LAB 4: Clustering

Assignment 1

(a) Generate 4 Gaussian clusters in two dimensional space given by two numeric variables in range [-10, +10]. The Gaussian distributions of the clusters must have different means and the same standard deviation, initially set to 0.6. The total number of instances for all the four clusters is 300.
Hint: use function `make_blobs` from `sklearn.datasets.samples_generator`.

(b) Run *k*-means clustering algorithm on the data obtained in (a) and visualize the clusters and their centers for *k* in {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}.
Hint: use function `KMeans` from `sklearn.cluster`. Set parameter `random_state` to `None`.
Print the contingency tables of the clustering solutions.
Hint: use function `contingency_matrix` from **`sklearn.metrics.cluster`.**

(c) Plot the sum of square errors (SSE) for *k* in {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}. Does the plot indicate that the natural number of clusters is 4?

(d) Repeat (a)-(c) when you generate the clusters with the standard deviation of 0.1 and 2.5. Do the SSE plots indicate that the natural number of clusters is 4?

(e) Repeat (d) with another cluster-center initialization by setting the parameter `random_state` of `KMeans` to an integer number.

- Do you receive clustering solutions similar to those obtained for `random_state`=`None`? Can you explain why?
- Can you propose an extension of the k-means initialization that is less dependent on `random_state`; i.e. it results in similar clustering solutions?

Assignment 2

(a) Load and print the vertebrate.csv data.

(b) Run single-link, max-link and average-link hierarchical clustering on this data. Visualize the hierarchies. Choose the hierarchy that in your view is most natural given the data and explain why.
Hint: To run hierarchical clustering use function `hierarchy.linkage` from `scipy.cluster`. To visualize `hierarchy.dendrogram` from `scipy.cluster`.

Assignment 3

(a) Load and visualize the chameleon.csv data.

(b) Run the DBSCAN method on this data for eps=15.5 and min_samples=5. Vizualize the clustering solutions.
Hint: To run the DBSCAN method hierarchical clustering use function `DBSCAN` from `sklearn.cluster`.

(c) Experiment with the DBSCAN method for eps in [1, 21] with step 5 and min_samples in [1, 21] with step 5. Comment on the clusters for different settings.

Submission Requirements
Please follow the guidelines below for submitting your solution:

- Submit a PDF file that serves as your Analytical Report. This should contain all your written findings, interpretations, and graphical visualizations for each assignment. Ensure that the graphics are clearly labeled and appropriately integrated into your explanations.
- Additionally, submit a PDF version of your Jupyter Notebook that contains all the code used for data generation, cluster training/validation, analysis, and visualization. Make sure that the code is well-commented for readability.
- Self-Evaluation PDF: Submit another PDF containing your self-evaluation of the Analytical Report and the Jupyter Notebook code. End this file with a summary of what you have learned from completing the assignments. The rubrics for self-evaluation are given in Appendices A and B (see below).

Note: Submitting all three files is essential for a complete submission. Failure to submit any of them will result in a deduction of points.

Academic Integrity Declaration

By submitting the Analytical Report, you are declaring that you have not used large language models or any other automated tools to generate written answers and interpretations on a semantic level and/or a language level. The work you submit must be your own.
Failure to adhere to this academic integrity guideline will be considered a violation and may result in a grade penalty or other disciplinary actions.

Please include this declaration at the end of your Analytical Report PDF.

"*I, [Your Name], hereby declare that I have not used large language models or any automated*
tools for generating the written answers and interpretations in this Analytical Report. "

Appendix A: Grading Rubrics for Analytical Rubrics (0-100 points)

Assignment 1:

- Data generation and use of k-means 10 points
- Clusters' visualization 10 points
- Plots of the sum of square errors 10 points
- Answers, interpretation, and conclusion 20 points

Assignment 2:

- Use of hierarchical clustering and visualization 12.5 points
- Motivation of the chosen hierarchical clustering 12.5 points

Assignment 2:

- Use of DBSCAN and visualization 12.5 points
- Explaining cluster quality in function of settings 12.5 points

Appendix B: Code Evaluation Rubrics (0-100 points)

- Code organization: 10 points
- Proper commenting and documentation: 30 points
- Proper use of Python libraries and functions: 20 points
- Correctness of the implemented logic: 20 points
- Efficiency of code: 20 points