VIRGINIA COMMONWEALTH UNIVERSITY

Statistical Analysis & Modelling

A5 – Perceptual mapping using NSSO dataset

State: **Mizoram**

Using Python and R

Submitted by

Chandhini Kerachan Muraleedharan

V01107497

Date of Submission: 15/07/2024

Table of Contents

1. Introduction

1.1. About the Dataset

The NSSO-Consumption dataset is an extensive compilation of consumption data for all Indian states and union territories.

It provides detailed insights into the consumption patterns of various commodities, including grains, oils, fruits, vegetables, and more.

Additionally, the dataset contains basic demographic information for each sample, allowing for a comprehensive analysis of consumption trends across different regions of India.

All data in the dataset, including the states and union territories, is presented in numerical format, making it readily accessible for statistical analysis.

1.2. Objective

To Visualize and do Perceptual Mapping to show the total consumption across different districts

To Visualize consumption per district with district names

To plot any variable of our choice in Mizoram

1.3. Business Significance

Drawing a histogram of district-wise consumption data helps businesses visualize and understand the consumption patterns at a granular level.

This information is crucial for identifying high-demand areas and tailoring products or services to meet local preferences.

By depicting consumption on a state map and showing consumption in each district, companies can easily identify regions with higher or lower consumption rates.

This allows for strategic market penetration efforts, focusing on areas with untapped potential. Understanding district-wise consumption patterns enables businesses to allocate resources more

2. Results

2.1 Output and Inference

Missing Value:

```
In [6]: missing_values = Mizoram_data.isna().sum()
        print("Missing values in each column:")
        print(missing_values)
```

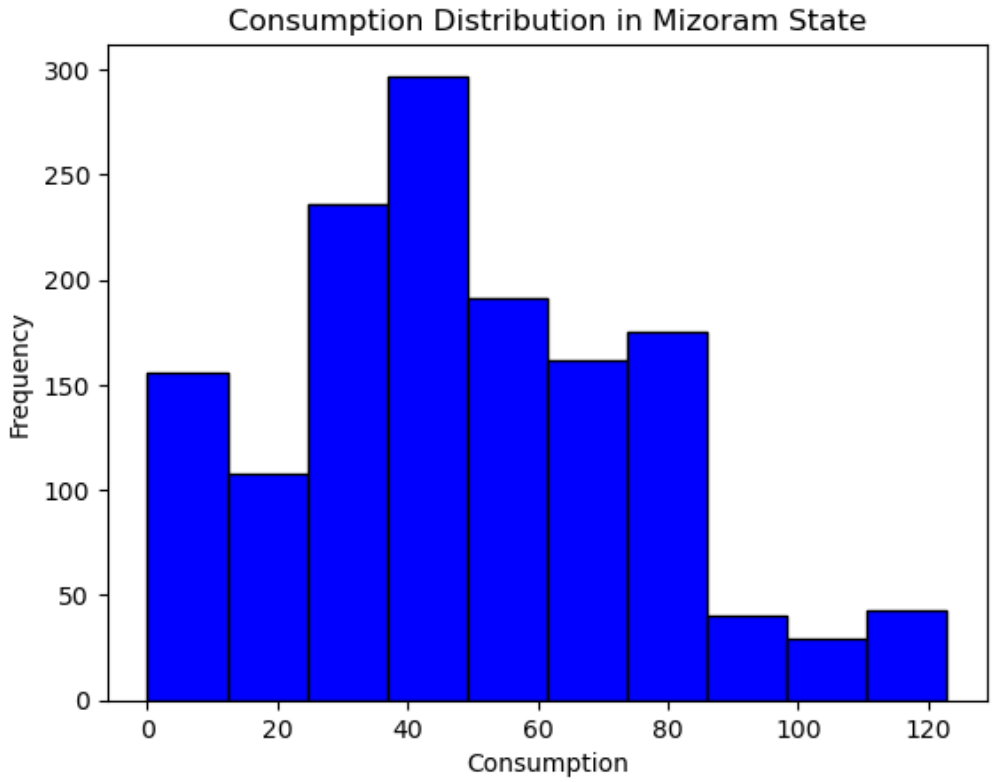
```
Missing values in each column:
slno              0
grp               0
Round_Centre      0
FSU_number        0
Round             0
                 ..
foodtotal_q       0
state_1           0
Region            0
fruits_df_tt_v    0
fv_tot            0
Length: 384, dtype: int64
```

