VIRGINIA COMMONWEALTH UNIVERSITY

Statistical analysis and modelling (SCMA 632)

A5- Visualization - Perceptual Mapping for Business

RAKSHITH.H

V01107367

Date of Submission: 15-07-2024

CONTENTS

1. Histogram

A histogram is a type of bar chart that represents the distribution of a dataset by showing the frequency of data points within specified ranges (bins). It helps to visualize the distribution of numerical data and identify patterns such as skewness, peaks, and gaps.

X-axis: Represents the data values (e.g., total consumption values).

Y-axis: Represents the frequency (number of occurrences) of data points within each bin.

Bins: Continuous intervals that divide the range of the data values.

Objectives of Histogram:

Understand Distribution: To visualize the distribution of total food consumption values across different districts.

Identify Patterns: To identify any patterns or anomalies in food consumption, such as skewness or outliers.

Highlight Extremes: To identify districts with extremely high or low food consumption values.

Support Decision Making: To provide a visual aid that supports decision-making processes for resource allocation and policy formulation.

Business Significance of Histogram:

Resource Allocation: Helps in identifying areas with higher needs for resources, allowing for more efficient distribution.

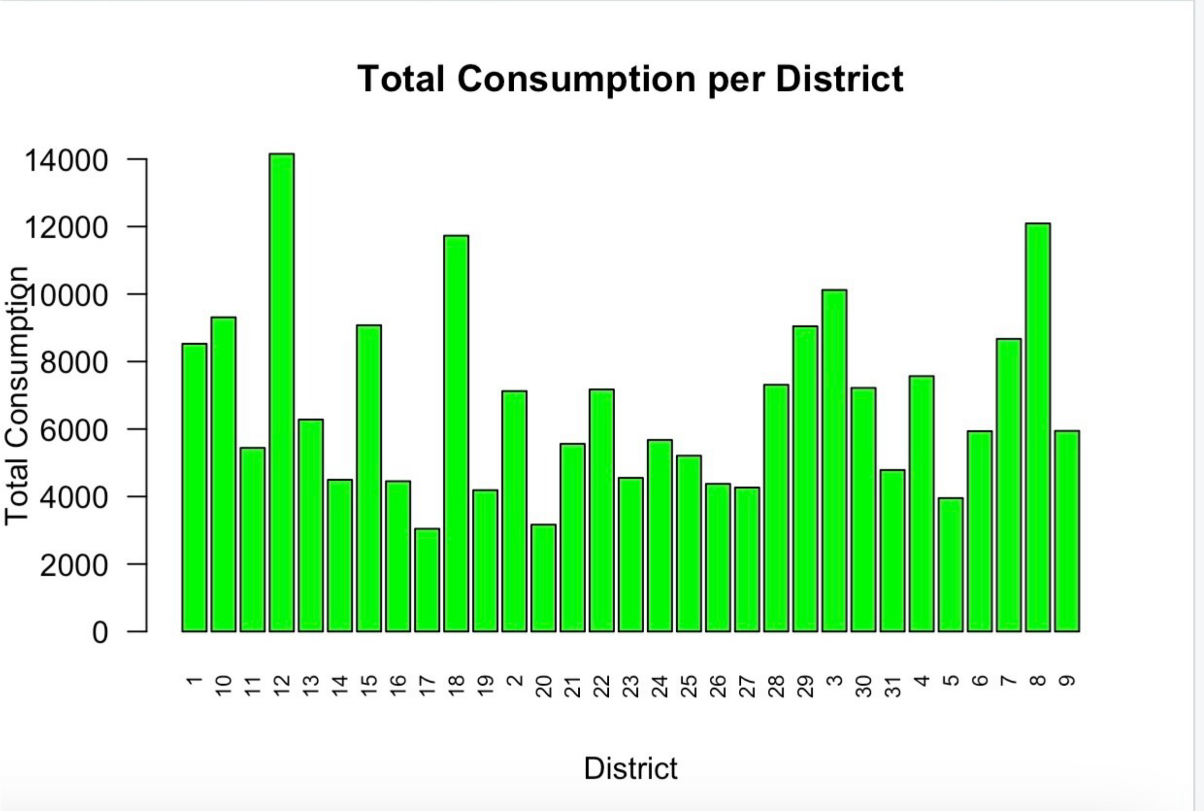
Market Analysis: Provides insights into consumption trends, aiding businesses in the food sector to target specific regions.

