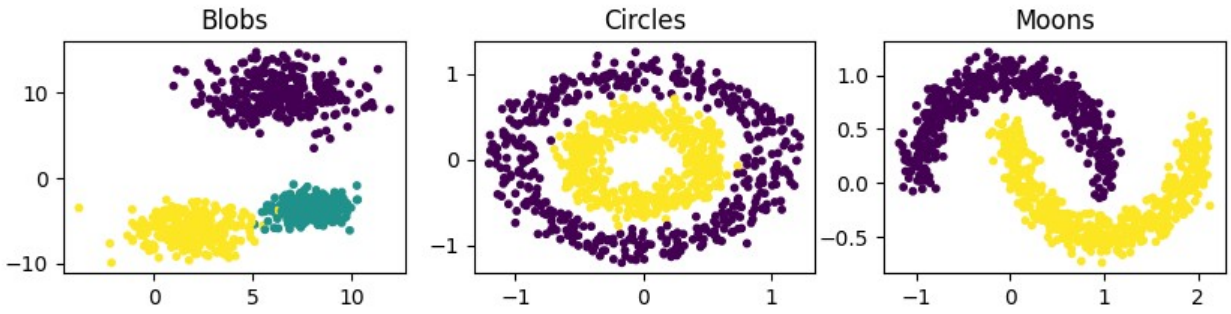
Clustering Methods in Machine Learning

Lab team: J04

Members: Ibón de Mingo y Adam Maltoni

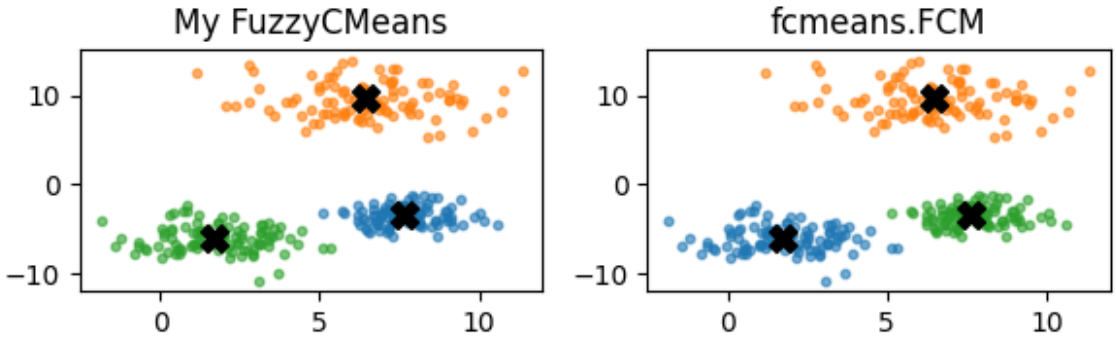
1. Generating Synthetic Data

```
from simulations.ex1 import run
run()
```



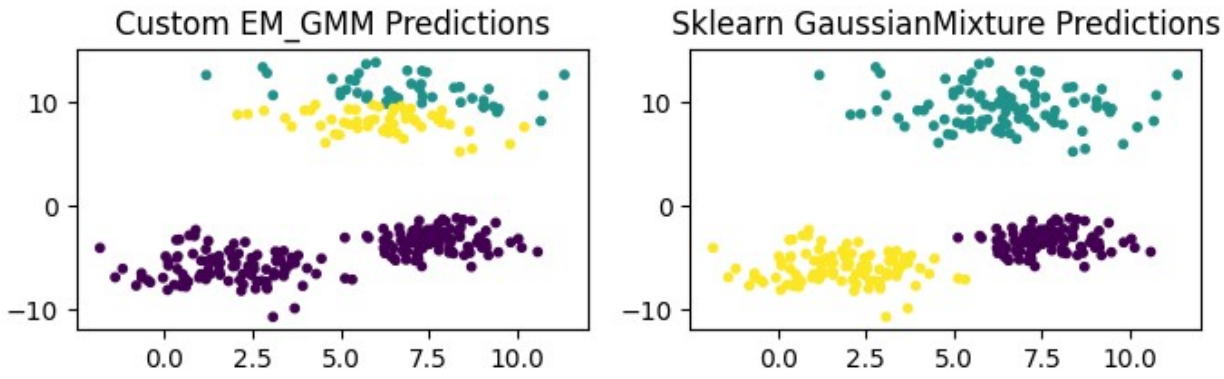
2. Implementing Fuzzy C-Means Clustering

```
from simulations.ex2 import run
run()
```