Inference:

There are no missing values in any of the columns of the Mizoram data DataFrame. This is indicated by the fact that all columns have a missing value count of 0.

2.3. Histogram indicating the consumption District wise

In [39]:
```
plt.hist(MIZ['total_consumption'], bins=10, color='blue', edgecolor='black')
plt.xlabel("Consumption")
plt.ylabel("Frequency")
plt.title("Consumption Distribution in Mizoram State")
plt.show()
```



Inference:

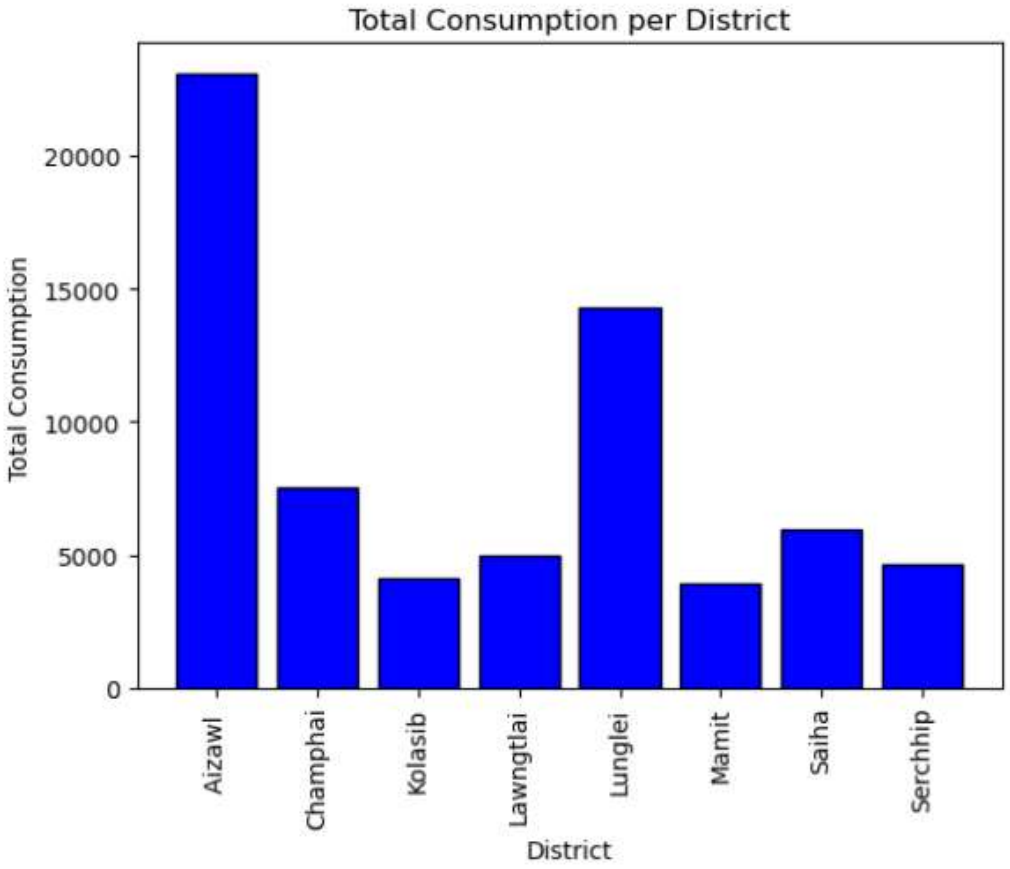
X-Axis (Consumption): Represents the range of consumption values.
Y-Axis (Frequency): Represents the frequency of occurrences for each bin range of consumption values.

