ijamrsd.com
ISSN: 2583-6404
Journal of Advanced Multidisciplinary Research Studies and Development
Peer Reviewed Referred Journal | Bi-monthly | Open Access

UNSUPERVISED LEARNING APPROACHES FOR CLUSTERING: AN IN-DEPTH ANALYSIS OF K-MEANS AND HIERARCHICAL METHODS

Dr. Deepa Pankaj Nyayadhish[1], Mrs. Archana Sunil Jadhav[2]
[1] Asst. Professor, Department of Computer Science, V. K. Krishna Menon College, University of Mumbai, India.
[2] Asst. Professor, Department of Information Technology & Data Science, Bunts Sangha Mumbai's Anna Leela College of Commerce & Economics and Shobha Jayaram Shetty College for BMS, University of Mumbai, India.

Abstract
Unsupervised learning is essential for discovering hidden patterns in unlabeled data, with clustering serving as one of its most widely used analytical techniques. This study presents an in-depth evaluation of K-Means and Hierarchical Clustering, two foundational algorithms commonly applied across scientific, industrial, and data-driven domains. The research examines their algorithmic behavior, performance characteristics, and suitability for different data types. K-Means, a centroid-based method, is assessed for its computational efficiency, sensitivity to initialization, and effectiveness in partitioning large datasets. Hierarchical clustering, explored in both agglomerative and divisive forms, is evaluated for its ability to reveal nested cluster structures and provide intuitive visual representations through dendrograms. Using benchmark datasets, the study compares both algorithms based on clustering quality metric silhouette score. Results indicate that K-Means excels in scalability and speed, whereas hierarchical clustering offers greater flexibility and interpretability for smaller or structurally complex datasets. The analysis highlights the strengths and limitations of each method, offering practical guidance on selecting appropriate clustering techniques for various unsupervised learning tasks and real-world applications.

Keywords: Unsupervised Learning, Clustering Techniques, K-Means Algorithm, Hierarchical Clustering, Agglomerative Clustering, Silhouette Coefficient.

► *Corresponding Author: Dr. Deepa Pankaj Nyayadhish*

1. Introduction

Unsupervised learning is an important area of machine learning focused on identifying hidden patterns and structural relationships in data without relying on labeled outcomes. Clustering is one of the key methods used in unsupervised learning. This paper examines and implements two widely used clustering techniques—K-Means Clustering and Hierarchical Clustering. Given only a dataset, the objective is to discover underlying patterns or structures within it. Unlike supervised learning, it does not rely on predefined labels, which makes it highly useful for applications such as customer segmentation, anomaly detection, and similar tasks. [1].

2. What is Unsupervised Learning?

Unsupervised learning can be compared to navigating an unfamiliar city without a map. Given only a dataset, the objective is to discover underlying patterns or structures within it. Unlike

supervised learning, it does not rely on predefined labels, which makes it highly useful for applications such as customer segmentation, anomaly detection, and similar tasks [1].

2.1 Part A: K-Means Clustering

K-Means is a flexible and commonly applied clustering technique. It partitions a dataset into *K* separate, non-overlapping clusters. Each data point is assigned to the cluster whose mean is closest, with the mean acting as the representative of that cluster. The value of *K* is specified by the user, and determining the most suitable value is often an important area of analysis [1].

We will compare the performance of both algorithms using the 'mall_customer' dataset.

Step 1: Importing all necessary libraries as shown below

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

from sklearn.preprocessing import StandardScaler
from sklearn.cluster import KMeans, AgglomerativeClustering
from sklearn.metrics import silhouette_score

from scipy.cluster.hierarchy import dendrogram, linkage

sns.set(style="whitegrid")
```

Step 2: Load the 'Mall_Customer' dataset

```
Dataset = "/content/Mall_Customers.csv"
df = pd.read_csv(Dataset)
df.head()
```

	CustomerID	Gender	Age	Annual Income (k$)	Spending Score (1-100)
0	1	Male	19	15	39
1	2	Male	21	15	81
2	3	Female	20	16	6
3	4	Female	23	16	77
4	5	Female	31	17	40

Step 3: Select the features & scale them

Here we are using the 2 attributes for segmentation: Annual Income (k$) Spending Score (1–100) and performing feature Scaling.

