VIRGINIA COMMONWEALTH UNIVERSITY

Statistical analysis and modelling (SCMA 632)

A4.2: Cluster Analysis

SAYA SANTHOSH

V01101901

Date of Submission: 08-07-2024

CONTENTS

Introduction

Cluster analysis :
For this assignment, background factors from the "Survey.csv" dataset will be used to profile respondents using a thorough cluster analysis. The dataset includes a range of socioeconomic and demographic characteristics, such as age, income, work status, and degree of education. The main objective is to use clustering techniques, such KMeans, to separate individuals with similar features and find unique groups within the respondent population in order to discover patterns. The best number of clusters will be determined using techniques like silhouette analysis and the Elbow approach after preparing the data to handle missing values, encode categorical variables, and scale features. This study attempts to offer a deeper knowledge of the respondent profiles by looking at the generated clusters. This insight can help with focused interventions, policy-making, and customized marketing strategies. By using this thorough cluster analysis, the assignment not only improves abilities in exploratory data analysis and unsupervised learning, but it also shows how clustering can be used practically to comprehend and handle the varied backgrounds of respondents.

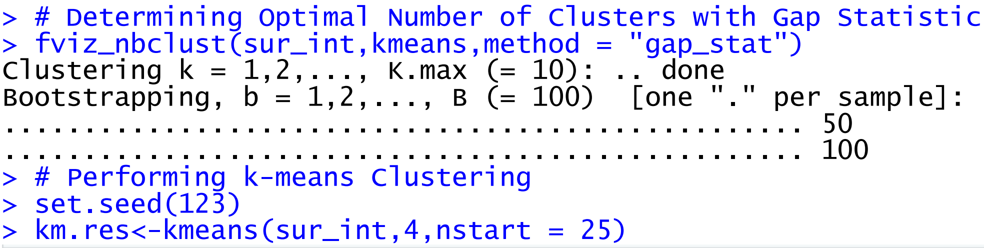
Objectives :

- Identify distinct respondent groups
- Understand respondent profiles
- Perform data preprocessing
- Determine optimal number of clusters
- Apply cluster analysis
- Evaluate clustering results

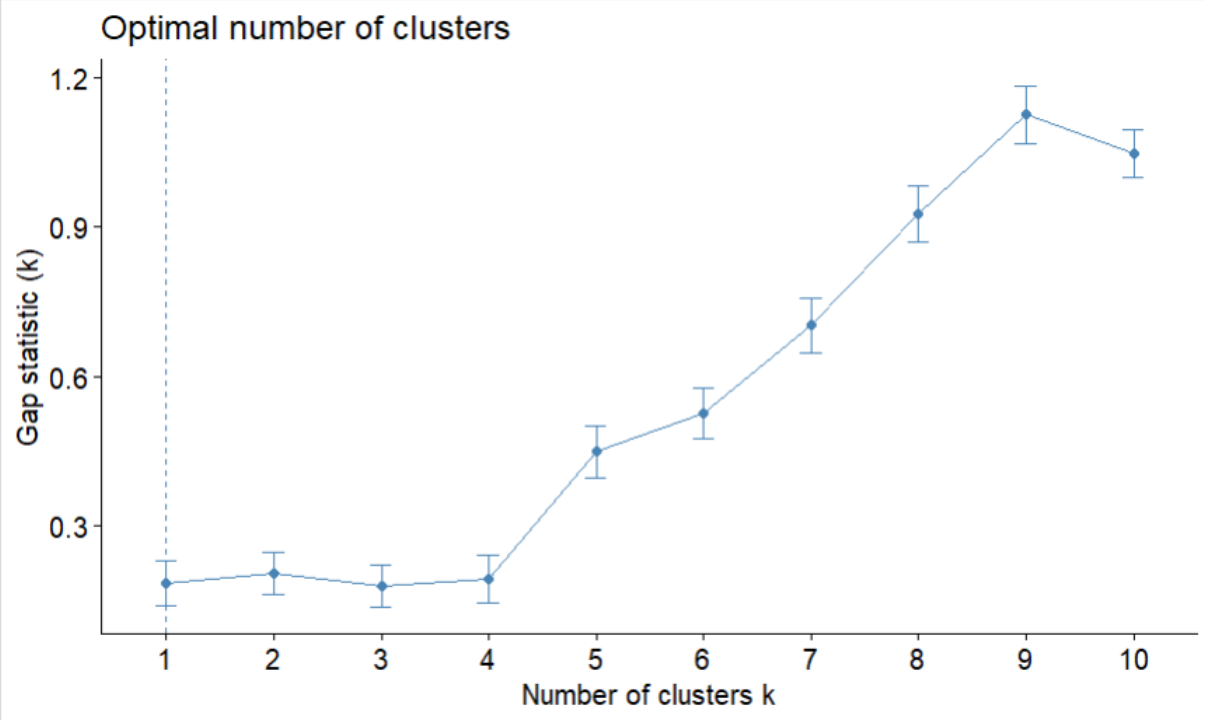
Business Significance :

Applying cluster analysis to respondents' background variable-based characterization has important business ramifications in a number of sectors. Businesses can more successfully adjust their products and services to fit the varied demands and preferences of various client segments by identifying separate respondent groups. By using clustering to understand responder profiles, personalized marketing techniques that appeal to particular demographic or behavioral trends can be implemented, improving consumer happiness and engagement. In addition, cluster analysis makes it easier to find high-value markets with more room for revenue expansion and client retention. Businesses can enhance their competitive edge in a constantly changing market by optimizing resource allocation, streamlining operational efficiency, and creating customized strategies through the utilization of insights obtained by cluster analysis. In the end, this analytical method ensures continued business success and relevance by enhancing decision-making procedures and encouraging creativity and response to market changes.