3. Implementing Expectation-Maximization (EM) for Gaussian Mixture Models (GMM)

```
from simulations.ex3 import run
run()
```

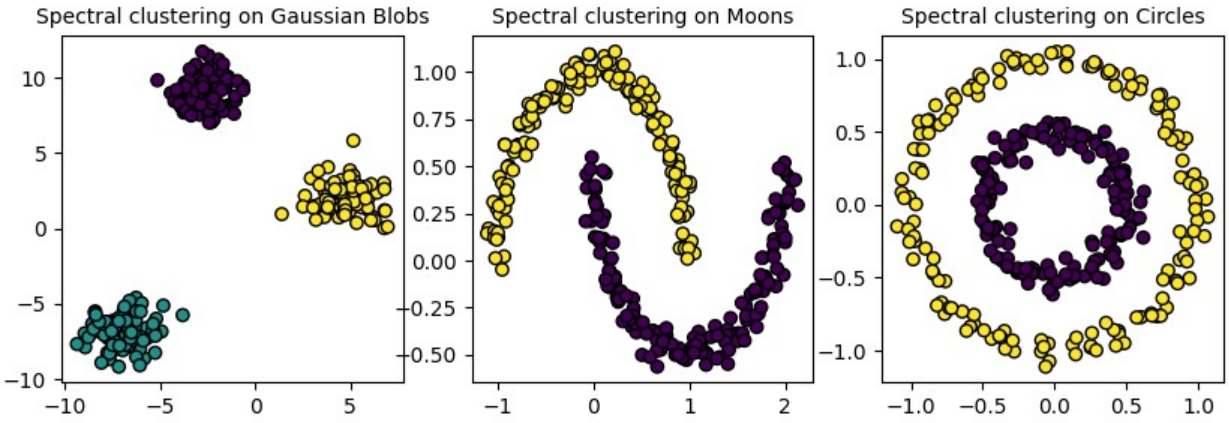


4. Spectral Clustering

Spectral clustering is a technique that uses eigenvalues and vectors of matrices derived from data to perform dimensionality reduction before clustering in fewer dimensions. It is effective for identifying not linearly separable data in clusters (or with complex structures).

First we have to construct a similarity graph. There we represent data as a graph where nodes are data points and edges reflect similarity, often using an RBF kernel or nearest neighbors similarity metrics. After we calculate the degree matrix and derive the Laplacian matrix by subtracting the adjacency matrix, capturing the graph's structure. After that we compute eigenvalues and eigenvectors of the Laplacian, selecting those with the smallest non-zero eigenvalues for dimensionality reduction. Finally, we apply a standard clustering algorithm to the transformed data for final clustering.

```
from simulations.ex4 import run
run()
```



5. Assessing the quality of the clustering

1. Silhouette Coefficient

Measures clustering quality by comparing intra-cluster cohesion and inter-cluster separation, ranging from –1 to 1. **Close to +1** indicates well-clustered points, **0** suggests boundary points,

and **negative** for values that are not well clasiffied. The **Advantages** are its intuitive interpretation and suitability for unsupervised learning and the **disadvantages** are high computational cost and struggles with irregular or varying-density clusters.

---

2. Dunn Index

Assesses clustering quality by comparing the worst-case inter-cluster separation with the worst intra-cluster spread. A **high Dunn Index** indicates well-separated, compact clusters. **Advantages** are its ability to detect poorly defined clusters and ensure even the closest clusters are well-separated. **Disadvantages** include sensitivity to outliers and computational cost for large datasets.

---

3. Adjusted Rand Index (ARI)

Evaluates clustering accuracy by comparing data point pairs to true labels, adjusting for randomness. Scores range from **1 (perfect match), 0 (random grouping), to negative (poor agreement). Advantages** include its correction for chance and direct alignment with labels. **Disadvantages** are its reliance on true labels, limiting unsupervised applications, and the assumption that labels reflect true clusters.

