****

**VIRGINIA COMMONWEALTH UNIVERSITY**

**Statistical Analysis and Modelling (SCMA 632)**

**A4C: Multidimensional Scaling**

**NIHARIHA KAMALANATHAN**

**V01108259**

**Date of Submission: 08-07-2024**

**CONTENTS**

|  |  |  |
| --- | --- | --- |
| **Sl. No.** | **Title** | **Page No.** |
| **1.** | **Introduction** | **1** |
| **2.** | **Business Significance** | **1** |
| **3.** | **Objectives** | **1** |
| **4.** | **R** | **2** |
| **5.** | **Python** | **6** |
| **6.** | **Overview of Multidimensional Scaling** | **10** |

**PART C:  Apply Multidimensional Scaling and interpret the results  (**[**icecream.csv**](https://virginiacommonwealth.instructure.com/courses/106161/files/12061842?wrap=1)**)**

**Introduction**

The dataset used for this analysis is derived from a survey aimed at understanding various attributes of different ice cream brands. This dataset, denoted as "icecream.csv," contains a range of variables including price, availability, taste, flavor, consistency, and shelf life of various ice cream brands. In this analysis, we perform Multidimensional Scaling (MDS) and create a heatmap to visualize the similarities and differences between these ice cream brands. These visualizations help identify distinct clusters and relationships within the data, providing insights into consumer preferences and brand positioning.

**Business Significance**

1. **Targeted Product Development**: By identifying distinct clusters of ice cream brands based on their attributes, manufacturers can tailor their product development efforts. For example, brands with similar flavor profiles and high taste ratings can focus on enhancing other attributes like consistency or shelf life to differentiate themselves further in the market.
2. **Strategic Marketing Initiatives**: Businesses in the ice cream industry can use these insights to tailor their marketing strategies. Understanding the unique positioning of different brands allows for more precise targeting, ensuring that marketing messages resonate with the intended audience. For instance, brands that cluster together based on price and taste can be marketed to price-sensitive yet taste-conscious consumers.
3. **Retail and Distribution Planning**: Retailers can utilize the clustering results to optimize their shelf space and product placement. By grouping similar brands together, retailers can create sections that cater to specific consumer preferences, such as premium, budget, or unique flavor profiles, thereby enhancing the shopping experience.
4. **Competitive Analysis and Benchmarking**: Competitors can use the MDS and heatmap visualizations to benchmark their products against others in the market. This analysis helps in identifying areas where a brand excels or lags compared to its competitors, guiding strategic improvements and competitive positioning.
5. **Consumer Insights and Trends**: Understanding the overall structure and clustering of ice cream brands provides valuable insights into consumer preferences and trends. This information can guide not only product innovation but also promotional strategies and seasonal offerings that align with consumer expectations.

**Objectives**

1. **To Perform Multidimensional Scaling (MDS)**: To conduct MDS on the "icecream.csv" dataset to visualize the similarities and differences between ice cream brands based on their attributes.
2. **To Create a Heatmap**: To generate a heatmap of the distance matrix, illustrating the pairwise distances between ice cream brands and highlighting clusters and relationships.
3. **To Visualize the MDS Results**: To create a scatter plot of the MDS results, providing a clear representation of the positioning of different ice cream brands in a two-dimensional space.
4. **To Interpret Clustering Results**: To analyze the clusters formed in the MDS plot and heatmap, summarizing the characteristics of each cluster and offering insights into the attributes that drive similarities and differences.
5. **To Provide Business Recommendations**: To translate the findings into actionable insights for ice cream manufacturers, marketers, and retailers. This includes discussing the practical implications of the clustering results and how they can inform decision-making processes in product development, marketing strategies, and retail planning.

By achieving these objectives, the analysis aims to provide a comprehensive understanding of the different segments within the ice cream market and offer valuable insights for improving product offerings, marketing initiatives, and strategic planning.

**R LANGUAGE**

**Install and Load Necessary Libraries**

if (!requireNamespace("pheatmap", quietly = TRUE)) {

install.packages("pheatmap")

}

library(ggplot2)

library(scales)

library(dplyr)

library(tidyr)

library(readr)

library(pheatmap)

**Purpose**: This block ensures that all required libraries are installed and loaded. The pheatmap library is used for creating heatmaps, while ggplot2 is used for plotting, and dplyr, tidyr, and readr are used for data manipulation.