Most of the consumption values are clustered between 0 and 80 units.
The distribution shows a peak in the 30-40 units range.

As consumption values increase beyond 40 units, the frequency decreases.

This right-skewed distribution indicates that while most data points have lower consumption values, there are a few instances of significantly higher consumption.

In [31]:
```
plt.bar(MIZ_consumption['District'], MIZ_consumption['total_consumption'], co
plt.xlabel("District")
plt.ylabel("Total Consumption")
plt.title("Total Consumption per District")
plt.xticks(rotation=90)  # Rotate district names for better visibility
plt.show()
```



Inference:

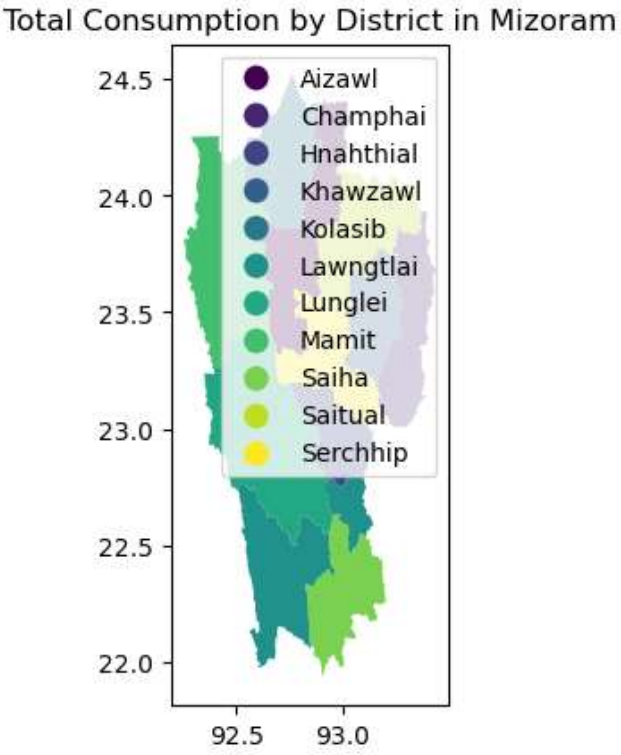
X-Axis (District): Represents the different districts in Mizoram State.
Y-Axis (Total Consumption): This represents the total consumption values for each district.

Aizawl: Has the highest total consumption, significantly higher than other districts, exceeding 20,000 units.
Champhai: Has a total consumption of around 6,000 units.
Lunglei: Has a high total consumption as well, just below 10,000 units.
Saiha: Shows a total consumption of approximately 5,000 units.
Other districts such as Kolasib, Lawngtlai, Mamit, and Serchhip have lower total consumption values compared to Aizawl, Champhai, Lunglei, and Saiha.

Aizawl dominates in terms of total consumption, suggesting it might be a more populous or resource-intensive district.
Lunglei and Champhai also show relatively high total consumption.
The rest of the districts have lower total consumption values, indicating a less intense use of resources or smaller populations.
The bar chart effectively highlights the disparity in consumption across different districts in Mizoram State.

2.4. Mizoram state map showing consumption in each district

In [49]:
```
fig, ax = plt.subplots(1, 1)
data_map.plot(column='total_consumption', cmap='viridis', legend=True, ax=ax)
ax.set_title('Total Consumption by District in Mizoram')
plt.show()
```



Inference:

Map of Mizoram with the total consumption by district. The districts are color-coded according to their total consumption, with the highest consumption in Aizawl and the lowest in Serchhip. The map also shows the latitude and longitude of each district.