6. Determining the optimal number of clusters

1. Elbow Method

Plots WCSS (inertia) against the number of clusters to find the "elbow" point, balancing **low intra-cluster variance** and avoiding excessive clusters. The **advantages** are its simplicity and effectiveness for well-separated clusters. The **disadvantages** are that the elbow point can be subjective and may not work well for non-spherical clusters.

---

2. Silhouette Score

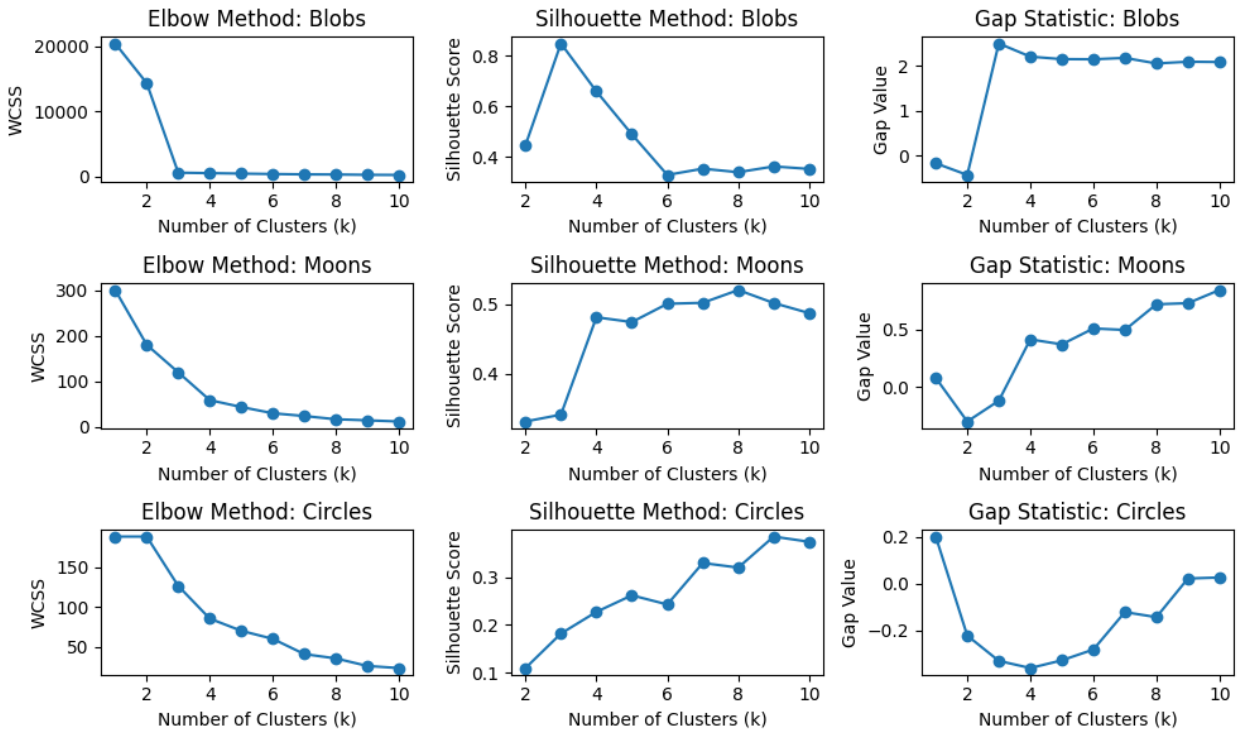
Evaluates clustering quality by comparing intra-cluster cohesion and inter-cluster separation. The **advantages** are that it works well for non-globular clusters and considers both cohesion and separation. The **disadvantages** are its high computational cost and potential misleading results in high-dimensional data.