```
X = df[["Annual Income (k$)", "Spending Score (1-100)"]]
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)
```

Step 4: We will find the optimal number of clusters (k) by using Elbow Method

```
inertia = []
K = range(1, 11)

for k in K:
    km = KMeans(n_clusters=k, random_state=42)
    km.fit(X_scaled)
    inertia.append(km.inertia_)

plt.figure(figsize=(7,4))
plt.plot(K, inertia, marker='o')
plt.title("Elbow Method for K-Means")
plt.xlabel("Number of Clusters (k)")
plt.ylabel("Inertia")
plt.show()
```

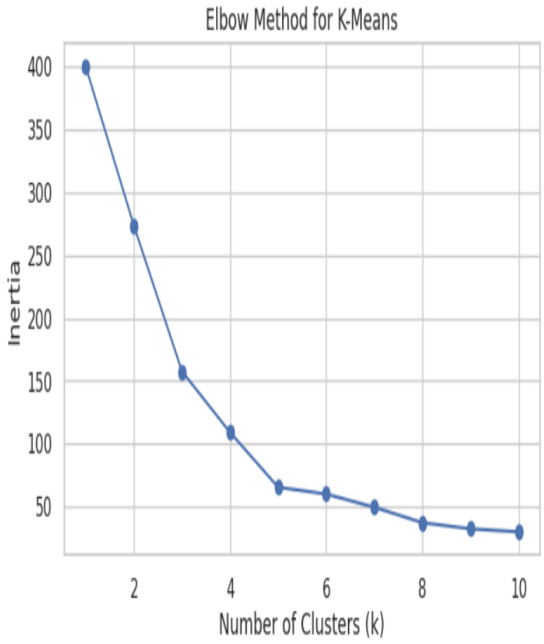

Fig. 1: Elbow Method

Fig. 1 shows that the optimal no of clusters (k) is 5

Step 5: We will fit the K-Means for k=5 as below

```
kmeans = KMeans(n_clusters=5, random_state=42)
labels_kmeans = kmeans.fit_predict(X_scaled)
df["KMeans_Cluster"] = labels_kmeans
```

Step 6 : Now Visualize the created clusters.

```
plt.figure(figsize=(7,5))
plt.scatter(X["Annual Income (k$)"],X["Spending Score (1-100)"],
    c=labels_kmeans,cmap="viridis")
plt.xlabel("Annual Income (k$)")
plt.ylabel("Spending Score (1-100)")
plt.title("K-Means Clustering")
plt.show()
```

Fig.2 visualizes the resultant five clusters using k-Means.

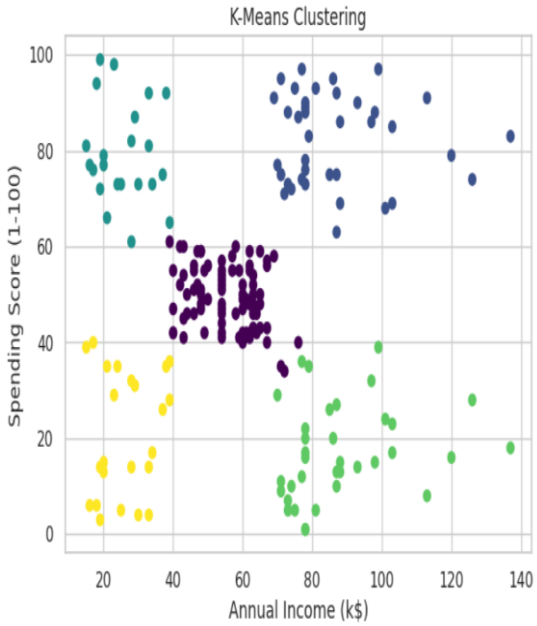

Fig. 2: Resultant 5 clusters

2.2 Part B: Hierarchical Clustering (Agglomerative)

Hierarchical Clustering constructs a tree-structured representation, known as a dendrogram, to illustrate how clusters are formed. The algorithm begins by treating each data point as an individual cluster and progressively combines them until a single cluster is formed. This iterative merging results in a hierarchy of clusters that can be effectively visualized through a dendrogram.

The Python code below uses the 'single' linkage method to merge clusters based on the closest data points. The Fig. 3 shows the resulting Hierarchical Clustering dendrogram which provides insights into the hierarchical structure of the data.

