VIRGINIA COMMONWEALTH UNIVERSITY

Statistical Analysis and Modelling (SCMA 632)

A4B: Cluster Analysis

NIHARIHA KAMALANATHAN
V01108259
Date of Submission: 08-07-2024

CONTENTS

PART B: Conduct Cluster Analysis to characterize respondents based on background variables (Survey.csv)

Introduction

The dataset used for this analysis is derived from a survey aimed at understanding various socio-economic indicators related to housing preferences and expenditures. This dataset, denoted as "Survey.csv," contains a wide range of variables including income, proximity to amenities (such as schools, transport, and shopping centers), and preferences for housing features (like parking space, gym facilities, and security). In this analysis, we perform a cluster analysis to group respondents based on these background variables. Clustering helps identify distinct groups within the data, providing insights into different socio-economic segments and their housing preferences.

Business Significance

1. **Targeted Housing Solutions**: By identifying distinct clusters of respondents based on their socio-economic characteristics and preferences, real estate developers can design and market housing solutions tailored to specific demographic groups. For example, luxury apartments with extensive amenities can be targeted towards high-income clusters, while affordable housing with essential features can be targeted towards moderate-income clusters.
2. **Enhanced Marketing Strategies**: Businesses in the housing and home improvement sectors can use these insights to tailor their marketing strategies. Understanding the unique preferences of different clusters allows for more precise targeting, ensuring that marketing messages and product offerings resonate with the intended audience.
3. **Policy Formulation and Urban Planning**: Government agencies and urban planners can utilize the clustering results to design and implement policies that address the specific needs of different socio-economic groups. For instance, improving public transportation and school proximity in areas where these factors are highly valued can enhance overall community satisfaction and living standards.
4. **Investment and Financial Planning**: Financial institutions and investors can use the clustering insights to assess the demand for various types of housing in different regions. This can guide investment decisions and financial planning, ensuring resources are allocated to projects with the highest potential return on investment.

Objectives

1. **To Perform Cluster Analysis**: To conduct a cluster analysis on the "Survey.csv" dataset to identify distinct groups of respondents based on background variables such as income, proximity to amenities, and housing feature preferences.
2. **To Determine Optimal Number of Clusters**: To use methods such as the Elbow method and Silhouette method to determine the optimal number of clusters, ensuring meaningful and interpretable groupings within the data.
3. **To Visualize Clusters**: To create visualizations that illustrate the distribution and separation of the clusters, providing a clear understanding of the distinct groups identified in the analysis.
4. **To Characterize Clusters**: To summarize and interpret the characteristics of each cluster by analyzing the mean values of key variables, offering insights into the socio-economic profiles and preferences of each group.
5. **To Provide Business and Policy Recommendations**: To translate the findings into actionable insights for businesses, policymakers, and urban planners. This includes

discussing the practical implications of the clustering results and how they can inform decision-making processes in housing development, marketing, and urban planning.

By achieving these objectives, the analysis aims to provide a comprehensive understanding of the different socio-economic segments within the dataset and offer valuable insights for improving housing solutions, marketing strategies, and policy-making.

R LANGUAGE

1. Installation and Loading of Necessary Packages

```
# Install and load necessary packages
install.packages(c("tidyverse", "cluster", "factoextra", "dendextend", "GGally", "pheatmap", "MASS", "Rtsne"))
library(tidyverse)
library(cluster)
library(factoextra)
library(dendextend)
library(GGally)
library(pheatmap)
library(MASS)
library(Rtsne)
```

Purpose:
This block installs and loads various R packages necessary for data manipulation (tidyverse), clustering (cluster, factoextra, dendextend), visualizations (GGally, pheatmap), and dimensionality reduction (MASS, Rtsne).

2. Loading the Data

```
# Load the data
file_path <- "C:\\Users\\nihar\\OneDrive\\Desktop\\Bootcamp\\SCMA 632\\DataSet\\Survey.csv"
survey_data <- read.csv(file_path)
```

Purpose:
This code loads the survey data from a CSV file into the R environment.

3. Handling Missing Values

```
# Handle missing values (if any)
survey_data <- na.omit(survey_data)
```

Purpose:
This line removes any rows with missing values from the dataset to ensure clean data for analysis.