---

3. Gap Statistic

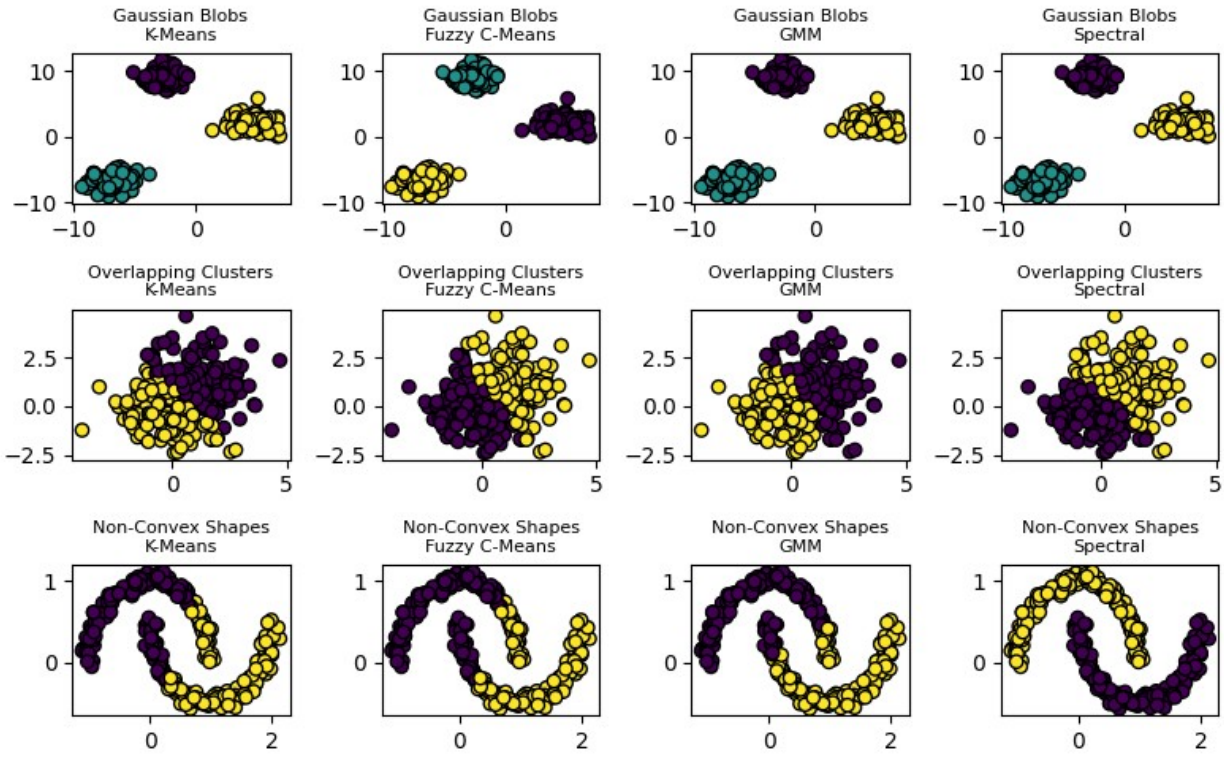
Compares clustering performance on real data vs. random reference data, selecting the k with the largest gap. The **advantages** are its statistical rigor and ability to succeed where other methods fail. The **disadvantages** are its computational intensity and sensitivity to reference distribution choice.

```
from simulations.ex6 import run
run()
```



6.1 Synthetic Datasets

```
from simulations.ex6_1 import run
run()
```



Gaussian Blobs (Well-Separated Clusters)

All methods perform and predict perfectly (ARI = 1.0, high silhouette and Dunn Index, classification error = 0)

Overlapping clusters

Low Silhouette & Dunn Index confirm weak separation between clusters. Spectral Clustering achieves the highest ARI (0.209) and lowest classification error (0.27), suggesting it best captures the structure. K-Means, Fuzzy C-Means, and GMM perform similarly, with slightly lower ARI and classification errors around 0.28–0.29.

Non convex shapes

Spectral Clustering achieves perfect ARI (1.0) and no classification error, indicating it captures the structure perfectly (as it was said before, since it is a non linearly separable shape). GMM performs well (ARI = 0.498, Classification Error = 0.147), showing that it adapts better than K-Means and Fuzzy C-Means (ARI = 0.247, Classification Error = 0.25), which have limitations with non-convex shapes.