-

The total consumption in Mizoram is highest in Aizawl, followed by Champhai and Khawzawl. The lowest consumption is in Serchhip, followed by Saitual and Lunglei.

The map shows that there is a significant variation in the total consumption across the districts of Mizoram.

This could be due to several factors, such as the size of the district, the population density, the level of economic development, and the availability of resources.

3. Recommendations

3.1. Business Implications:

Consumption Disparity: There's a significant difference in consumption patterns across Mizoram's districts, with Aizawl at the top and Serhiy at the bottom.

Geographical Influence: Consumption seems to correlate with geographical location, with higher consumption in the northern districts.

Potential Factors: Factors like population density, economic development, and infrastructure could be influencing consumption patterns.

3.2. Business Recommendations:

Market Segmentation: Businesses can identify high-potential markets (like Aizawl) and tailor their products or services accordingly.

Supply Chain Optimization: Understanding consumption patterns can help optimize supply chain management, reducing costs and improving efficiency.

Infrastructure Development: Businesses can collaborate with government and other stakeholders to advocate for infrastructure development in low-consumption areas to stimulate economic growth.

Product Adaptation: Businesses might need to adapt their product offerings or marketing strategies to cater to different consumer preferences in various districts.

Risk Assessment: Businesses should consider the potential risks associated with operating in areas with lower consumption levels.

4. Codes

4.1. Python Jupyter Notebook codes

```
import pandas as pd
import numpy as np
from scipy import stats
import matplotlib.pyplot as plt
import seaborn as sns
import geopandas as gpd
data = pd.read_csv("C:\\Users\\Chand\\Downloads\\A5\\NSSO68.csv", low_memory=False)
display(data)
Mizoram_data = data[data['state_1'] == 'MIZ']
missing_values = Mizoram_data.isna().sum()
```

```
print("Missing values in each column:")
print(missing_values)
MIZ = Mizoram_data[['state_1', 'District', 'Region', 'Sector', 'State_Region', 'Meals_At_Home',
     'ricepds_v', 'Wheatpds_q', 'chicken_q', 'pulsep_q', 'wheatos_q', 'No_of_Meals_per_day']]
def impute_with_mean(column):
    if column.hasnans:
        column.fillna(column.mean(), inplace=True)
    return column
MIZ['Meals_At_Home'] = impute_with_mean(MIZ['Meals_At_Home'])
def remove_outliers(df, column_name):
    Q1 = df[column_name].quantile(0.25)
    Q3 = df[column_name].quantile(0.75)
    IQR = Q3 - Q1
    lower_threshold = Q1 - (1.5 * IQR)
    upper_threshold = Q3 + (1.5 * IQR)
    df = df[(df[column_name] >= lower_threshold) & (df[column_name] <= upper_threshold)]
    return df

outlier_columns = ['ricepds_v', 'chicken_q']
for col in outlier_columns:
    MIZ = remove_outliers(MIZ, col)
MIZ['total_consumption'] = MIZ[['ricepds_v', 'Wheatpds_q', 'chicken_q', 'pulsep_q',
