Project 2 (Part 1): K-means clustering algorithm
Total marks: 15
Due date: Check Canvas
Submission: No file submission. **You will** demonstrate your solution on your own computer **during class.**

Goal

Implement K-means from scratch in Python and explore a method to choose the number of clusters. You may use common libraries (e.g., `numpy`, `matplotlib`, `math`) for arrays, plotting, and math, but **do not** use any off-the-shelf K-means implementation.

Environment

- Work **locally on your laptop**.
- You can use Jupyter Notebook

Data

Use the provided **`data.txt`** (2-D points so you can visualize clusters). You will find this file on Canvas along with this file.

Part A — Implement `kmeans_fall2025()` (10 pts)

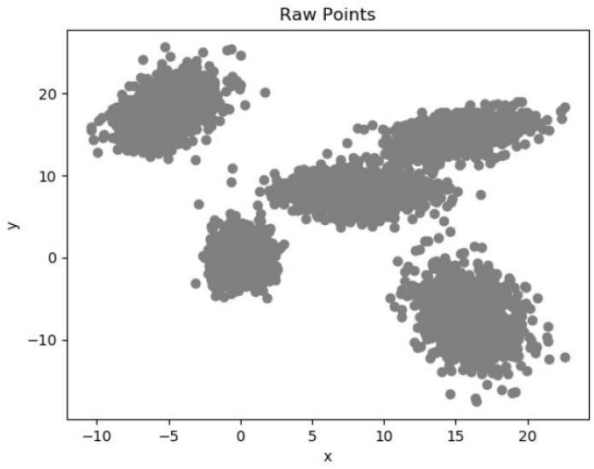
Write a function **`kmeans_fall2025(data, k, ...)`** that performs K-means on the dataset.

Include these steps with clear, commented code:

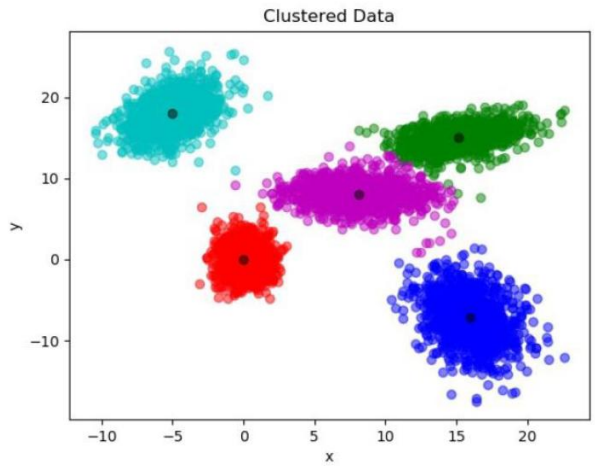
1. **Randomly initialize** `k` cluster centers.
2. **Assign** each point to its **nearest** center (Euclidean distance).
3. **Update** centers as the means of assigned points.
4. **Iterate** steps 2–3 until your **stopping criteria** are met (e.g., centroid displacement threshold **and/or** max iterations). Check the class lecture if necessary.

Visualization:

- Plot the data **before** clustering and **after** clustering.
- Use distinct colors for different clusters.
- Your plots will look like the image below.



Plot before clustering



Plot after clustering

Part B — Choosing *k* with the Elbow Method (5 pts)

Implement the **Elbow method** (or another method of your choice) **without** using off-the-shelf clustering selection utilities.

1. Run **`kmeans_fall2025()`** in a loop over candidate values of **`k`**.
2. For each **`k`**, compute an appropriate **distortion/SSE** (within-cluster sum of squares).
3. **Plot** your metric vs. **`k`** and **identify** the "elbow" (your chosen optimal **`k`**).
4. **Show** the clustered data using the selected **`k`**.

Be prepared to explain **how** you determined the elbow.

In-Class Demo (what to show)

You will **walk through and run** the following on your laptop:

- Your **code** for `kmeans_fall2025()` and the **Elbow** routine.
- **Plots**:

    - Data before clustering
    - Data after clustering (for your chosen `k`)
    - **Elbow curve** (metric vs. `k`), with your selected `k` clearly indicated
- A brief explanation of your **stopping criteria**, **distance metric**, and **why** your `k` is reasonable.

No files need to be uploaded; evaluation is based on your live demonstration and explanation.

Grading Criteria

- **Correctness & completeness** (algorithm steps, no off-the-shelf K-means).
- **Code quality & organization** (clear structure, comments, readability).
- **Visualizations** (before/after plots; clear elbow curve).
- **Live performance** (runs without errors on your machine; reproducible results).
- **Understanding** (able to answer questions about choices, stopping criteria, and the selected `k`).