4. Identifying and Separating Numeric Columns

```
# Identify numeric columns
numeric_columns <- sapply(survey_data, is.numeric)

# Separate numeric and non-numeric data
numeric_data <- survey_data[, numeric_columns]
```

```
non_numeric_data <- survey_data[, !numeric_columns]
```

Purpose:
This section identifies numeric columns in the dataset and separates numeric data from non-numeric data. This is essential for standardization and clustering steps that follow.

5. Standardizing the Numeric Data

```
# Standardize the numeric data
survey_data_scaled <- scale(numeric_data)
```

Purpose:
Standardization ensures that all numeric variables contribute equally to the clustering process by having a mean of 0 and a standard deviation of 1.

6. Determining the Optimal Number of Clusters

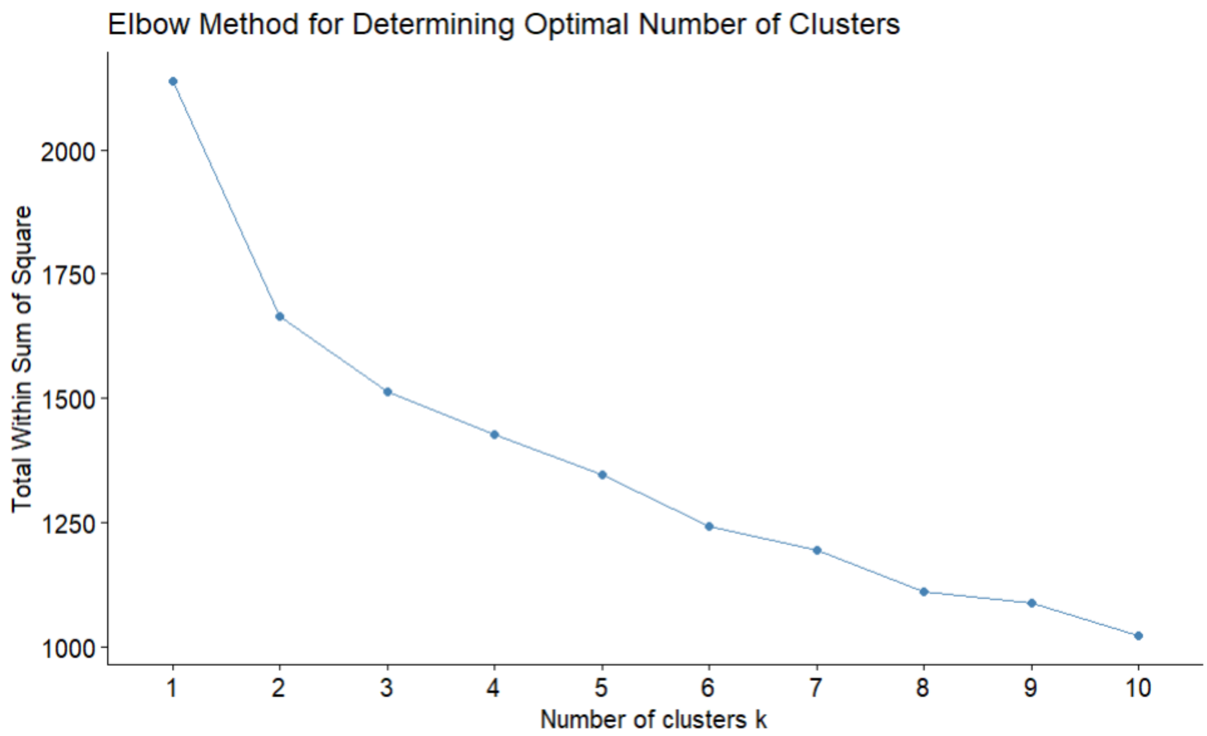
```
# Determine the optimal number of clusters using Elbow and Silhouette methods
fviz_nbclust(survey_data_scaled, kmeans, method = "wss") +
  labs(title = "Elbow Method for Determining Optimal Number of Clusters")

fviz_nbclust(survey_data_scaled, kmeans, method = "silhouette") +
  labs(title = "Silhouette Method for Determining Optimal Number of Clusters")
```

Purpose:
These lines use the Elbow and Silhouette methods to determine the optimal number of clusters for the K-means algorithm.