**Load the Dataset**

data\_filepath <- "C:/Users/nihar/OneDrive/Desktop/Bootcamp/SCMA 632/DataSet/icecream.csv"

icecream\_data <- read\_csv(data\_filepath)

head(icecream\_data)

**Purpose**: The dataset is loaded from the specified file path, and the first few rows are displayed to understand the structure of the data.

**Output**:

# A tibble: 6 × 7

Brand Price Availability Taste Flavour Consistency Shelflife

<chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>

1 Amul 4 5 4 3 4 3

2 Nandini 3 2 3 2 3 3

3 Vadilal 2 2 4 3 4 4

4 Vijaya 3 1 3 5 3 4

5 Dodla 3 3 3 4 4 3

6 Hatson 2 2 4 4 3 4

**Interpretation**: The dataset contains 7 columns: Brand, Price, Availability, Taste, Flavour, Consistency, and Shelflife. Each row represents a different ice cream brand.

**Check the Structure of the Dataset**

str(icecream\_data)

**Purpose**: The structure of the dataset is checked to confirm that the columns have the correct data types.

**Output**:

spc\_tbl\_ [10 × 7] (S3: spec\_tbl\_df/tbl\_df/tbl/data.frame)

$ Brand : chr [1:10] "Amul" "Nandini" "Vadilal" "Vijaya" ...

$ Price : num [1:10] 4 3 2 3 3 2 2 4 3 4

$ Availability: num [1:10] 5 2 2 1 3 2 3 1 4 2

$ Taste : num [1:10] 4 3 4 3 3 4 4 2 5 3

$ Flavour : num [1:10] 3 2 3 5 4 4 3 3 5 2

$ Consistency : num [1:10] 4 3 4 3 4 3 4 3 4 3

$ Shelflife : num [1:10] 3 3 4 4 3 4 4 3 4 3

- attr(\*, "spec")=

.. cols(

.. Brand = col\_character(),

.. Price = col\_double(),

.. Availability = col\_double(),

.. Taste = col\_double(),

.. Flavour = col\_double(),

.. Consistency = col\_double(),

.. Shelflife = col\_double()

.. )

- attr(\*, "problems")=<externalptr>

**Interpretation**: The structure confirms that Brand is a character column, while Price, Availability, Taste, Flavour, Consistency, and Shelflife are numeric columns.

**Select Only the Numeric Columns for MDS**

icecream\_data\_numeric <- icecream\_data %>% select(-Brand)

str(icecream\_data\_numeric)

summary(icecream\_data\_numeric)

**Purpose**: The Brand column is excluded as it is not numeric. The structure and summary of the cleaned data are then verified.

**Output**:

tibble [10 × 6] (S3: tbl\_df/tbl/data.frame)

$ Price : num [1:10] 4 3 2 3 3 2 2 4 3 4

$ Availability: num [1:10] 5 2 2 1 3 2 3 1 4 2

$ Taste : num [1:10] 4 3 4 3 3 4 4 2 5 3

$ Flavour : num [1:10] 3 2 3 5 4 4 3 3 5 2

$ Consistency : num [1:10] 4 3 4 3 4 3 4 3 4 3

$ Shelflife : num [1:10] 3 3 4 4 3 4 4 3 4 3

**Summary**:

Price Availability Taste Flavour Consistency

Min. :2.00 Min. :1.0 Min. :2.0 Min. :2.0 Min. :3.0

1st Qu.:2.25 1st Qu.:2.0 1st Qu.:3.0 1st Qu.:3.0 1st Qu.:3.0

Median :3.00 Median :2.0 Median :3.5 Median :3.0 Median :3.5

Mean :3.00 Mean :2.5 Mean :3.5 Mean :3.4 Mean :3.5

3rd Qu.:3.75 3rd Qu.:3.0 3rd Qu.:4.0 3rd Qu.:4.0 3rd Qu.:4.0

Max. :4.00 Max. :5.0 Max. :5.0 Max. :5.0 Max. :4.0

Shelflife

Min. :3.0

1st Qu.:3.0

Median :3.5

Mean :3.5

3rd Qu.:4.0

Max. :4.0

**Interpretation**: The summary statistics indicate that the values of Price, Availability, Taste, Flavour, Consistency, and Shelflife are reasonably distributed. All columns are numeric and contain values within expected ranges.