6.2 German Credit Dataset

```
import pandas as pd
from sklearn.preprocessing import StandardScaler

data = pd.read_csv('./data/german_credit_data_numeric.csv', sep=';')
```

```
X = StandardScaler().fit_transform(data.drop(columns=['Class']))
y = data['Class']
```

Analysis

1. Structure & initial findings

- The **Hopkins statistic is 0.62**, suggesting only *weak* cluster tendency. Ideally, values > 0.75 indicate strong clustering potential (measures whether a dataset is randomly distributed or contains meaningful clusters)
- **PCA does not separate classes**, meaning linear transformations fail to reveal intrinsic structure.
- **Pairwise distances follow a normal distribution**, suggesting that points are evenly spread rather than forming distinct groups (compute straight-line distances between all data points, forming a symmetric distance matrix)
- The **class distribution is imbalanced** (70% class 0, 30% class 1).
- The **correlation heatmap is cold**, indicating weak relationships between features.
- The **feature distributions by class show little separation**, suggesting limited individual feature importance for clustering.

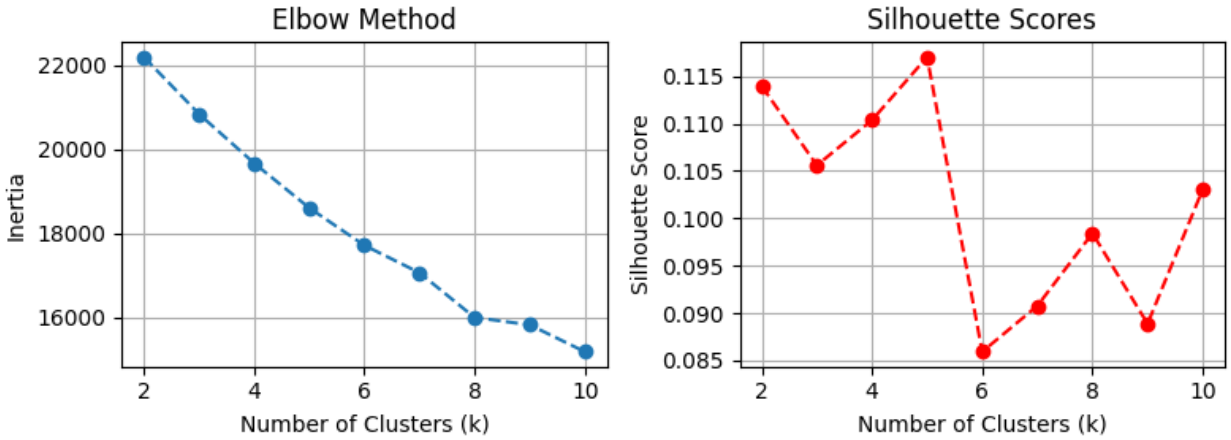
2. Clustering justification

Since the dataset does not show natural clustering tendency at all, traditional clustering methods (like k-means) may not perform well. However, we still proceed because we want to validate that clustering does not work well (negative results are important insights), to test alternative clustering techniques, such as spectral clustering or probabilistic models (GMM).

3. Clustering - classes relationship

Clustering is unsupervised, so we risk biasing the result if we determine our approach based on classes distribution. However, knowing the class distribution after clustering helps evaluate performance.

```
from simulations.ex6_2 import run_elbow_silhouette
run_elbow_silhouette()
```



