UL task: SpectralClustering

Motivation: Group Olivetti face images into identity-consistent clusters **without labels**. The dataset is high-dimensional and lies on a curved manifold (pose, lighting, expression). Clustering exposes this intrinsic structure directly from features.

Datasets (design choice & rationale):

- **Olivetti Faces (real-world, images):** 400 images (64×64 → 4096-D). High-D, small-n; benefits from **PCA** and is a realistic testbed for face embeddings.

  Model fit:

  - **HDBSCAN:** after PCA (≈100–200), can find identity pockets and mark outliers (`min_cluster_size` tunes strictness).
  - **OPTICS:** density ordering shows structure; reasonable on PCA features.
  - **Spectral (ours):** with `affinity='nearest_neighbors'` often recovers ≈40 subjects on PCA features; captures **non-linear**separations.

  (We initially considered the generated Two-Moons dataset to illustrate non-convex structure, but we ultimately report only Olivetti for a purely real-world evaluation.)

Metrics:

1. **Silhouette Coefficient** *(unsupervised)* — average "closer to own cluster than the next best" score using distances in feature space; [−1,1][−1,1], higher is better.

   Why here: lets us tune Spectral's `n_neighbors` and `n_clusters` **without labels** on the PCA space. *(Caveat: sensitive to cluster-size imbalance and large kk; we report overall values.)*

2. **Adjusted Rand Index (ARI)** *(external)* — chance-adjusted agreement with ground-truth identities; [−1,1][−1,1]with 0≈0≈ random and 1=1= perfect.

   Why here: Olivetti ships labels; ARI quantifies identity consistency post-hoc and enables a fair comparison to density methods. *(Caveat: ARI tends to prefer matching the true number of clusters; we sweep kk around ≈40 and report both ARI and Silhouette.)*

CI.3

Model: Spectral Clustering

Motivation: Spectral converts clustering into **graph partitioning**. It builds a **k-nearest-neighbors (kNN)** similarity graph and clusters in the space of the **graph Laplacian's** leading eigenvectors, which captures **non-linear** structure (pose/lighting/expression) better than centroid-based models.

Key characteristics:

1. Handles **non-linear** cluster geometry via a neighborhood graph.
2. Less biased toward spherical clusters than k-means/GMM; tolerates shape/size imbalance.
3. Requires `n_clusters`—useful here because prior knowledge is ≈40 subjects.

CI.4

How Spectral Clustering works (method, optimization, implementation)

Step 1 – Affinity graph: Build sparse **kNN graph** WW (edges between nearest neighbors).

Step 2 – Normalized Laplacian: Lsym=I−D−1/2WD−1/2Lsym=I−D−1/2WD−1/2, with Dii=∑jWijDii=∑jWij.

Step 3 – Spectral embedding: Take the kk eigenvectors of LsymLsym with **smallest** eigenvalues; row-normalize to get U∈Rn×kU∈Rn×k.

Step 4 – Discretization: Run **k-means** on rows of UU to obtain kk clusters.

Optimization view: Spectral is the relaxed solution of **Normalized Cuts**

Ncut(A1,…,Ak)=∑r=1kcut(Ar,A¯r)assoc(Ar,V).

Ncut(A1,…,Ak)=r=1∑kassoc(Ar,V)cut(Ar,A¯r).

The exact minimization is NP-hard; the **spectral relaxation** reduces it to an eigenproblem on the Laplacian, then k-means discretizes the relaxed solution.

Our implementation (scikit-learn):

- Preprocessing: **PCA(100, whiten=True)** on flattened images (denoise + stable distances).
- Model: `SpectralClustering(affinity="nearest_neighbors", assign_labels="kmeans")`.

- Swept hyper-params: `n_clusters ∈ {30,35,38,40,42,45,50}` , `n_neighbors ∈ {8,10,12,15,20}` , `random_state=42` .

CI.5

Results for Spectral Clustering (Olivetti Faces)

Input: n=400n=400, 40 ground-truth subjects; features used: **PCA(100)** → X∈R400×100X∈R400×100

===

Parameters: `n_clusters=38` , `n_neighbors=10` **Evaluation metrics:silhouette_score: 0.053adjusted_rand_score: 0.272** *(best ARI in our sweep)(see figure: spectral_olivetti_pca100_k38_nn10.png)*

Parameters: `n_clusters=40` , `n_neighbors=12`

Evaluation metrics:

- **silhouette_score: 0.047**
- **adjusted_rand_score: 0.199**

(see figure: spectral_olivetti_pca100_k40_nn12.png)