Results and Interpretation using R



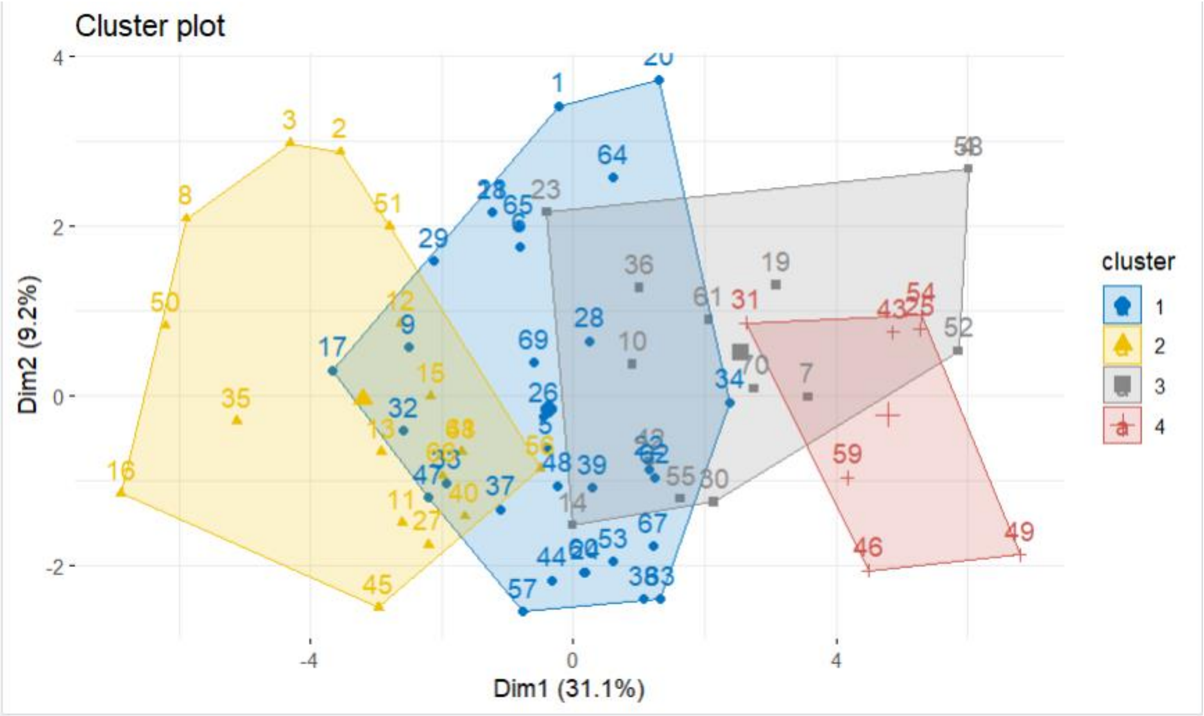
```
> # Determining Optimal Number of Clusters with Gap Statistic
> fviz_nbclust(sur_int,kmeans,method = "gap_stat")
Clustering k = 1,2,..., K.max (= 10): .. done
Bootstrapping, b = 1,2,..., B (= 100)  [one "." per sample]:
.................................................. 50
.................................................. 100
> # Performing k-means Clustering
> set.seed(123)
> km.res<-kmeans(sur_int,4,nstart = 25)
```



Interpretation:

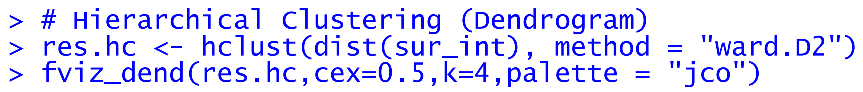
The provided graph illustrates the determination of the optimal number of clusters using the Gap Statistic method. The x-axis represents the number of clusters (k), while the y-axis shows the Gap Statistic values. The plot indicates that the Gap Statistic increases significantly as the number of clusters increases, peaking at k=9 before decreasing. The optimal number of clusters is typically chosen as the value of k where the Gap Statistic first reaches its maximum value, which in this case is around 9 clusters. This suggests that segmenting the data into 9 distinct clusters would be most appropriate for capturing the underlying structure of the dataset.

```
> # Visualizing k-means Clustering Results
> fviz_cluster(km.res,data=sur_int,palette="jco", ggtheme = theme_minimal(
))
```