```
#Plot Dendrogram
plt.figure(figsize=(12,5))
linked = linkage(X_scaled, method='ward')
dendrogram(linked)
plt.title("Hierarchical Clustering Dendrogram")
plt.xlabel("Samples")
plt.ylabel("Distance")
plt.show()
```

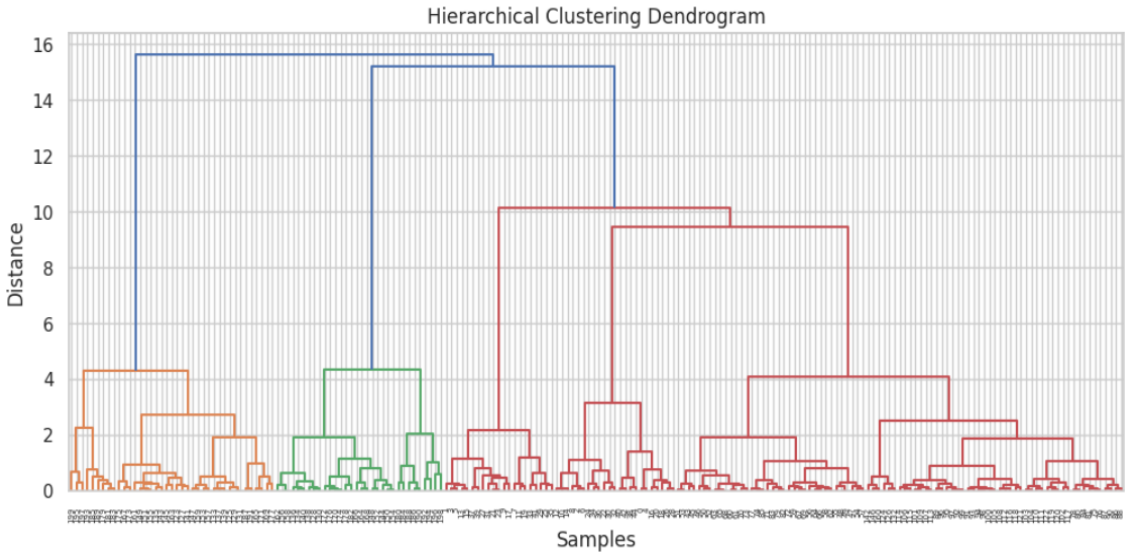

Fig 3: Hierarchical Clustering Dendrogram

Now Fit the Hierarchical Clustering model for optimal k value (i.e 5)

```
hc = AgglomerativeClustering(n_clusters=5,metric='euclidean',linkage='average')
labels_hc = hc.fit_predict(X_scaled)
df["Hierarchical_Cluster"] = labels_hc
```

Visualize the resultant clusters