GMM

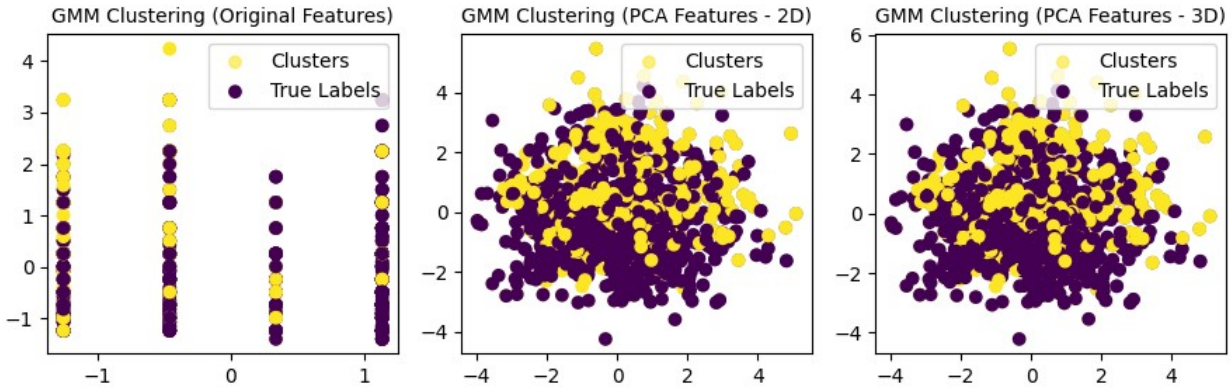
```
from simulations.ex6_2 import run_clustering
# run_clustering(X, y, method="gmm", n_clusters=5,
optimize_params=True)
run_clustering(X, y, method="gmm", n_clusters=2, optimize_params=True)


Running GMM clustering with           2 clusters...

GMM Clustering (Original) - ARI: 0.0197, NMI: 0.0035, Accuracy:
0.3980, F1-score: 0.4017
Contingency Matrix (pctg):
[[65.50632911 34.49367089]
 [72.07602339 27.92397661]]

GMM Clustering (PCA - 2D) - ARI: 0.0713, NMI: 0.0287, Accuracy:
0.3520, F1-score: 0.3609
Contingency Matrix (pctg):
[[57.69230769 42.30769231]
 [76.28398792 23.71601208]]

GMM Clustering (PCA - 3D) - ARI: 0.0511, NMI: 0.0189, Accuracy:
0.3720, F1-score: 0.3830
Contingency Matrix (pctg):
[[60.28571429 39.71428571]
 [75.23076923 24.76923077]]
```



Interpretation of GMM Results Adjusted Rand Index (ARI) = 0.01, meaning the clusters do not align with true labels at all. This confirms that there is no meaningful separation in the data using probabilistic clustering.

Clustering Challenges in This Dataset

Despite the dataset being separable (as seen in Practica 0 with accuracies up to 80%), clustering methods perform poorly. Feature distributions suggest meaningful class differences, yet clustering fails due to:

1. **Feature Overlap** – Many class distributions overlap, preventing clear decision boundaries. Clustering assumes homogeneous groups, but if classes share feature space, separation is weak.

2. **Distance-Based Limitations** – K-Means, GMM, and Spectral Clustering rely on distance metrics. When clusters lack compactness or clear separation, these methods fail.

3. **Lack of Natural Clustering** – PCA, pairwise distances, and Hopkins statistics indicate the data does not form distinct clusters. If labels are based on complex decision rules rather than inherent similarity, clustering will not succeed.

We tested **K-Means, Fuzzy C-Means, Spectral Clustering, and Gaussian Mixture**, all yielding low ARI scores (~0–0.03). Alternative approaches may be needed.

Although all the models with the different variants show poor performance, the **GMM PCA 2D version** achieves the highest ARI (0.0713) and NMI (0.0287), indicating a slight improvement in alignment with the true classes. However, accuracy and F1-score are low, meaning that the cluster structure does not reflect the actual labels.

Clusters do not correctly capture the true classes. In the best case (PCA 2D), 57.7% of the first cluster comes from class 0 and 42.3% from class 1, indicating significant mixing. Class 0, which makes up 70% of the dataset, is split between both clusters without a dominant assignment. The second cluster is not representative of a single class, containing 76.3% of class 0 and only 23.7% of class 1.

Finally, we tested different cluster counts, but matching the number of clusters to the number of classes did not improve the results. This happens because the data's representation does not naturally group into structures that align with the supervised class labels.

7. Feature Importance in Clustering

1. Centroid analysis

Examines cluster centroids to identify influential features by comparing mean or median values across clusters. The **advantages** of this method are that it is simple and intuitive, leveraging clustering output and that ther is no extra modeling needed for K-Means or Fuzzy C-Means. While the **disadvantages** are that it requires feature scaling. and it is limited to linear differences, missing complex patterns. Additionally, extra steps are needed if centroids aren't directly available.