     'wheatos_q']].sum(axis=1)
MIZ['total_consumption'] = MIZ[['ricepds_v', 'Wheatpds_q', 'chicken_q', 'pulsep_q',
     'wheatos_q']].sum(axis=1)
def summarize_consumption(group_col):
    summary = MIZ.groupby(group_col)['total_consumption'].sum().reset_index()
    summary.sort_values(by='total_consumption', ascending=False, inplace=True)
    return summary
district_summary = summarize_consumption('District')
region_summary = summarize_consumption('Region')
print("Top Consuming Districts:")
print(district_summary.head(4))
print("Region Consumption Summary:")
print(region_summary)
district = {'1': 'Mamit',
    '2': 'Kolasib',
    '3': 'Aizawl ',
    '4': 'Champhai',
    '5': 'Serchhip',
    '6': 'Lunglei',
    '7': 'Lawngtlai',
    '8': 'Saiha',
}

sector = {
    '2': 'URBAN',
    '1': 'RURAL'
}
MIZ['District'] = MIZ['District'].astype(str)
```

```
MIZ['Sector'] = MIZ['Sector'].astype(str)

MIZ['District'] = MIZ['District'].map(district).fillna(MIZ['District'])
MIZ['Sector'] = MIZ['Sector'].map(sector).fillna(MIZ['Sector'])
print(MIZ.head())
plt.hist(MIZ['total_consumption'], bins=10, color='blue', edgecolor='black')
plt.xlabel("Consumption")
plt.ylabel("Frequency")
plt.title("Consumption Distribution in Mizoram State")
plt.show()
MIZ_consumption = MIZ.groupby('District')['total_consumption'].sum().reset_index()
print(MIZ_consumption.head())
plt.bar(MIZ_consumption['District'],  MIZ_consumption['total_consumption'],  color='blue',
    edgecolor='black')
plt.xlabel("District")
plt.ylabel("Total Consumption")
plt.title("Total Consumption per District")
plt.xticks(rotation=90)  # Rotate district names for better visibility
plt.show()
data_map                                                                                =
    gpd.read_file("C:\\Users\\Chand\\Downloads\\MIZORAM_DISTRICTS.geojson")
print(data_map.columns)
print(MIZ_consumption.columns)
data_map['District'] = MIZ_consumption['District']
data_map_data = data_map.merge(MIZ_consumption, left_on='dtname', right_on='District')
print(data_map.columns)
import geopandas as gpd
import pandas as pd
import matplotlib.pyplot as plt
data_map                                                                                =
    gpd.read_file("C:\\Users\\Chand\\Downloads\\MIZORAM_DISTRICTS.geojson")
data_map = data_map.rename(columns={'dtname': 'District'})
display(data_map.rename)
MIZ_consumption      =      pd.read_csv("C:\\Users\\Chand\\Downloads\\A5\\NSSO68.csv",
    low_memory=False)
MIZ_consumption = MIZ.groupby('District')['total_consumption'].sum().reset_index()
print(MIZ_consumption.head())
data_map                                                                                =
    gpd.read_file("C:\\Users\\Chand\\Downloads\\MIZORAM_DISTRICTS.geojson")
data_map = data_map.rename(columns={'dtname': 'total_consumption'})
fig, ax = plt.subplots(1, 1)
data_map.plot(column='total_consumption', cmap='viridis', legend=True, ax=ax)
ax.set_title('Total Consumption by District in Mizoram')
plt.show()
```