```
plt.figure(figsize=(7,5))
plt.scatter(X["Annual Income (k$)"],X["Spending Score (1-100)"],
            c=labels_hc,cmap="plasma")
plt.xlabel("Annual Income (k$)")
plt.ylabel("Spending Score (1-100)")
plt.title("Hierarchical Clustering")
plt.show()
```

Fig. 4 shows the resultant clusters by Hierarchical method.

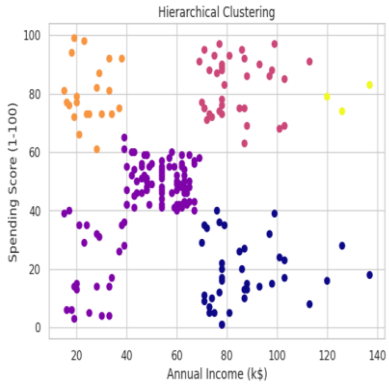

Fig. 4 Hierarchical Clusters

2.3 Part C: Cluster Evaluation

Cluster validation relies on measures of similarity or dissimilarity, such as the distances between data points within and across clusters. A clustering algorithm is considered effective if it groups similar observations together while placing dissimilar ones into separate clusters. Among the most commonly used evaluation metrics for clustering are the Silhouette Coefficient and Dunn's Index. The Silhouette Coefficient is calculated for each data sample and is based on two components: **a)** the average distance between a sample and all other points within the same cluster, and **b)** the average distance between a sample and all points in the nearest neighboring cluster.

The overall Silhouette Coefficient for a dataset is obtained by averaging the individual sample scores. Its value ranges from −1, indicating poor clustering, to +1, representing well-defined and compact clusters. Values close to zero suggest overlapping clusters. Higher scores reflect clusters that are both dense and well separated, aligning with the fundamental principles of effective clustering [3].

$$\text{Silhouette Coefficient } (s) = \frac{b - a}{\max(a, b)}$$

where

a = Average distance between sample and all other points in same cluster
b = Average distance between sample and all other points in next nearest cluster

Equation 1: Silhouette Coefficient [4]

The code snippet below depicts the K-means and Hierarchical cluster evaluation using Silhouette Coefficient**.**

```
sil_km = silhouette_score(X_scaled, labels_kmeans)
sil_hc = silhouette_score(X_scaled, labels_hc)

print("Silhouette Score - KMeans:       ", sil_km)
print("Silhouette Score - Hierarchical: ", sil_hc)
```