**Compute the Distance Matrix**

icecream\_dist <- dist(icecream\_data\_numeric)

**Purpose**: Compute the distance matrix, which quantifies the pairwise distances between the samples in the dataset.

**Apply Multidimensional Scaling (MDS)**

mds\_fit <- cmdscale(icecream\_dist, k = 2) # k = 2 for 2D plot

**Purpose**: Apply MDS to the distance matrix to reduce the dimensionality of the data to two dimensions for visualization purposes.

**Create a Data Frame with MDS Results**

mds\_data <- as.data.frame(mds\_fit)

colnames(mds\_data) <- c("Dim1", "Dim2")

mds\_data$Sample <- icecream\_data$Brand

**Purpose**: Create a data frame to store the MDS results and add the Brand names for labeling.

**Plot the MDS Results**

mds\_plot <- ggplot(mds\_data, aes(x = Dim1, y = Dim2, label = Sample)) +

geom\_point(size = 3) +

geom\_text(vjust = -0.5) +

labs(title = "MDS Plot of Ice Cream Dataset",

x = "Dimension 1",

y = "Dimension 2") +

theme\_minimal()

**Purpose**: Use ggplot2 to create a scatter plot of the MDS results, with each point representing a different ice cream brand.

**Output**:

A graph with black dots

Description automatically generated

**Interpretation**: The MDS plot provides a visual representation of the similarity between the ice cream brands based on their attributes. Brands that are closer together in the plot are more similar in terms of their features, while brands that are farther apart are more dissimilar.

**Create a Heatmap of the Distance Matrix**

heatmap\_data <- as.matrix(icecream\_dist)

rownames(heatmap\_data) <- icecream\_data$Brand

colnames(heatmap\_data) <- icecream\_data$Brand

heatmap\_plot <- pheatmap(heatmap\_data,

clustering\_distance\_rows = icecream\_dist,

clustering\_distance\_cols = icecream\_dist,

display\_numbers = TRUE,

fontsize\_number = 8,

main = "Heatmap of Ice Cream Distance Matrix")

**Purpose**: Convert the distance matrix to a matrix format suitable for plotting and create a heatmap using the pheatmap library.

**Output**:

A chart with different colored squares

Description automatically generated

**Interpretation**: The heatmap visualizes the pairwise distances between the ice cream brands. The color gradient indicates the magnitude of the distances, with darker colors representing smaller distances (greater similarity) and lighter colors representing larger distances (less similarity). The dendrograms on the axes show the hierarchical clustering of the brands based on the distance matrix.

**PYTHON LANGUAGE**

**1. Library Installation and Import**

**Input Code:**

# Install necessary libraries if you haven't already

# !pip install pandas matplotlib seaborn scipy

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

from sklearn.manifold import MDS

from scipy.spatial.distance import pdist, squareform

**Purpose of the Input Code:** This code ensures that all necessary libraries are installed and imported for data manipulation (pandas), data visualization (matplotlib and seaborn), distance computation (scipy), and Multidimensional Scaling (MDS) (sklearn).

**2. Load the Dataset**

**Input Code:**

# Load the dataset

data\_filepath = "C:/Users/nihar/OneDrive/Desktop/Bootcamp/SCMA 632/DataSet/icecream.csv"

icecream\_data = pd.read\_csv(data\_filepath)

**Purpose of the Input Code:** This code loads the ice cream dataset from a CSV file into a pandas DataFrame for further analysis.

**3. Display the First Few Rows of the Dataset**

**Input Code:**

# Display the first few rows of the dataset

print(icecream\_data.head())

**Output:**

Brand Price Availability Taste Flavour Consistency Shelflife

0 Amul 4 5 4 3 4 3

1 Nandini 3 2 3 2 3 3

2 Vadilal 2 2 4 3 4 4

3 Vijaya 3 1 3 5 3 4

4 Dodla 3 3 3 4 4 3

**Interpretation:** The output shows the first few rows of the dataset, providing an initial glimpse into the structure and contents of the data, which includes columns for Brand, Price, Availability, Taste, Flavour, Consistency, and Shelflife.

