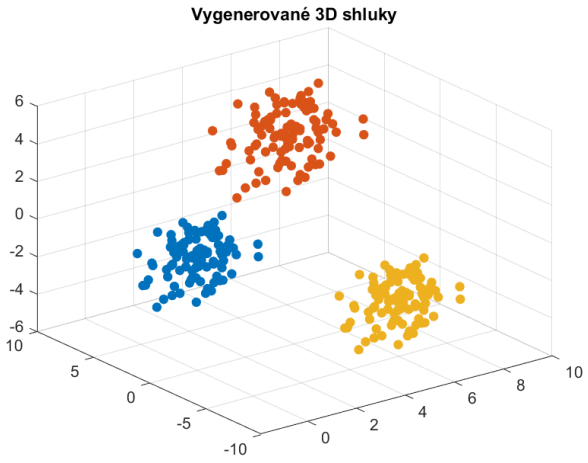


Obrázek 3: Vlastní funkce na clusterizace

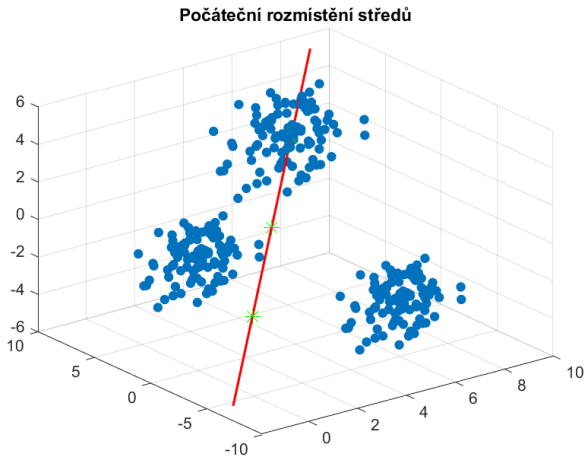


Obrázek 4: Použití funkce kmeans

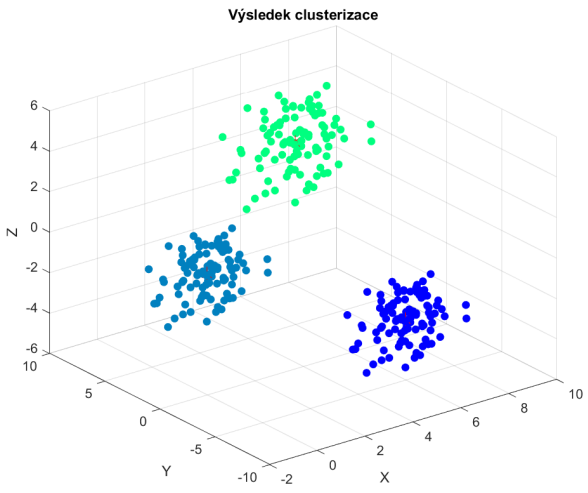
5.2 3D



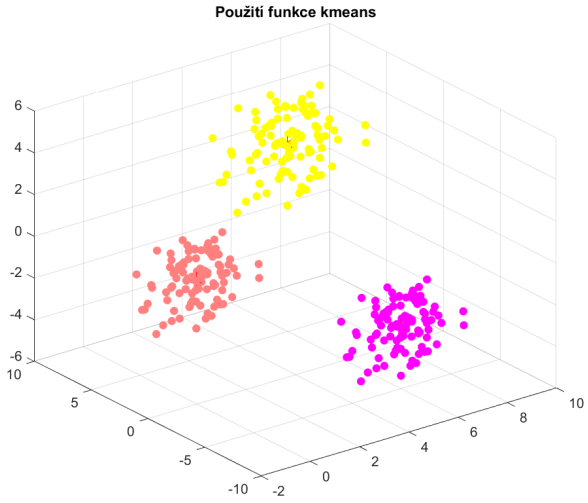
Obrázek 5: Vygenerované 3D shluky



Obrázek 6: Počáteční rozmístění středů



Obrázek 7: Vlastní funkce na clusterizace



Obrázek 8: Použití funkce kmeans

6 Závěr

Pro ověření správnosti algoritmu byly výsledky porovnány s funkcí kmeans v MATLABu, která slouží jako standardní nástroj pro shlukování. Nakonec byly výsledky obou přístupů vizualizovány pomocí 2D a 3D grafů, kde byly body barevně přiřazeny k jednotlivým shlukům a středy těchto shluků byly zobrazeny jako výrazné značky. Porovnání ukázalo, že implementovaný algoritmus dává podobné výsledky jako vestavěná funkce MATLABu. Avšak záleží na vygenerovaných shlucích. Funkce se shodují, pokud na počátku jsou shluka na první pohled od sebe dobře rozeznatelné. V případě, že se shluky více mísí, jsou výsledky odlišné.

V Praze dne: 13.12. 2024

T. Černohousová
M. Pokorný