```
Silhouette Score - KMeans:       0.5546571631111091
Silhouette Score - Hierarchical: 0.4794263081846086
```

3. Analysis and Interpretation

3.1 K-Means Clustering Results

Typical Cluster Patterns

K-Means often produces clusters such as:

1. High Income – High Spending (luxury customers)
2. Low Income – Low Spending (budget customers)
3. High Income – Low Spending (practical / saver customers)
4. Mid Income – Balanced Spending
5. Low Income – High Spending (uncommon but sometimes observed)

3.2 Hierarchical Clustering Results

Observations

1. Clear separation between extreme spenders and low spenders
2. Captures nested subgroups effectively
3. Dendrogram provides strong visual interpretability

Fig. 5a and 5b indicate Silhouette Plot for K-Means & Hierarchical clustering respectively.

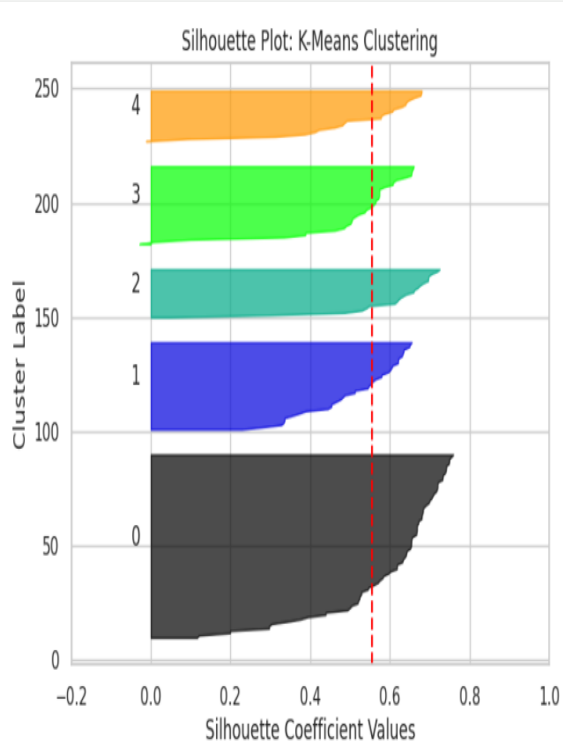

Fig. 5a Silhouette Score K-Means

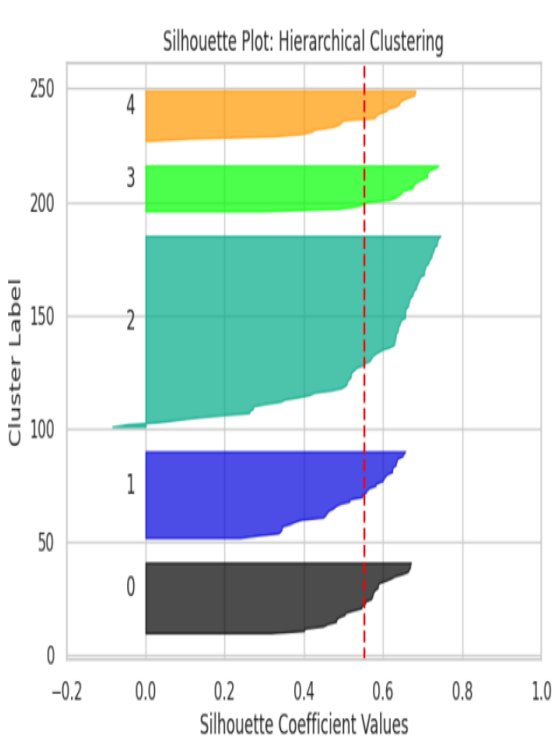

Fig. 5b Silhouette Score Hierarchical

4. Conclusion

Unsupervised learning techniques play a crucial role in extracting meaningful patterns from unlabeled data. This paper presented a comparative analysis of two widely used clustering algorithms, namely K-Means and Hierarchical Clustering, using a real-world customer segmentation dataset. The study demonstrated that K-Means clustering is computationally efficient and performs well for ‘mall-customer’ dataset with relatively spherical cluster structures,

making it suitable for large-scale practical applications. In contrast, Hierarchical Clustering provides greater interpretability by revealing the inherent hierarchical structure of the data through dendrogram visualization, which is particularly beneficial for exploratory data analysis.

The experimental results, evaluated using the Silhouette Coefficient, highlight the strengths and limitations of both approaches. While K-Means exhibits superior scalability and execution speed, Hierarchical Clustering offers enhanced flexibility and insight into cluster relationships for smaller or structurally complex datasets. These findings emphasize that the selection of an appropriate clustering technique depends on dataset characteristics and application requirements.

Future work may extend this study by incorporating additional datasets, exploring alternative linkage methods, and employing multiple evaluation metrics such as the Dunn Index to further enhance the robustness of clustering performance analysis.

5. References

1. https://medium.com/@ilyurek/unsupervised-learning-understanding-clustering-with-k-means-and-hierarchical-clustering-1161e7270a40
2. C. Manning, P. Raghavan, and H. Schütze, ―Introduction to Information Retrieval, Cambridge University Press
3. https://www.analyticsvidhya.com/blog/2020/10/quick-guide-to-evaluation-metrics-for-supervised-and-unsupervised-machine-learning/
4. https://towardsdatascience.com/7-evaluation-metrics-for-clustering-algorithms-bdc537ff54d2/
5. Samreen Naeem et.al- An Unsupervised Machine Learning Algorithms: Comprehensive Review, International Journal of Computing and Digital Systems ISSN (2210-142X) Int. J. Com. Dig. Sys.13, No.1 (Apr-23).