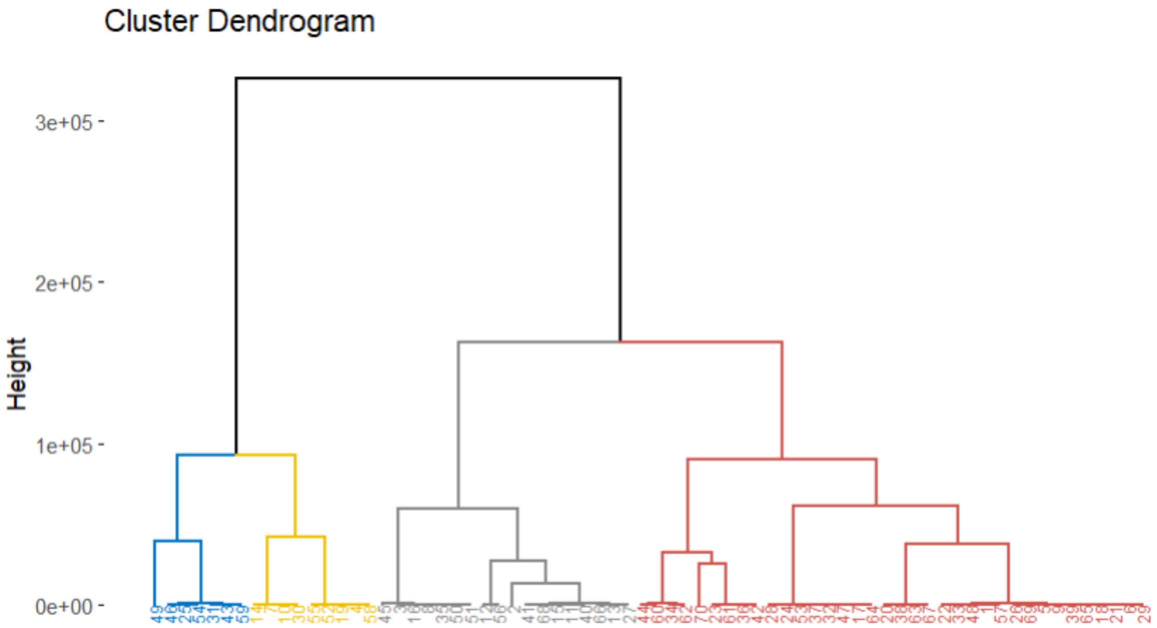


Interpretation:

The cluster plot visualizes the results of k-means clustering on the dataset, displaying four distinct clusters in a two-dimensional space defined by Dim1 (31.1% of the variance) and Dim2 (9.2% of the variance). Each cluster is represented by a different color and shape: blue circles for cluster 1, yellow triangles for cluster 2, grey squares for cluster 3, and red crosses for cluster 4. The plot highlights the separation and overlap between clusters, showing that clusters 1 and 2 have some overlap, while clusters 3 and 4 are more distinct. This visualization aids in understanding the distribution and characteristics of the clusters within the dataset .



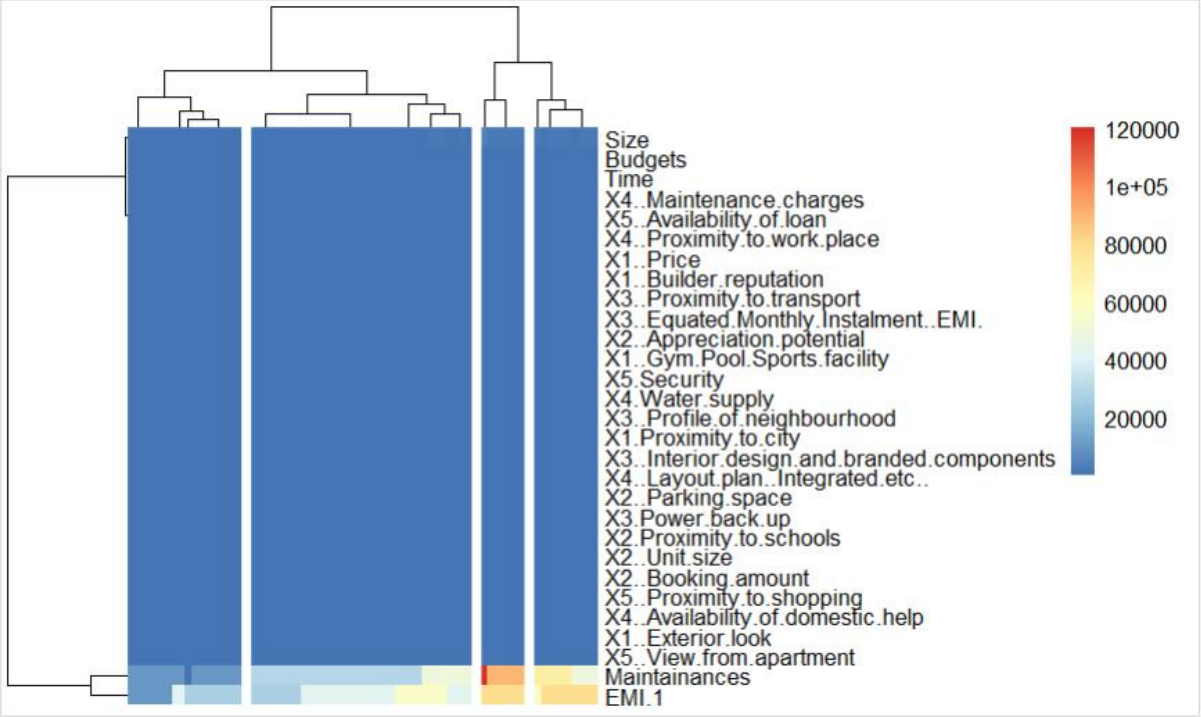
```
> # Hierarchical Clustering (Dendrogram)
> res.hc <- hclust(dist(sur_int), method = "ward.D2")
> fviz_dend(res.hc,cex=0.5,k=4,palette = "jco")
```



