Project 2 — Unsupervised Learning Models in Industry (10%)

Models: K-Means | Agglomerative Clustering | Gaussian Mixture Models

Due Date: Saturday, December 6th @ 11:59 PM (EST)

Objective

Apply Unsupervised Learning techniques to identify natural clusters or hidden patterns in a real-world dataset.

You will perform preprocessing, visualize feature relationships, experiment with multiple configurations, and compare model performance to determine which one works best for your chosen dataset.

Each student in the group must select one model (either K-Means, Agglomerative Clustering, or GMM) to work on and demonstrate individually in their video submission. As a group, you must ensure that all three models are represented and documented in your final submission.

Submission Requirements and

Evaluation Weight Distribution

Submission Requirements & Evaluation Weight Distribution

1. **Total Weight:** 100 marks

Component	Weight	Description
1. Individual Video Demonstration	**50%**	Each member records a short video explaining their contribution, visualizations, and findings.

Submission Requirements & Evaluation Weight Distribution

2. Group Document Submission **2.**	**50%**	Combined group deliverables listed below.

Files to Submit (Required Parts)

1. **Jupyter Notebook — full implementation Steps 1–15 with all plots inline.**
   Naming Format: Group#_Unsupervised_PythonCode.ipynb
2. **Metric Table — Excel or CSV summarizing all models and evaluation metrics.**
   Naming Format: Group#_Unsupervised_Project_Metrics.xlsx
3. **Short Report (2–4 pages) — include problem statement, EDA summary, key insights, best model selection, and business interpretation.**
   Naming Format: Group#_Unsupervised_Project.docx
4. **Dataset Used — Excel or CSV file of the dataset (limit ≤ 5,000 rows).**

Individual Video Demo

Each team member must submit their own video.
Length per person: **2–3 minutes**.

Your video must include:

1. **Camera ON** – introduce yourself & your group number.
2. Show the **part of your model code and visualization** you worked on.
3. Explain your **code and what worked and why it was best or worst**
4. Speak clearly. Use proper screen sharing.
   (Phone recordings will receive penalties.)
5. If you cannot upload your video due to Blackboard file size limits, upload it to **OneDrive, Google Drive, or YouTube**, set the link to **Allow View Access**, and then **submit the share link in Blackboard**. **Do not send your video by email**.

Project Workflow

Frame the Problem & Train/Test Split

- Define your **unsupervised objective** (e.g., customer segments, usage patterns).
- Identify **features** (no explicit target).
- Create **three splits**: train_size = {0.10, 0.25, 0.30}.
- Keep the test set unseen until final evaluation.

EDA (Exploratory Data Analysis)

- head(), info(), shape, missing counts.
- Histograms & boxplots for distribution.
- Correlation heatmap to spot redundant features.

Visual Insights

- **At least 2 plots** revealing structure (e.g., scatter matrix of top features, correlation map).
- Later, recolor plots by cluster labels for interpretation.

Categorical Handling

- Detect categorical columns.
- Apply **One-Hot Encoding** or **Ordinal Encoding** (briefly justify choice).

Custom Feature Engineering

- Add ≥ 1 new feature (e.g., ratio, grouped mean).
- Implement as FunctionTransformer or custom BaseEstimator.

Feature Scaling

- Use **StandardScaler** (or MinMaxScaler if justified).

- Show before/after boxplots for scaled features.

Pipelines

Create a **ColumnTransformer + Pipeline** including:

- Encoder (for categoricals)
- Scaler (for numerics)
- Placeholder for model
  Use fit() on train only and transform() on test.

Model Training & Evaluation

Run each model with

- n_clusters = {3, 5, 7, 9}
- train_size = {0.10, 0.25, 0.30}
- appropriate iteration parameters

For each combination, record metrics on train and test.

Model 1 – K-Means Clustering

- Grid:
  - n_clusters ∈ {3, 5, 7, 9}
  - n_init ∈ {10, 20, 50}
  - max_iter ∈ {100, 300, 600}
- Visuals:
  - 2D Cluster Scatter (colored by cluster)

Model 2 – Agglomerative (Hierarchical)

- Grid:
  - n_clusters ∈ {3, 5, 7, 9}

- Visuals:
    - Cluster distribution bar chart

Model 3 – Gaussian Mixture Model (GMM)

- Grid:
    - n_components ∈ {3, 5, 7, 9}
    - max_iter ∈ {100, 300, 600}
- Visuals:
    - 2D Scatter with probability ellipses
    - Cluster membership heat map

Metric Aggregation → Excel/CSV

Create a tidy table:

Model Train_Size n_clusters Params Silhouette DB CH Notes

Export to Excel/CSV for reporting.

Pick the Best Model

- Choose model with highest metrics score.
- Plot **bar chart** comparing models on Score.
- Show final **cluster scatter** and **profile table** (mean per feature per cluster).
- Briefly discuss outliers and pattern interpretation.

Final Test Evaluation + Save

- Refit best model on chosen train split.
- Evaluate on test set.
- Save model

- Document:
    - Model choice & insights
    - Limitations (e.g., scaling sensitivity)
    - Monitoring plan (e.g., retrain quarterly)