**4. Check the Structure of the Dataset**

**Input Code:**

# Check the structure of the dataset

print(icecream\_data.info())

**Output:**

<class 'pandas.core.frame.DataFrame'>

RangeIndex: 10 entries, 0 to 9

Data columns (total 7 columns):

# Column Non-Null Count Dtype

--- ------ -------------- -----

0 Brand 10 non-null object

1 Price 10 non-null int64

2 Availability 10 non-null int64

3 Taste 10 non-null int64

4 Flavour 10 non-null int64

5 Consistency 10 non-null int64

6 Shelflife 10 non-null int64

dtypes: int64(6), object(1)

memory usage: 692.0 bytes

**Interpretation:** The structure of the dataset is displayed, showing that there are 10 entries and 7 columns. The 'Brand' column is of type object, while the other columns are integers. This helps in understanding the data types and ensuring there are no missing values.

**5. Select Only the Numeric Columns for MDS**

**Input Code:**

# Select only the numeric columns for MDS

icecream\_data\_numeric = icecream\_data.drop(columns=['Brand'])

**Purpose of the Input Code:** This code removes the non-numeric 'Brand' column to prepare the dataset for MDS, which requires numerical data.

**6. Verify the Cleaned Data**

**Input Code:**

# Verify the cleaned data

print(icecream\_data\_numeric.info())

print(icecream\_data\_numeric.describe())

**Output:**

<class 'pandas.core.frame.DataFrame'>

RangeIndex: 10 entries, 0 to 9

Data columns (total 6 columns):

# Column Non-Null Count Dtype

--- ------ -------------- -----

0 Price 10 non-null int64

1 Availability 10 non-null int64

2 Taste 10 non-null int64

3 Flavour 10 non-null int64

4 Consistency 10 non-null int64

5 Shelflife 10 non-null int64

dtypes: int64(6)

memory usage: 612.0 bytes

None

Price Availability Taste Flavour Consistency Shelflife

count 10.000000 10.000000 10.000000 10.000000 10.000000 10.000000

mean 3.000000 2.500000 3.500000 3.400000 3.500000 3.500000

std 0.816497 1.269296 0.849837 1.074968 0.527046 0.527046

min 2.000000 1.000000 2.000000 2.000000 3.000000 3.000000

25% 2.250000 2.000000 3.000000 3.000000 3.000000 3.000000

50% 3.000000 2.000000 3.500000 3.000000 3.500000 3.500000

75% 3.750000 3.000000 4.000000 4.000000 4.000000 4.000000

max 4.000000 5.000000 5.000000 5.000000 4.000000 4.000000

**Interpretation:** The cleaned dataset contains only numeric columns. The info() method confirms that all columns are of integer type and contain no missing values. The describe() method provides descriptive statistics, giving insights into the distribution and central tendencies of the data.

**7. Compute the Distance Matrix**

**Input Code:**

# Compute the distance matrix

icecream\_dist = pdist(icecream\_data\_numeric)

**Purpose of the Input Code:** This code calculates the pairwise distances between the rows in the dataset, creating a distance matrix necessary for MDS.

**8. Apply Multidimensional Scaling (MDS)**

**Input Code:**

# Apply Multidimensional Scaling (MDS)

mds = MDS(n\_components=2, dissimilarity="precomputed", random\_state=42)

mds\_fit = mds.fit\_transform(squareform(icecream\_dist))

**Purpose of the Input Code:** This step applies MDS to reduce the dimensionality of the distance matrix to two dimensions for visualization purposes.

**9. Create a Data Frame with MDS Results**

**Input Code:**

# Create a data frame with MDS results

mds\_data = pd.DataFrame(mds\_fit, columns=['Dim1', 'Dim2'])

mds\_data['Sample'] = icecream\_data['Brand']

**Purpose of the Input Code:** This code creates a new DataFrame containing the MDS results along with the original sample labels for easy plotting.