Interpretation:

The image is a dendrogram, a tree-like diagram used to visualize the arrangement of clusters produced by hierarchical clustering. The y-axis represents the height, which is a measure of dissimilarity or distance between clusters. In this dendrogram, the colors (blue, yellow, gray, and red) indicate different clusters formed at various stages of the clustering process. The branches merge at higher heights, showing the points at which clusters are combined. The larger the height at which two clusters merge, the more dissimilar they are. The longest vertical line (black) indicates the most significant split in the data, separating the clusters that are most dissimilar from each other.

```
> #  Heatmap of Clustered Data
> library(pheatmap)
> pheatmap(t(sur_int),cutree_cols = 4)
```

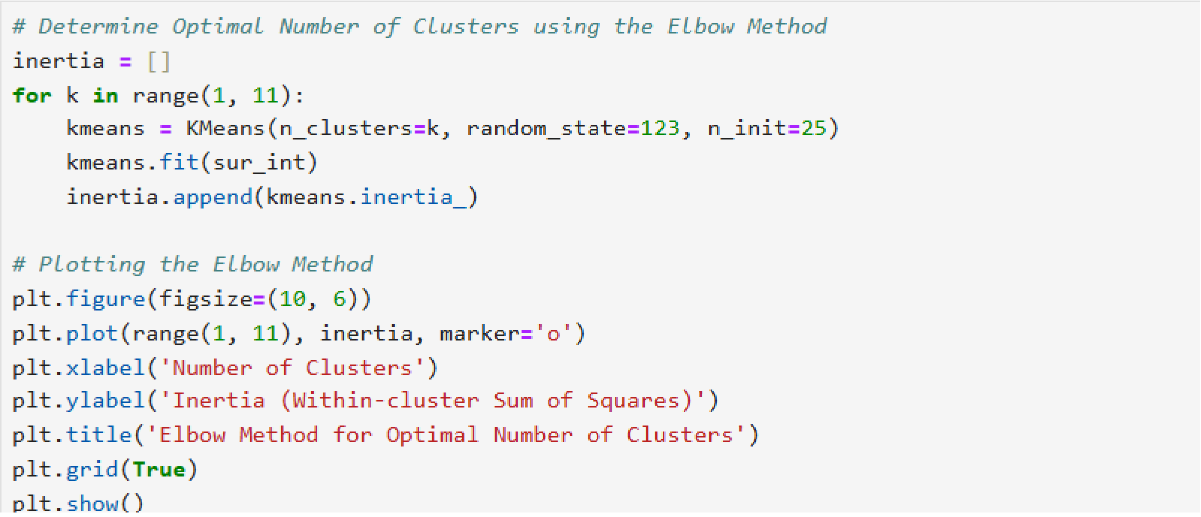


Interpretation:

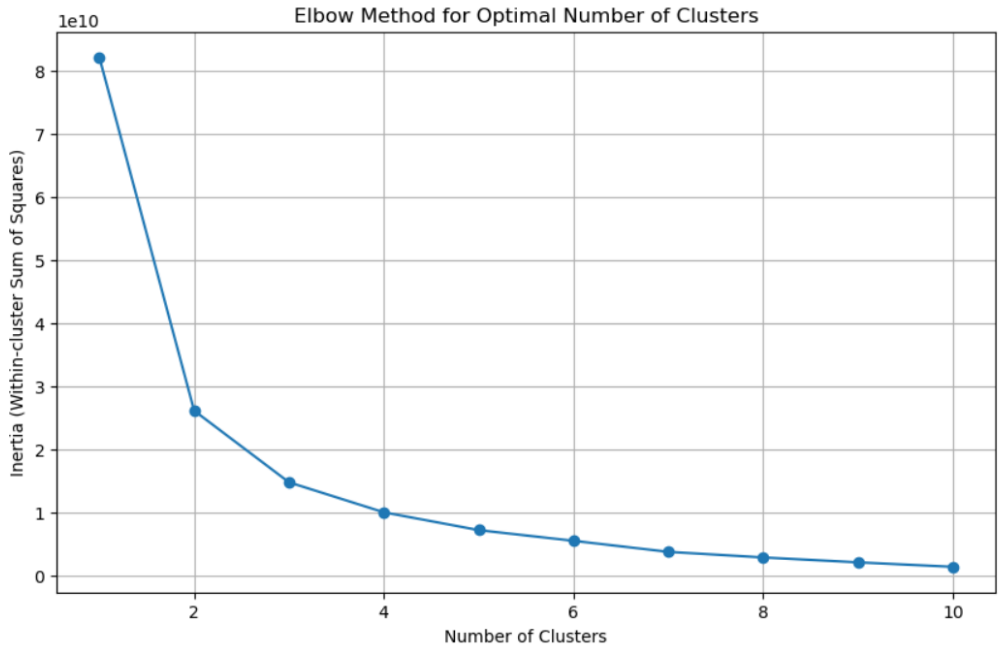
The image is a clustered heatmap, which combines a heatmap with a dendrogram. The rows represent different variables, while the columns represent samples or observations. The colors in the heatmap indicate the values of these variables, with the color scale on the right showing the range from lower values (blue) to higher values (red). The dendrogram on the top groups similar samples based on the variables, while the dendrogram on the left groups similar variables based on their values across the samples.