4.2. R codes

```
# Set the working directory and verify it
setwd('C:\\Users\\Chand\\Downloads\\A5')
getwd()
install.packages("sf")

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

# Load required libraries
libraries <- c("dplyr", "readr", "readxl", "tidyr", "ggplot2", "BSDA")
lapply(libraries, install_and_load)

# Reading the file into R
data <- read.csv("NSSO68.csv")

# Filtering for MIZ
df <- data %>%
  filter(state_1 == "MIZ")

# Display dataset info
cat("Dataset Information:\n")
print(names(df))
print(head(df))
print(dim(df))

# Finding missing values
missing_info <- colSums(is.na(df))
cat("Missing Values Information:\n")
print(missing_info)

# Subsetting the data
MIZnew <- df %>%
  select(state_1, District, Region, Sector, State_Region, Meals_At_Home, ricepds_v,
      Wheatpds_q, chicken_q, pulsep_q, wheatos_q, No_of_Meals_per_day)

# Impute missing values with mean for specific columns
impute_with_mean <- function(column) {
  if (any(is.na(column))) {
    column[is.na(column)] <- mean(column, na.rm = TRUE)
  }
```

```
  return(column)
}
MIZnew$Meals_At_Home <- impute_with_mean(MIZnew$Meals_At_Home)

# Finding outliers and removing them
remove_outliers <- function(df, column_name) {
  Q1 <- quantile(df[[column_name]], 0.25)
  Q3 <- quantile(df[[column_name]], 0.75)
  IQR <- Q3 - Q1
  lower_threshold <- Q1 - (1.5 * IQR)
  upper_threshold <- Q3 + (1.5 * IQR)
  df <- subset(df, df[[column_name]] >= lower_threshold & df[[column_name]] <=
     upper_threshold)
  return(df)
}
outlier_columns <- c("ricepds_v", "chicken_q")
for (col in outlier_columns) {
  MIZnew <- remove_outliers(MIZnew, col)
}

# Summarize consumption
MIZnew$total_consumption <- rowSums(MIZnew[, c("ricepds_v", "Wheatpds_q",
     "chicken_q", "pulsep_q", "wheatos_q")], na.rm = TRUE)

# Summarize and display top consuming districts and regions
summarize_consumption <- function(group_col) {
  summary <- MIZnew %>%
    group_by(across(all_of(group_col))) %>%
    summarise(total = sum(total_consumption)) %>%
    arrange(desc(total))
  return(summary)
}
district_summary <- summarize_consumption("District")
region_summary <- summarize_consumption("Region")

cat("Top Consuming Districts:\n")
print(head(district_summary, 4))
cat("Region Consumption Summary:\n")
print(region_summary)

# Rename districts and sectors
district_mapping <- c("1" = "Mamit","2" = "Kolasib","3" = "Aizawl","4" = "Champhai","5" =
     "Serchhip","6" = "Lunglei","7" = "Lawngtlai","8" = "Saiha")
sector_mapping <- c("2" = "URBAN", "1" = "RURAL")

MIZnew$District <- as.character(MIZnew$District)
MIZnew$Sector <- as.character(MIZnew$Sector)
MIZnew$District <- ifelse(MIZnew$District %in% names(district_mapping),
     district_mapping[MIZnew$District], MIZnew$District)
```

```
MIZnew$Sector <- ifelse(MIZnew$Sector %in% names(sector_mapping),
    sector_mapping[MIZnew$Sector], MIZnew$Sector)

View(MIZnew)

hist(MIZnew$total_consumption, breaks = 10, col = 'blue', border = 'black',
   xlab = "Consumption", ylab = "Frequency", main = "Consumption Distribution in Mizoram
    State")

MIZ_consumption <- aggregate(total_consumption ~ District, data = MIZnew, sum)
View(MIZ_consumption)
??barplot
barplot(MIZ_consumption$total_consumption,
       names.arg = MIZ_consumption$District,
       las = 2, # Makes the district names vertical
       col = 'blue',
       border = 'black',
       xlab = "District",
       ylab = "Total Consumption",
       main = "Total Consumption per District",
       cex.names = 0.7) # Adjust the size of district names if needed

# b) Plot {'any variable of your choice'} on the Karnataka state map using NSSO68.csv data

library(ggplot2)
library(sf) # mapping
library(dplyr)
Sys.setenv("SHAPE_RESTORE_SHX" = "YES")

data_map <- st_read("C:\\Users\\Chand\\Downloads\\MIZORAM_DISTRICTS.geojson")
View(data_map)

data_map <- data_map %>%
  rename(District = dtname)
colnames(data_map)
data_map_data <- merge(MIZ_consumption,data_map,by = "District")
View(data_map_data)
ggplot(data_map_data) +
  geom_sf(aes(fill =total_consumption, geometry = geometry)) +
  scale_fill_gradient(low = "yellow", high = "red") +
  ggtitle("Total Consumption_by_District")

ggplot(data_map_data) +
  geom_sf(aes(fill = total_consumption, geometry = geometry)) +
  scale_fill_gradient(low = "yellow", high = "red") +
  ggtitle("Total Consumption by District") +
  geom_sf_text(aes(label = District, geometry = geometry), size = 3, color = "black"
```