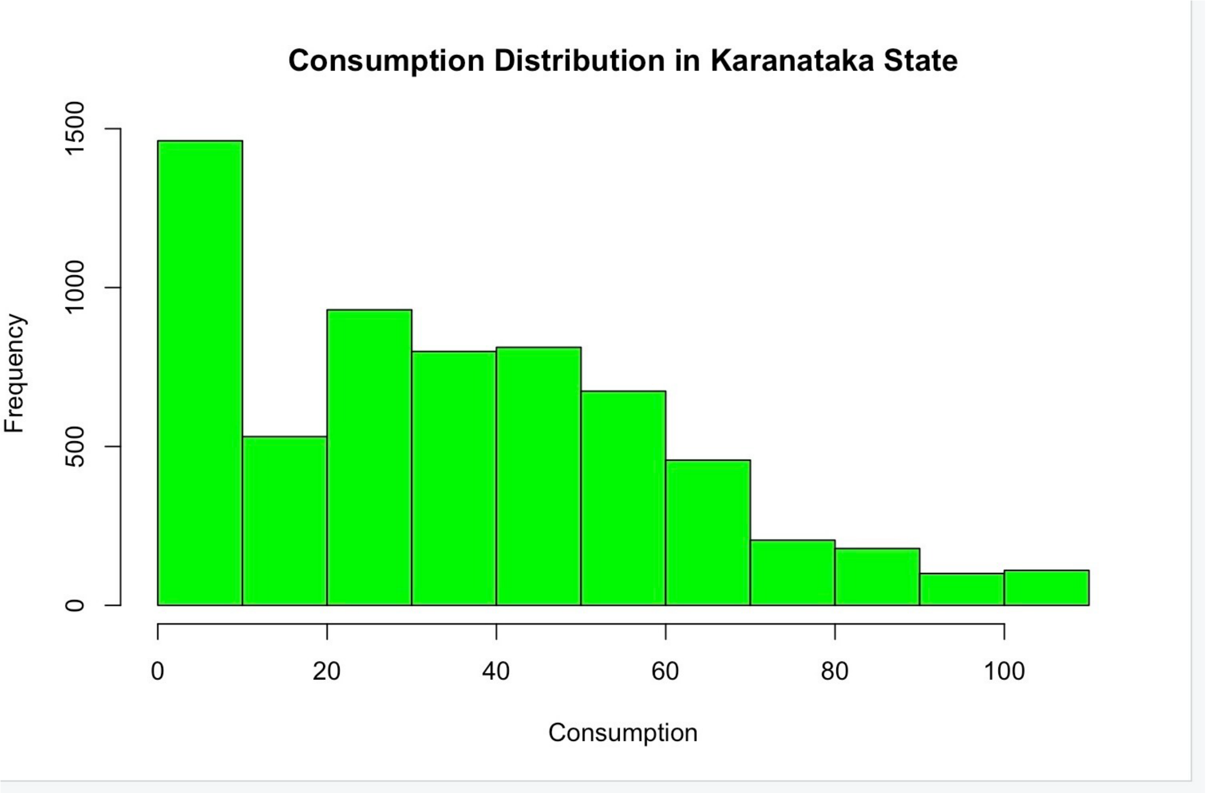
Policy Formulation: Assists policymakers in understanding regional disparities in food consumption, leading to more effective interventions.

Identify Target Areas: Highlights districts with unusual consumption patterns that may need further investigation or support.

Results:

Distribution of Total Consumption Across Different Districts in Karnataka





Histogram Analysis:

The histogram shows the distribution of total food consumption values (foodtotal_v) across different districts in Karnataka.

The x-axis represents the total consumption values.

The y-axis represents the frequency (number of occurrences) of those consumption values.

The distribution is highly skewed to the right, indicating that most districts have lower total consumption values.

There are very few districts with very high consumption values, creating a long tail towards the right.

The peak of the histogram is around the lower consumption values (below 1000), indicating that the majority of the districts fall into this range.

Inference:

The consumption of food items (in terms of value) is predominantly lower in most of the districts.