In this particular heatmap, we see a large block of blue, indicating low values for most variables across the majority of samples. However, some areas with yellow to red colors indicate higher values for certain variables in specific samples. The dendrograms help in identifying clusters of variables and samples that exhibit similar patterns. For example, variables like "Maintenance charges" and "Availability of loan" might be grouped together if they show similar values across samples. This visualization is useful for identifying patterns, correlations, and potential outliers in the data.

Results and Interpretation using Python



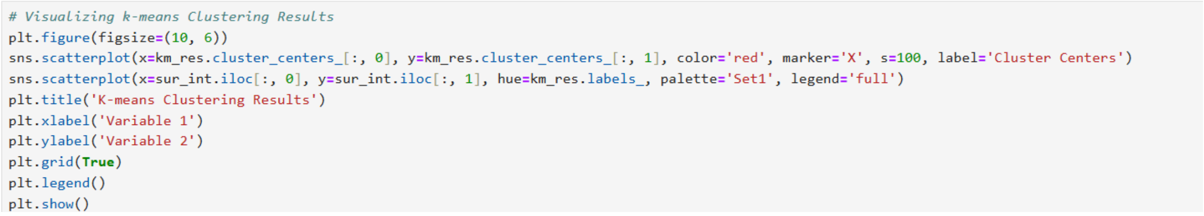
```python
# Determine Optimal Number of Clusters using the Elbow Method
inertia = []
for k in range(1, 11):
    kmeans = KMeans(n_clusters=k, random_state=123, n_init=25)
    kmeans.fit(sur_int)
    inertia.append(kmeans.inertia_)

# Plotting the Elbow Method
plt.figure(figsize=(10, 6))
plt.plot(range(1, 11), inertia, marker='o')
plt.xlabel('Number of Clusters')
plt.ylabel('Inertia (Within-cluster Sum of Squares)')
plt.title('Elbow Method for Optimal Number of Clusters')
plt.grid(True)
plt.show()
```



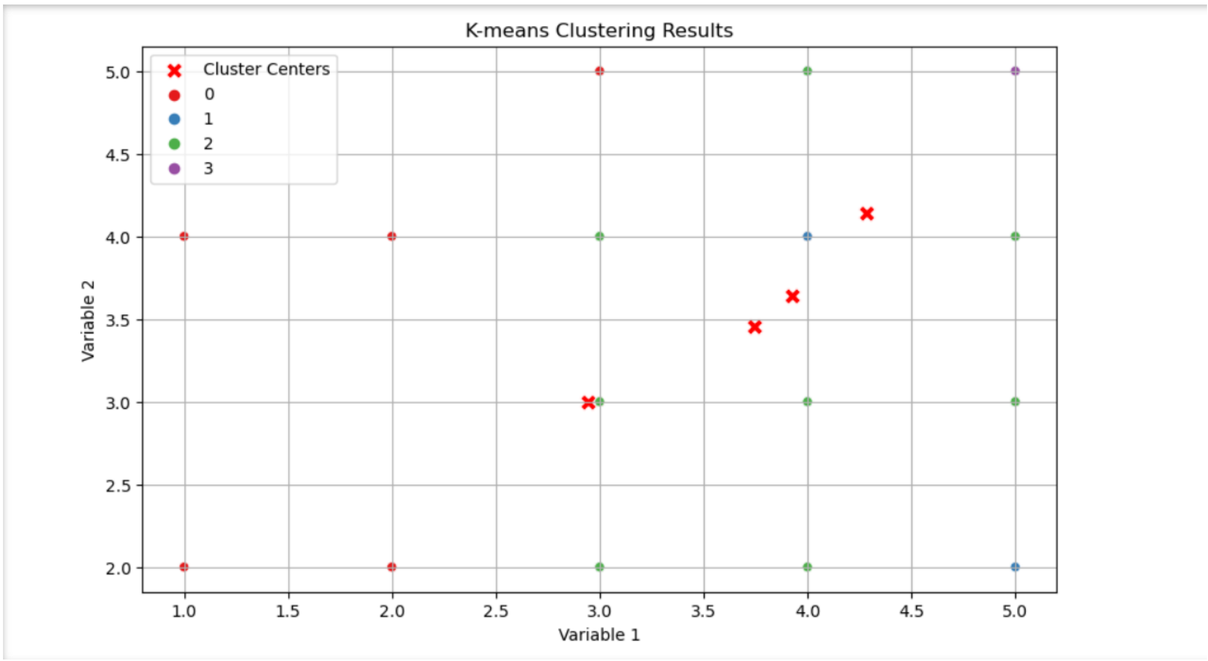
Interpretation:

The image is an elbow plot, which is used to determine the optimal number of clusters in a dataset for k-means clustering. The x-axis represents the number of clusters, while the y-axis represents the inertia (within-cluster sum of squares). As the number of clusters increases, the inertia decreases, indicating that the clusters are becoming more compact.