Output:

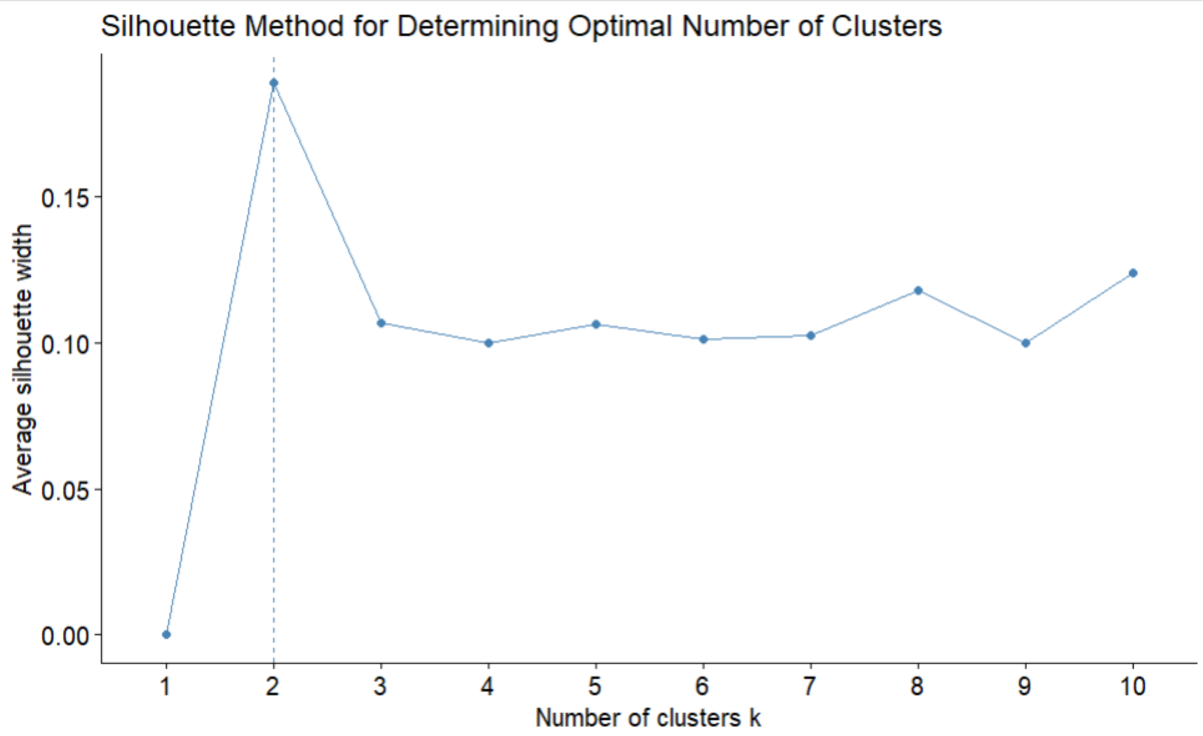
- **Elbow Method Plot**:



Interpretation:
The Elbow Method plot shows the total within-cluster sum of squares (WSS) as a function of

the number of clusters. The "elbow point" is where the WSS starts to level off, suggesting the optimal number of clusters. Here, it appears around 3 clusters.

- **Silhouette Method Plot**:



Interpretation:
The Silhouette Method plot shows the average silhouette width for different numbers of clusters. A higher silhouette width indicates better-defined clusters. The plot suggests 2 clusters, but since both 2 and 3 have high silhouette values, we continue with 3 clusters as indicated by the Elbow Method.

7. Performing K-means Clustering

```
# Perform K-means clustering (assuming 3 clusters as optimal)
set.seed(123)
kmeans_result <- kmeans(survey_data_scaled, centers = 3, nstart = 25)
```

Purpose:
This block performs K-means clustering on the standardized data with 3 clusters, ensuring reproducibility with set.seed.

8. Adding Cluster Assignment to Original Data

```
# Add the cluster assignment to the original data
survey_data$cluster <- as.factor(kmeans_result$cluster)
```

Purpose:
This line adds the cluster assignment from the K-means result to the original survey data, making it easier to analyze and visualize the clusters.