---

2. Dimensionality reduction (PCA loadings)

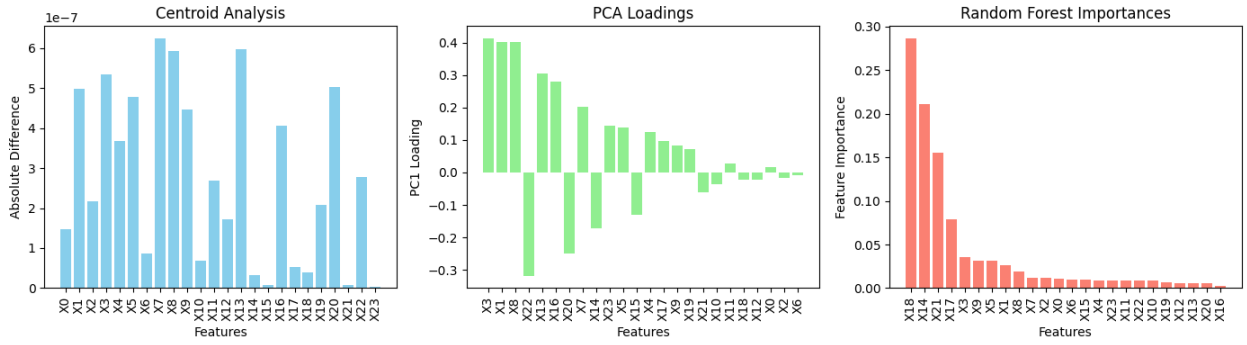
Applies PCA to identify key features based on their loadings in principal components where clusters separate. Some **Advantages:** are that it captures variance, highlighting key directions in the data and that it handles correlated features by combining them into components. PCA is unsupervised and it focusses on variance rather than clustering structure which may be a **disadvantage**. It is also Hard to interpret since components mix multiple features.

---

3. Supervised modeling on cluster labels

Trains a classifier to predict cluster labels, using feature importance metrics to identify key variables. The **advantages** of this method are that it provides clear rankings for feature importance and captures non-linear relationships and interactions. The **disadvantages** are that it assumes cluster labels are meaningful and stable, results vary with model choice and tuning, and it adds complexity by requiring an additional modeling step.

```
from simulations.ex7 import run
run()
```



8. Interpreting Clustering Results

Clusters in relation to original attributes

Clustering was applied to a numeric version of the **German Credit dataset**, but the original dataset includes key categorical features (**employment status, credit history, savings, housing**).

Understanding cluster alignment with these attributes is essential for real-world decision-making. One key challenge is separation, as histograms reveal variable differences, but clusters do not align well with creditworthiness labels. Additionally, the correlation matrix indicates that most numerical features lack strong linear relationships, making clustering difficult. Another issue is class imbalance, with 70% of the data belonging to one class, which hinders the formation of well-defined clusters. Dimensionality reduction through PCA further suggests weak class separation, highlighting the complexity of multi-feature interactions. Mapping clusters to categorical attributes, such as job type, housing, and credit history, could improve interpretability.

Findings on the German Credit Problem using clustering reinforce these challenges. The Adjusted Rand Index (ARI) remains low, around 0.02, indicating that clustering does not naturally recover credit risk labels. The Hopkins statistic, approximately 0.61, suggests no strong clustering tendency, further supporting this observation. Among the clustering methods, Spectral Clustering and Gaussian Mixture Models (GMM) outperform K-Means, revealing some underlying structure, though still misaligned with the actual labels. Given these limitations, alternative approaches such as supervised learning or hybrid models may be more effective in capturing meaningful patterns in the data.

9. Conclusions

Clustering Performance and Insights

Clustering groups similar data points to uncover hidden patterns. We tested K-Means, GMM, Spectral Clustering, and FCM on original and PCA-transformed data. K-Means was efficient but struggled with complex shapes. FCM allowed partial memberships for ambiguous cases. GMM handled overlapping data well with probabilistic assignments. Spectral Clustering captured complex structures but was computationally expensive.

Method Trade-offs

GMM provides a probabilistic view, while K-Means is simpler. K-Means is fast but rigid, whereas Spectral Clustering and FCM offer flexibility at a higher cost. PCA can enhance clustering by reducing noise but may remove critical variations.