In this plot, there is a sharp decline in inertia from 1 to 2 clusters, followed by a more gradual decline. The "elbow" point, where the rate of decrease in inertia slows down significantly, is around 3 clusters. This suggests that 3 clusters might be the optimal number, as adding more clusters beyond this point results in only a marginal reduction in inertia. Thus, the elbow plot helps to balance the trade-off between the number of clusters and the improvement in clustering performance.

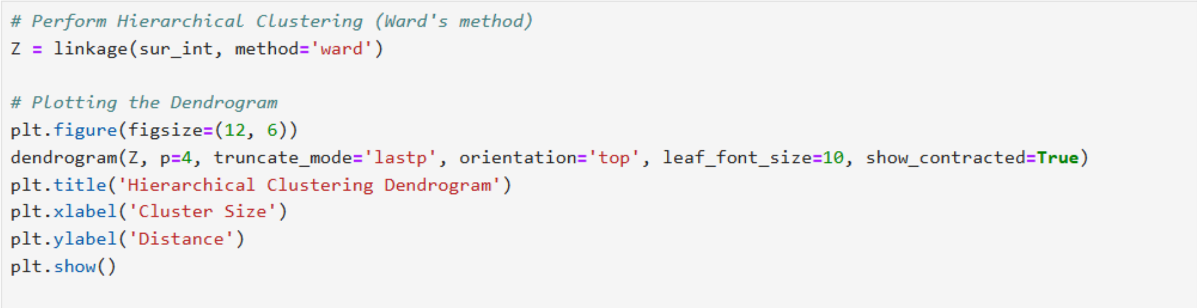


```python
# Visualizing k-means Clustering Results
plt.figure(figsize=(10, 6))
sns.scatterplot(x=km_res.cluster_centers_[:, 0], y=km_res.cluster_centers_[:, 1], color='red', marker='X', s=100, label='Cluster Centers')
sns.scatterplot(x=sur_int.iloc[:, 0], y=sur_int.iloc[:, 1], hue=km_res.labels_, palette='Set1', legend='full')
plt.title('K-means Clustering Results')
plt.xlabel('Variable 1')
plt.ylabel('Variable 2')
plt.grid(True)
plt.legend()
plt.show()
```

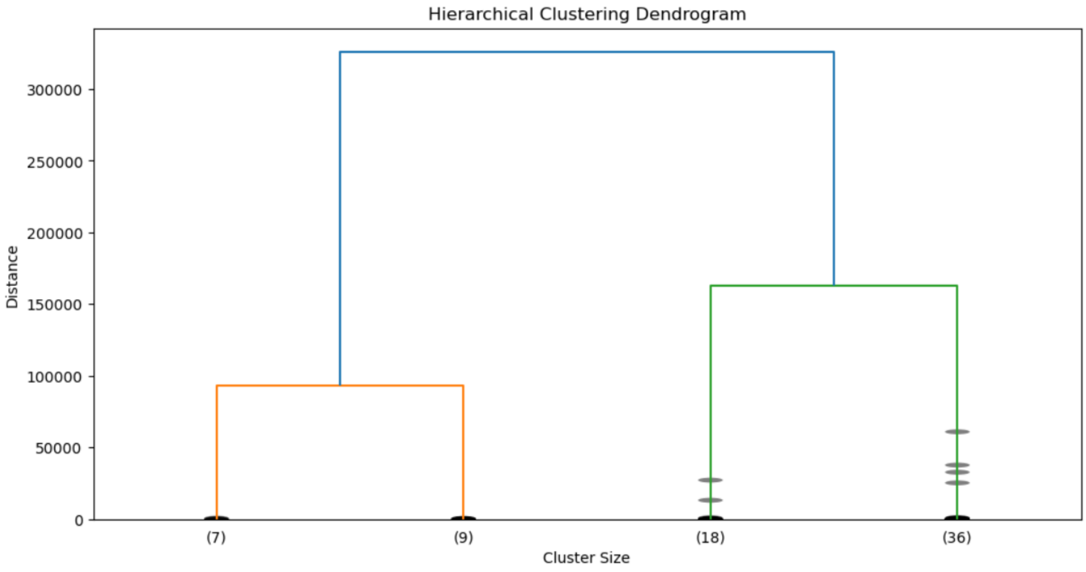


Interpretation:

The image displays the results of a K-means clustering analysis, showing data points plotted on a two-dimensional plane with respect to two variables. The data points are color-coded to represent different clusters: blue for cluster 0, red for cluster 1, green for cluster 2, and purple for cluster 3. Additionally, the red crosses indicate the centroids of each cluster. The centroids are positioned centrally among their respective cluster points, demonstrating the representative center of each group. This plot helps in visualizing the separation and distribution of the clusters, showing that the K-means algorithm has effectively grouped the data into four distinct clusters based on the variables considered.