9. Visualizing Clusters

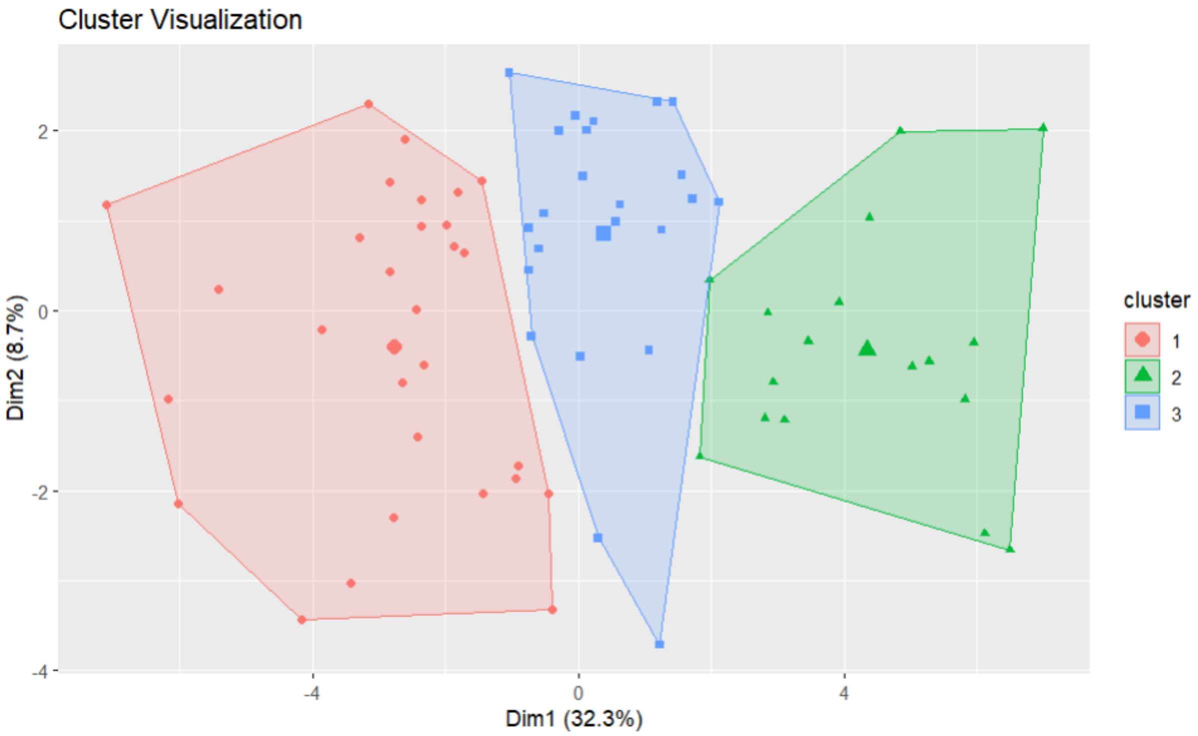
```
# Visualize clusters
```

```
fviz_cluster(kmeans_result, data = survey_data_scaled,
             geom = "point", ellipse.type = "convex") +
  labs(title = "Cluster Visualization")
```

Purpose:
This visualization helps in understanding the distribution and separation of the clusters.

Output:

- **Cluster Visualization Plot**:



Interpretation:
The plot shows three distinct clusters in a two-dimensional space, with each cluster represented by a different color and shape. This indicates a good separation between clusters, validating the choice of 3 clusters.

10. Cluster Interpretation and Characterization

```
# Interpretation and Characterization
cluster_summary <- survey_data %>%
  select_if(is.numeric) %>%
  bind_cols(cluster = survey_data$cluster) %>%
  group_by(cluster) %>%
  summarise_all(mean, na.rm = TRUE)

print(cluster_summary)
```

Purpose:
This block summarizes the clusters by calculating the mean values of numeric variables for each cluster.