**10. Plot the MDS Results**

**Input Code:**

# Plot the MDS results

plt.figure(figsize=(10, 6))

sns.scatterplot(data=mds\_data, x='Dim1', y='Dim2', s=100)

for i in range(mds\_data.shape[0]):

plt.text(mds\_data['Dim1'][i], mds\_data['Dim2'][i], mds\_data['Sample'][i])

plt.title("MDS Plot of Ice Cream Dataset")

plt.xlabel("Dimension 1")

plt.ylabel("Dimension 2")

plt.show()

**Output:**

A diagram of a graph

Description automatically generated with medium confidence

**Interpretation:** The MDS plot visually represents the similarities and differences between ice cream brands based on multiple attributes. Points that are closer together on the plot represent brands that are more similar to each other.

**11. Create a Heatmap of the Distance Matrix**

**Input Code:**

# Create a heatmap of the distance matrix

heatmap\_data = squareform(icecream\_dist)

heatmap\_data\_df = pd.DataFrame(heatmap\_data, index=icecream\_data['Brand'], columns=icecream\_data['Brand'])

**Purpose of the Input Code:** This code converts the distance matrix into a DataFrame suitable for plotting a heatmap.

**12. Plot the Heatmap**

**Input Code:**

# Plot the heatmap

plt.figure(figsize=(12, 10))

sns.heatmap(heatmap\_data\_df, annot=True, fmt=".2f", cmap="YlGnBu")

plt.title("Heatmap of Ice Cream Distance Matrix")

plt.show()

**Output:**

A screenshot of a graph

Description automatically generated

**Interpretation:** The heatmap provides a visual representation of the pairwise distances between different ice cream brands. Darker colors indicate larger distances, while lighter colors indicate smaller distances. This helps in identifying clusters or groups of similar brands.

**Overview of Multidimensional Scaling (MDS)**

**Meaning**

Multidimensional Scaling (MDS) is a statistical technique used to analyze and visualize the similarities or dissimilarities (distances) between a set of objects or data points. The goal of MDS is to place each object in a low-dimensional space (typically two or three dimensions) in such a way that the distances between points in this space reflect the given pairwise dissimilarities. This technique is widely used in fields such as psychology, marketing, and bioinformatics to explore patterns and relationships in complex datasets.

**Advantages**

1. **Visualization of High-Dimensional Data**: MDS provides a way to visualize high-dimensional data in a more interpretable, low-dimensional form. This can help in identifying patterns and clusters that are not immediately apparent in the original high-dimensional space.
2. **Insight into Relationships**: By plotting the data points in a reduced space, MDS helps reveal the underlying structure and relationships between the objects. This can be particularly useful in understanding the relative positioning and similarities among items.
3. **Flexibility with Different Distance Measures**: MDS can work with a variety of distance or dissimilarity measures, making it versatile for different types of data and applications. Whether the distances are Euclidean, Manhattan, or derived from other metrics, MDS can accommodate them.
4. **Uncovering Hidden Dimensions**: The technique helps in uncovering hidden dimensions or factors that explain the distances or dissimilarities between objects, providing deeper insights into the data.
5. **Application Across Fields**: MDS is applicable in numerous fields including market research, social sciences, genetics, ecology, and more, making it a valuable tool for diverse types of analyses.

**Real-Life Examples**

1. **Market Research**: In market research, MDS is used to understand consumer preferences and perceptions. For example, a company might use MDS to analyze how consumers perceive different brands of a product based on attributes like quality, price, and taste. The resulting map can help identify brand positioning and competition.
2. **Psychology**: Psychologists use MDS to study similarities and differences in psychological traits or behaviors. For instance, MDS can help in visualizing the relationships between different psychological tests or personality traits, aiding in the development of psychological theories and assessments.
3. **Bioinformatics**: In bioinformatics, MDS is used to visualize genetic or protein sequence similarities. By mapping genetic data into a low-dimensional space, researchers can identify patterns and evolutionary relationships among different species or strains.
4. **Sociology**: Sociologists employ MDS to analyze social networks and relationships. MDS can help visualize the social distance between individuals or groups, providing insights into social structures and interactions.
5. **Ecology**: Ecologists use MDS to study the similarities between different ecological sites based on species composition. This can help in understanding biodiversity patterns and the ecological relationships between different environments.

By transforming complex, high-dimensional data into an easily interpretable form, MDS serves as a powerful tool for uncovering insights and making informed decisions in a variety of disciplines.

Top of Form

Bottom of Form