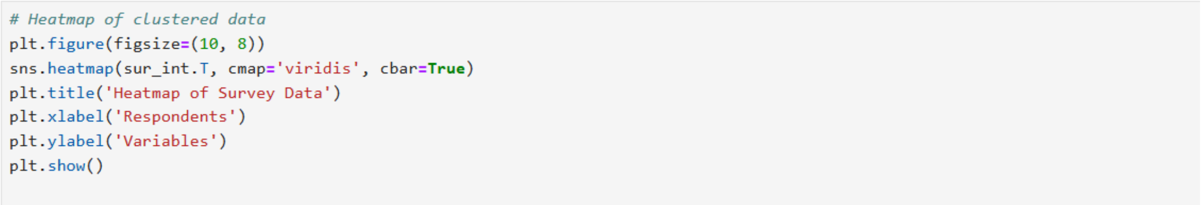
```python
# Perform Hierarchical Clustering (Ward's method)
Z = linkage(sur_int, method='ward')

# Plotting the Dendrogram
plt.figure(figsize=(12, 6))
dendrogram(Z, p=4, truncate_mode='lastp', orientation='top', leaf_font_size=10, show_contracted=True)
plt.title('Hierarchical Clustering Dendrogram')
plt.xlabel('Cluster Size')
plt.ylabel('Distance')
plt.show()
```



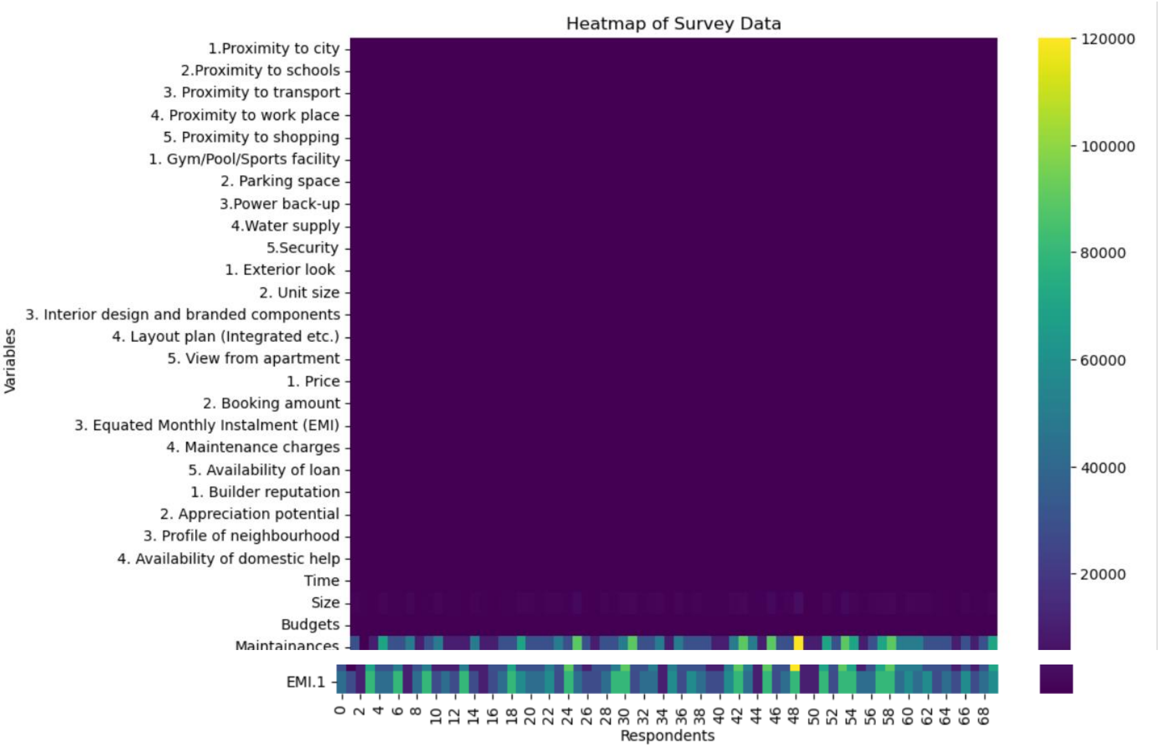
Interpretation:

The image displays a hierarchical clustering dendrogram, which illustrates the process of clustering data points based on their similarity. Each vertical line represents a cluster, and the height at which two clusters are joined reflects the distance or dissimilarity between them. The x-axis shows the cluster size, while the y-axis represents the distance or dissimilarity between clusters.

From the dendrogram, we can observe that the data points are initially grouped into smaller clusters, which are then merged into larger clusters as we move up the diagram. The largest clusters are represented by the highest vertical lines, indicating the highest level of dissimilarity before merging. The clusters at the bottom with labels (7), (9), (18), and (36) represent the initial individual data points or small clusters before they are combined into larger clusters. The dendrogram helps in determining the number of clusters and understanding the hierarchical relationships among the data points.



```
# Heatmap of clustered data
plt.figure(figsize=(10, 8))
sns.heatmap(sur_int.T, cmap='viridis', cbar=True)
plt.title('Heatmap of Survey Data')
plt.xlabel('Respondents')
plt.ylabel('Variables')
plt.show()
```



Interpretation:

The image shows a heatmap representing survey data, with respondents on the x-axis and various variables on the y-axis. The color intensity indicates the values associated with each respondent-variable pair, with the color scale on the right side showing the range from low (dark purple) to high (yellow).