Output:

- **Cluster Summary**:

```
# A tibble: 3 × 32
  cluster  Income X1.Proximity.to.city X2.Proximity.to.schools X3..Proximity.to.tra…¹
  <fct>     <dbl>                <dbl>                   <dbl>                 <dbl>
1 1        56333.                 3.17                    3.3                   4.13
2 2       195000                  4                       3.94                  4.18
3 3        83696.                 3.96                    3.26                  3.91
# ℹ abbreviated name: ¹X3..Proximity.to.transport
# ℹ 27 more variables: X4..Proximity.to.work.place <dbl>,
#   X5..Proximity.to.shopping <dbl>, X1..Gym.Pool.Sports.facility <dbl>,
#   X2..Parking.space <dbl>, X3.Power.back.up <dbl>, X4.Water.supply <dbl>,
#   X5.Security <dbl>, X1..Exterior.look <dbl>, X2..Unit.size <dbl>,
#   X3..Interior.design.and.branded.components <dbl>,
#   X4..Layout.plan..Integrated.etc.. <dbl>, X5..View.from.apartment <dbl>, …
```

Interpretation:

- **Cluster 1**:
  - Average Income: $56,333
  - Proximity to City: 3.17
  - Proximity to Schools: 3.3
  - Proximity to Transport: 4.13
- **Cluster 2**:
  - Average Income: $195,000
  - Proximity to City: 4
  - Proximity to Schools: 3.94
  - Proximity to Transport: 4.18
- **Cluster 3**:
  - Average Income: $83,696
  - Proximity to City: 3.96
  - Proximity to Schools: 3.26
  - Proximity to Transport: 3.91

Cluster 2 has the highest average income and places high importance on proximity to the city and transport. Cluster 1 has a moderate income and values transport proximity the most. Cluster 3 has a higher-than-moderate income with balanced priorities.

PYTHON LANGUAGE

1. Import Necessary Libraries

Input:

```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.cluster import KMeans
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import pairwise_distances_argmin_min
from scipy.cluster.hierarchy import dendrogram, linkage
from sklearn.decomposition import PCA
from pheatmap import pheatmap
```

Purpose:

- Import essential libraries for data manipulation (Pandas, Numpy), visualization (Seaborn, Matplotlib), clustering (KMeans, hierarchical clustering), and preprocessing (StandardScaler).

Output:

- No direct output; these libraries provide the functionality for subsequent steps.

2. Load the Dataset

Input:

```
def load_data(file_path):
    return pd.read_csv(file_path)

file_path = "C:\\Users\\nihar\\OneDrive\\Desktop\\Bootcamp\\SCMA 632\\DataSet\\Survey.csv"
survey_df = load_data(file_path)
```

Purpose:

- Define a function to load a CSV file into a Pandas DataFrame and then load the specific dataset.

Output:

- survey_df: DataFrame containing the loaded survey data.

Interpretation:

- Data is successfully loaded into a DataFrame for further analysis.

3. Select Relevant Columns

Input:

```
sur_int = survey_df.iloc[:, 19:46]
```

Purpose:

- Select specific columns (from index 19 to 45) from the DataFrame for analysis.

Output:

- sur_int: DataFrame containing only the selected columns.

Interpretation:

- Narrow down the dataset to the columns of interest for clustering analysis.

4. Standardize the Data

Input:

```
scaler = StandardScaler()
sur_int_scaled = scaler.fit_transform(sur_int)
```

Purpose:

- Standardize the selected columns to have a mean of 0 and a standard deviation of 1.

Output:

- sur_int_scaled: Numpy array of standardized data.

Interpretation:

- Standardized data ensures that all features contribute equally to the clustering algorithm.

5. Determine the Optimal Number of Clusters using the Elbow Method

Input:

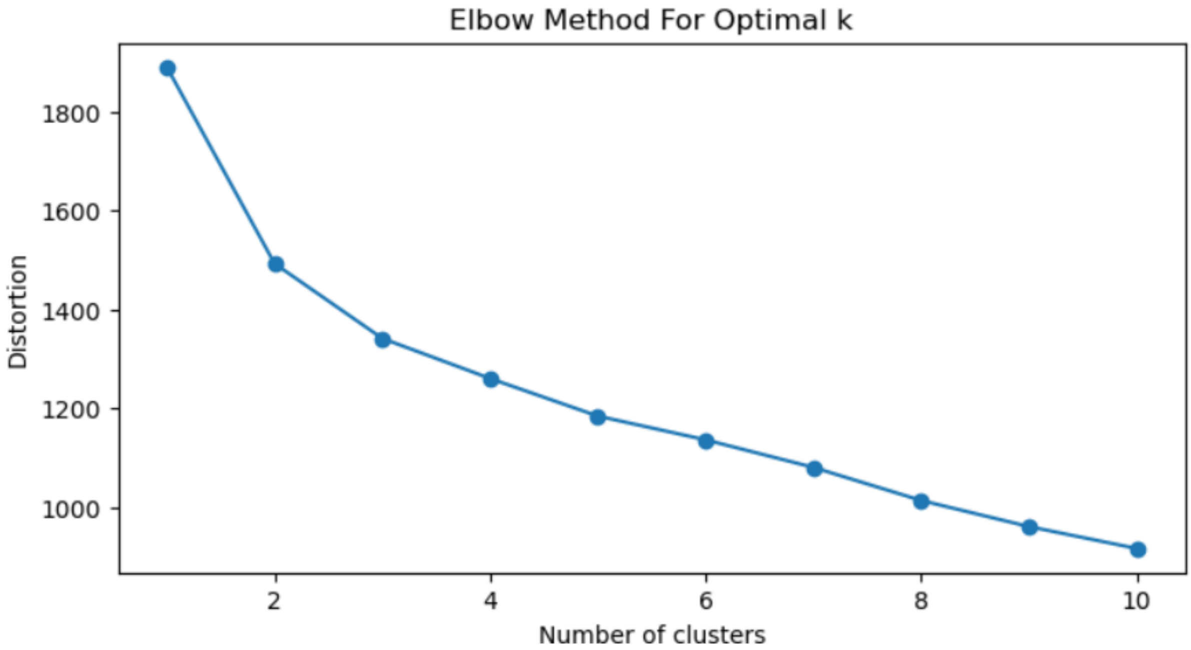
```
def optimal_kmeans(data, max_k=10):
    distortions = []
    for k in range(1, max_k+1):
        kmeans = KMeans(n_clusters=k, n_init=25, random_state=123)
        kmeans.fit(data)
        distortions.append(kmeans.inertia_)
    return distortions

distortions = optimal_kmeans(sur_int_scaled)
plt.figure(figsize=(8, 4))
plt.plot(range(1, 11), distortions, marker='o')
plt.title('Elbow Method For Optimal k')
plt.xlabel('Number of clusters')
plt.ylabel('Distortion')
plt.show()
```

Purpose:

- Implement the Elbow method to determine the optimal number of clusters by fitting KMeans for a range of cluster numbers and plotting the distortion (inertia) for each.

Output:



- Elbow plot showing distortions for different numbers of clusters.

Interpretation:

- The "elbow point" on the plot, where the rate of distortion decrease sharply slows, indicates the optimal number of clusters. In this case, it appears around 4 clusters.

6. Apply KMeans with the Optimal Number of Clusters
Input:

```
optimal_clusters = 4
kmeans = KMeans(n_clusters=optimal_clusters, n_init=25, random_state=123)
km_res = kmeans.fit(sur_int_scaled)
```

Purpose:

- Apply KMeans clustering with the determined optimal number of clusters (4).

Output:

- km_res: Result of KMeans clustering containing cluster labels and other details.

Interpretation:

- Data points are clustered into 4 groups based on similarity.