Discussion & findings:

- **Best setting by ARI:** *(k=38, nn=10)* slightly **under-specifies** the nominal 40 identities, avoiding over-splitting and yielding the best identity consistency (**ARI ≈ 0.272**).
- **Silhouette magnitudes: low (~0.05)** across settings ⇒ clusters are **partly overlapping** in PCA(100) with Euclidean distance; identity separation exists but is weak in this space.
- **Effect of `n_neighbors`:** Too small fragments the graph; too large over-smooths and merges identities. The sweet spot is **10–12 neighbors**.
- **Full assignment:** Spectral labels **every** sample (no `1` noise), which simplifies Silhouette reporting and contrasts with density methods that may leave hard cases as noise.

CI.6

What we achieved.

Using **Spectral Clustering** with a **k-NN affinity** on **PCA(100, whiten=True)** features of the Olivetti faces, we obtained **moderate identity consistency**: the best setting in our sweep (e.g., `n_clusters=38`, `n_neighbors=10`) delivered **ARI ≈ 0.27** with **Silhouette ≈ 0.05**. This indicates that Spectral can recover a meaningful portion of the underlying identity structure without supervision, but separability remains limited in this feature space. We also observed a consistent trade-off: slightly **under-specifying** the number of clusters relative to the 40 subjects avoided over-fragmentation and improved ARI, while larger `k` marginally increased Silhouette but split identities.

Interpretation.

Low Silhouette across settings suggests **overlapping clusters** under Euclidean distances in PCA(100)—i.e., the geometry is still "blurry" for identity. Nonetheless, Spectral's non-linear partitioning (via the graph Laplacian eigenvectors) captures local structure better than centroid baselines would on the same features and gives a clean, full assignment of all samples (useful for internal metrics).

Limitations

1. **Requires `n_clusters`.** Spectral needs `k` ahead of time. While we justified it via the dataset prior (~40 subjects) and a sweep, the choice still influences outcomes.
2. **Sensitivity to `n_neighbors`.** Too small fragments the graph; too large over-smooths and merges identities. Performance varies with this single knob.
3. **Feature space is linear PCA.** PCA preserves variance, not identity separability. Important, low-variance identity cues may be discarded; Euclidean distances in PCA space may not reflect semantic similarity.
4. **Full assignment (no noise class).** Spectral cannot label "hard" images as noise the way density methods can, which can depress Silhouette when clusters overlap.
5. **Scale & generalization.** Results are on a small dataset (n=400). Out-of-sample assignment isn't native (requires a transform strategy like Nyström); scalability depends on sparse eigensolvers and kNN graph construction.

Possible Future Improvements

Better representations

- **More expressive features:** Replace PCA with **learned embeddings** (e.g., a pre-trained face CNN or a small autoencoder/contrastive encoder). Then run Spectral on those embeddings.
- **Non-linear DR before Spectral:** Try **UMAP (n_neighbors≈15, n_components≈30)** → Spectral; UMAP often improves neighborhood faithfulness for faces.
- **PCA tuning:** Re-run with **150–200 PCs** and **no whitening** as an ablation; on some runs this preserves more identity signal for graph construction.

Graph & objective variants

- **Affinity choices:** Compare `nearest_neighbors` vs **RBF/heat kernel** affinity; tune σσ via local scaling (self-tuning spectral clustering) to adapt to density variation.
- **Laplacian variants:** Try **random-walk Laplacian** and check stability; inspect the **eigengap** to guide kk.

Model selection & robustness

- **Selection protocol:** Choose (k, nneighbors)
(k, nneighbors) via **Silhouette** (primary) and report ARI post-hoc; include **per-cluster Silhouette** and **cluster size balance** to detect over-fragmentation.
- **Stability checks:** Repeat with multiple seeds for the k-means discretization; report variance of metrics.
- **Baselines:** Add k-means/GMM on the same features as reference points; include Davies–Bouldin or Calinski–Harabasz for completeness.

Scaling & deployment

- **Approximate kNN & Nyström:** For larger n, use **ANN graphs** (e.g., FAISS) and **Nyström out-of-sample**extensions to map new images to clusters.
- **Noise handling hybrid:** If “unclusterable” images are problematic, consider a **hybrid**: Spectral to get coarse partitions, then **HDBSCAN within clusters** to allow noise labeling.

Reporting polish

- Include a small **hyperparameter grid table** (top 5 rows by ARI and by Silhouette), the **best-config scatter** (2D PCA colored by spectral labels), and, optionally, an **eigengap plot** to justify the kk region.