The variables listed include factors such as proximity to city, schools, transport, and other amenities, interior design elements, price-related factors, and neighborhood profiles, among others. The respondents are numbered sequentially on the x-axis.

The heatmap visualizes how different variables influence each respondent's preferences or responses. Brighter colors indicate higher values, suggesting stronger preferences or higher importance placed on those variables by certain respondents. Darker colors

, in contrast, indicate lower values or lesser importance. This visualization helps in identifying patterns, correlations, and significant factors across the survey data.

Recommendations

Based on the results of the survey data analysis, several key insights have been identified that can enhance the understanding of respondent characteristics and improve decision-making processes. The k-means clustering analysis revealed four distinct clusters of respondents, each with unique background variables. Visualizations such as dendrograms and heatmaps provided a clear depiction of these clusters, highlighting the differences and similarities among respondents. These insights can be leveraged to tailor marketing strategies and product offerings to better meet the needs and preferences of each identified segment. Additionally, the hierarchical clustering approach further validated the robustness of the identified clusters. It is recommended to integrate these findings into the company's strategic planning to optimize targeting and increase overall effectiveness in addressing customer requirements.

R Codes

```
# Function to auto-install and load packages
install_and_load <- function(packages) {
  for (package in packages) {
    if (!require(package, character.only = TRUE)) {
      install.packages(package, dependencies = TRUE)
    }
    library(package, character.only = TRUE)
  }
}

# List of packages to install and load
```

```
packages <- c("cluster", "FactoMineR", "factoextra", "pheatmap")

install_and_load(packages)
survey_df<-read.csv('C:\\Users\\sayas\\OneDrive\\New folder\\python projects\\Survey.csv',header=TRUE)
sur_int=survey_df[,18:46]



# Cluster analysis to characterize respondents based on background variables.
library(cluster)
library(factoextra)
show(sur_int)

# Determining Optimal Number of Clusters with Gap Statistic
fviz_nbclust(sur_int,kmeans,method = "gap_stat")

# Performing k-means Clustering
set.seed(123)
km.res<-kmeans(sur_int,4,nstart = 25)

# Visualizing k-means Clustering Results
fviz_cluster(km.res,data=sur_int,palette="jco", ggtheme = theme_minimal())

# Hierarchical Clustering (Dendrogram)
res.hc <- hclust(dist(sur_int), method = "ward.D2")
fviz_dend(res.hc,cex=0.5,k=4,palette = "jco")

#  Heatmap of Clustered Data
library(pheatmap)
pheatmap(t(sur_int),cutree_cols = 4)
```

Python Codes

```
# Import required libraries
import pandas as pd
import numpy as np
from scipy.cluster.hierarchy import dendrogram, linkage
from sklearn.cluster import KMeans
import seaborn as sns
import matplotlib.pyplot as plt
# Load the dataset
survey_df = pd.read_csv('C:\\Users\\sayas\\OneDrive\\New folder\\python projects\\S
urvey.csv')

# Select columns of interest for clustering
sur_int = survey_df.iloc[:, 17:46]
# Determine Optimal Number of Clusters using the Elbow Method
inertia = []
for k in range(1, 11):
    kmeans = KMeans(n_clusters=k, random_state=123, n_init=25)
    kmeans.fit(sur_int)
    inertia.append(kmeans.inertia_)

# Plotting the Elbow Method
plt.figure(figsize=(10, 6))
plt.plot(range(1, 11), inertia, marker='o')
plt.xlabel('Number of Clusters')
plt.ylabel('Inertia (Within-cluster Sum of Squares)')
plt.title('Elbow Method for Optimal Number of Clusters')
plt.grid(True)
plt.show()
# Visualizing k-means Clustering Results
plt.figure(figsize=(10, 6))
sns.scatterplot(x=km_res.cluster_centers_[:, 0], y=km_res.cluster_centers_[:, 1], color
='red', marker='X', s=100, label='Cluster Centers')
```

```
sns.scatterplot(x=sur_int.iloc[:, 0], y=sur_int.iloc[:, 1], hue=km_res.labels_, palette='Set1', legend='full')
plt.title('K-means Clustering Results')
plt.xlabel('Variable 1')
plt.ylabel('Variable 2')
plt.grid(True)
plt.legend()
plt.show()
# Perform Hierarchical Clustering (Ward's method)
Z = linkage(sur_int, method='ward')

# Plotting the Dendrogram
plt.figure(figsize=(12, 6))
dendrogram(Z, p=4, truncate_mode='lastp', orientation='top', leaf_font_size=10, show_contracted=True)
plt.title('Hierarchical Clustering Dendrogram')
plt.xlabel('Cluster Size')
plt.ylabel('Distance')
plt.show()
# Heatmap of clustered data
plt.figure(figsize=(10, 8))
sns.heatmap(sur_int.T, cmap='viridis', cbar=True)
plt.title('Heatmap of Survey Data')
plt.xlabel('Respondents')
plt.ylabel('Variables')
plt.show()
```

References

1. www.github.com