7. Visualize the Clusters
Input:

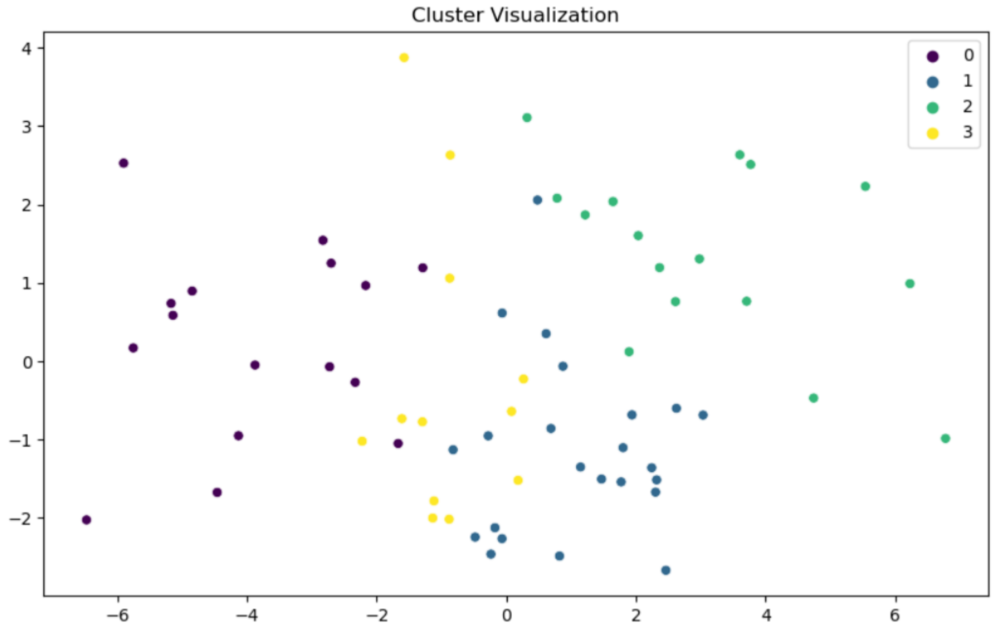
```
pca = PCA(2)
pca_transformed = pca.fit_transform(sur_int_scaled)

plt.figure(figsize=(10, 6))
sns.scatterplot(x=pca_transformed[:, 0], y=pca_transformed[:, 1], hue=km_res.labels_,
palette='viridis')
plt.title('Cluster Visualization')
plt.show()
```

Purpose:

- Use PCA to reduce data to two dimensions and visualize the clusters with a scatter plot.

Output:



- Scatter plot showing clusters in two-dimensional space.

Interpretation:

- Different clusters are visualized in 2D space, colored differently to show the grouping. This helps in understanding the distribution and separation of clusters.

8. Hierarchical Clustering and Dendrogram

Input:

```
linked = linkage(sur_int_scaled, 'ward')
plt.figure(figsize=(10, 7))
dendrogram(linked, truncate_mode='level', p=5, show_leaf_counts=False, no_labels=True,
color_threshold=0.7 * np.max(linked[:, 2]), above_threshold_color='grey')
plt.title('Hierarchical Clustering Dendrogram')
plt.xlabel('Sample index')
plt.ylabel('Distance')
plt.show()
```

Purpose:

- Perform hierarchical clustering and visualize it using a dendrogram.

Output:

- Dendrogram showing the hierarchical clustering of the data.

Interpretation:

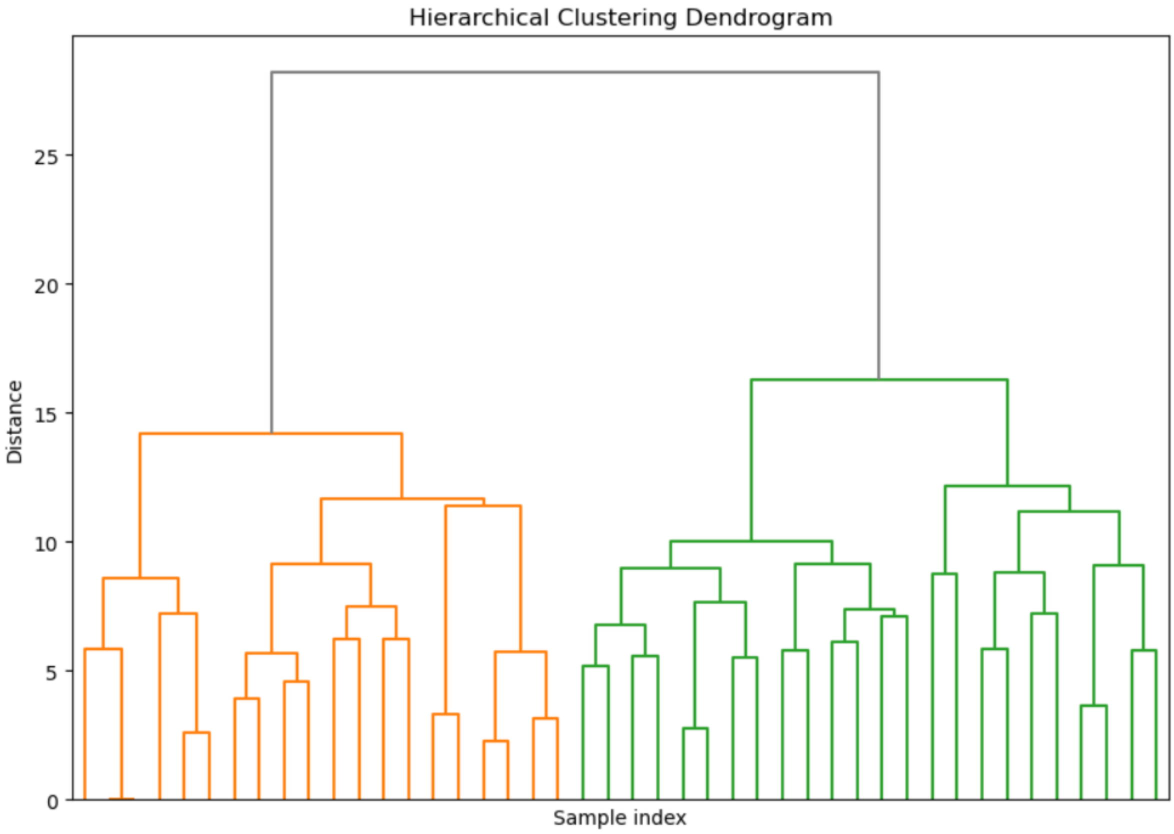
- The dendrogram shows how data points are grouped hierarchically. The x-axis represents the data points, and the y-axis represents the distance or dissimilarity. The height of the branches indicates the distance at which clusters are merged.