Applications and Conclusion

Clustering is widely used in marketing, fraud detection, medicine, and data organization. No method is universally best; selecting the right one depends on data characteristics, objectives, and constraints. Combining metrics, visualization, and domain knowledge ensures meaningful insights.

10. References

Hastie, T., Tibshirani, R., & Friedman, J. (2009). Spectral clustering. In *The elements of statistical learning: Data mining, inference, and prediction* (2nd ed., ch. 14). Springer.

Murphy, K. P. (2021). Chapter 21: Clustering. In *Probabilistic Machine Learning: An Introduction* (pp. 709-734). The MIT Press.

IEEE. (2011). *Fuzzy*. IEEE. Retrieved from http://ieeexplore.ieee.org/document/5941851

Analytics Vidhya. (2024, May). *Understanding fuzzy c-means clustering*. Retrieved from https://www.analyticsvidhya.com/blog/2024/05/understanding-fuzzy-c-means-clustering/

Pivei, D. (2019). *Gaussian Mixture Models (GMM) in R*. RPubs. Retrieved from https://rpubs.com/dapivei/705612

Pawar, T. (2021, May 19). *Gaussian mixture models explained: Applying GMM and EM for effective data clustering*. Medium. Retrieved from https://medium.com/@tejaspawar21/gaussian-mixture-models-explained-applying-gmm-and-em-for-effective-data-clustering-ca24f8911609

Codex. (2021, March 28). *EM algorithm and Gaussian mixture model (GMM)*. Medium. Retrieved from https://medium.com/codex/em-algorithm-and-gaussian-mixture-model-gmm-6ea5e0cf9d6e

Python Course. (n.d.). *Expectation-maximization and Gaussian mixture models (GMM)*. Retrieved from https://python-course.eu/machine-learning/expectation-maximization-and-gaussian-mixture-models-gmm.php

GeeksforGeeks. (n.d.). *Spectral clustering*. Retrieved from https://www.geeksforgeeks.org/ml-spectral-clustering/

Wikipedia. (n.d.). *Spectral clustering*. Retrieved from https://en.wikipedia.org/wiki/Spectral_clustering

scikit-learn. (n.d.). *SpectralClustering*. Retrieved from https://scikit-learn.org/stable/modules/generated/sklearn.cluster.SpectralClustering.html

GeeksforGeeks. (n.d.). *Measuring clustering quality in data mining*. Retrieved from https://www.geeksforgeeks.org/measuring-clustering-quality-in-data-mining/

Analytics Vidhya. (2020, October 5). *Quick guide to evaluation metrics for supervised and unsupervised machine learning*. Retrieved from https://www.analyticsvidhya.com/blog/2020/10/quick-guide-to-evaluation-metrics-for-supervised-and-unsupervised-machine-learning/

Datanovia. (n.d.). *Determining the optimal number of clusters: 3 must-know methods*. Retrieved from https://www.datanovia.com/en/lessons/determining-the-optimal-number-of-clusters-3-must-know-methods/

Towards Data Science. (2021, May 19). *Cheat sheet to implementing 7 methods for selecting optimal number of clusters in Python*. Retrieved from https://towardsdatascience.com/cheat-

sheet-to-implementing-7-methods-for-selecting-optimal-number-of-clusters-in-python-898241e1d6ad/

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, É. (2021). *scikit-learn: Machine learning in Python*. Journal of Machine Learning Research, 12, 2825–2830. Retrieved from https://scikit-learn.org/stable/user_guide.html

Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N. J., Kern, R., & Picus, M. (2020). *Array programming with NumPy*. Nature, 585(7825), 357–362. Retrieved from https://numpy.org/doc/stable/

Wikipedia contributors. (n.d.). *Hopkins statistic*. Wikipedia, The Free Encyclopedia. Retrieved from https://en.wikipedia.org/wiki/Hopkins_statistic

Wikipedia contributors. (n.d.). *Rand index*. Wikipedia, The Free Encyclopedia. Retrieved from https://en.wikipedia.org/wiki/Rand_index

Wikipedia contributors. (n.d.). *Multivariate normal distribution*. Wikipedia, The Free Encyclopedia. Retrieved from https://en.wikipedia.org/wiki/Multivariate_normal_distribution