9. Heatmap with Dendrogram

Input:

```
sns.clustermap(sur_int, method='ward', col_cluster=True, row_cluster=False, cmap='viridis',
figsize=(10, 10))
plt.show()
```

Purpose:



- Plot a heatmap of the data with hierarchical clustering shown on the columns.

Output:

- Heatmap with dendrogram showing hierarchical clustering of columns.

Interpretation:

- The heatmap provides a visual representation of the data matrix, highlighting patterns and similarities. The dendrogram on top shows the hierarchical clustering of the columns, indicating which columns are similar to each other.

Overview of Cluster Analysis

Meaning of Cluster Analysis

Cluster Analysis is a statistical technique that aims to group similar objects into clusters. The objects within a cluster share more similarities with each other than with those in other clusters. This technique helps in identifying patterns and structures within a dataset, providing insights that might not be immediately apparent.

Advantages of Cluster Analysis

1. **Identifies Patterns:**
   - Cluster analysis helps in discovering natural groupings within data, which can lead to significant insights. For example, understanding customer segments in a retail business.
2. **Data Reduction:**
   - By grouping similar data points, cluster analysis reduces the complexity of the data, making it easier to understand and interpret.
3. **Improved Decision Making:**

    - Highlighting significant groupings and differences in data assists in making informed decisions. For instance, identifying distinct customer segments can lead to better marketing strategies.
4. **Market Segmentation:**
    - Businesses can segment their markets or customer bases into distinct groups, leading to more targeted and effective marketing strategies.
5. **Anomaly Detection:**
    - Cluster analysis can identify outliers or anomalies in the data, which might indicate errors, fraud, or unusual behavior.
6. **Enhancement of Other Techniques:**
    - It can be used as a preprocessing step for other techniques such as classification, regression, and association rule mining, improving their performance.

Real-Life Examples of Cluster Analysis

1. **Customer Segmentation:**
    - Businesses use cluster analysis to segment their customer base into distinct groups based on purchasing behavior, demographics, and other attributes. This allows for personalized marketing strategies and improved customer service.
2. **Image Segmentation:**
    - In computer vision, cluster analysis partitions an image into segments or clusters of pixels, making it easier to identify and analyze objects within the image.
3. **Healthcare:**
    - Medical researchers use cluster analysis to identify groups of patients with similar symptoms or responses to treatment, which can help in diagnosing diseases and developing personalized treatment plans.
4. **Market Research:**
    - Market researchers use cluster analysis to group respondents with similar preferences and behaviors, aiding in the development of targeted products and services.
5. **Document Clustering:**
    - Used in information retrieval to group similar documents together, improving the efficiency and accuracy of search engines and recommendation systems.
6. **Social Network Analysis:**
    - Identifies communities or groups within social networks based on interactions and relationships, which can be used to understand social dynamics and influence.