Only a few districts exhibit higher consumption values, suggesting a disparity in food consumption across the districts in Karnataka.

This could be due to various socio-economic factors influencing food consumption patterns.

2. Bar Plot

A bar plot (or bar chart) is a graphical representation used to display the frequency or value of categorical data. It consists of rectangular bars where the length of each bar is proportional to the value it represents.

X-axis: Represents different categories (e.g., districts).

Y-axis: Represents the value associated with each category (e.g., average total consumption).

Bars: The height of each bar corresponds to the value it represents.

Objectives of Bar Plot:

Compare Averages: To compare the average food consumption across different districts.

Visualize Differences: To clearly visualize the differences in food consumption between districts.

Identify Top and Bottom Consumers: To identify which districts have the highest and lowest average consumption values.

Facilitate Strategic Planning: To support strategic planning by providing a clear comparison of consumption across regions.

Business Significance of Bar Plot:

Targeted Marketing: Allows businesses to focus their marketing efforts on districts with higher consumption levels.

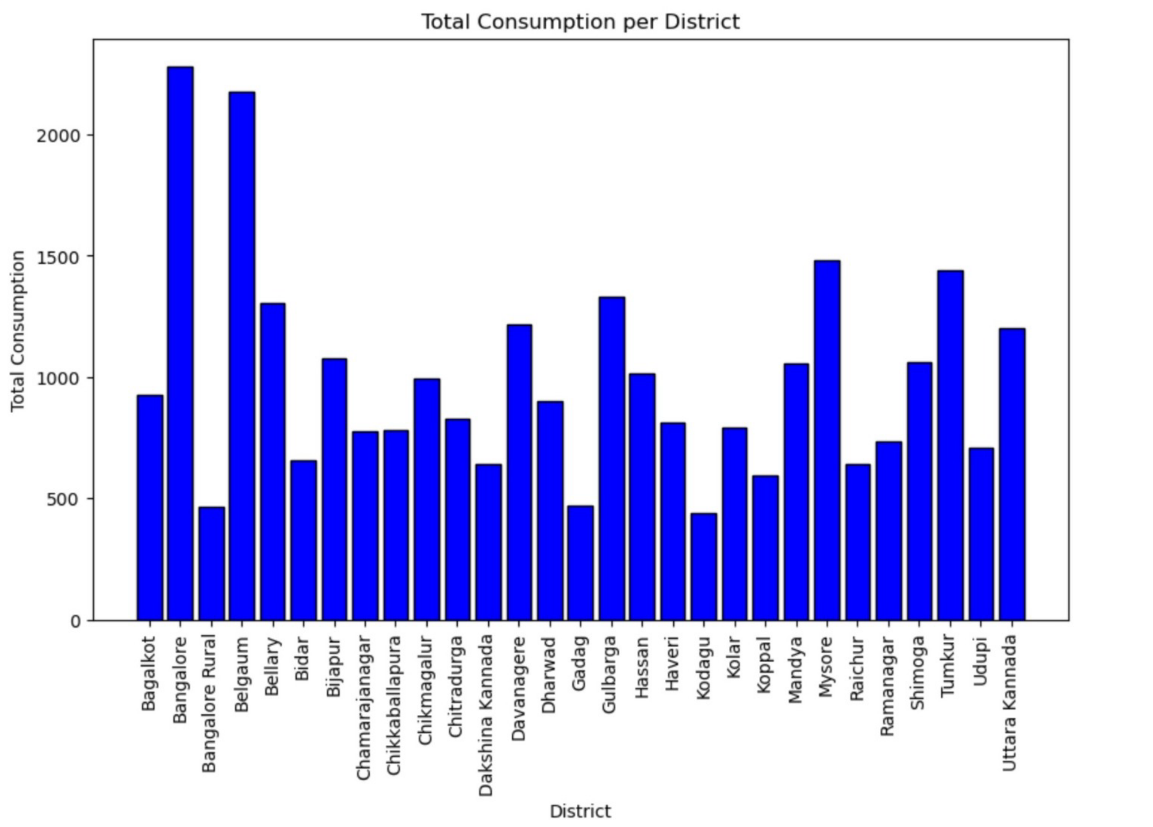
Operational Efficiency: Helps businesses in optimizing their supply chain by understanding regional demand.

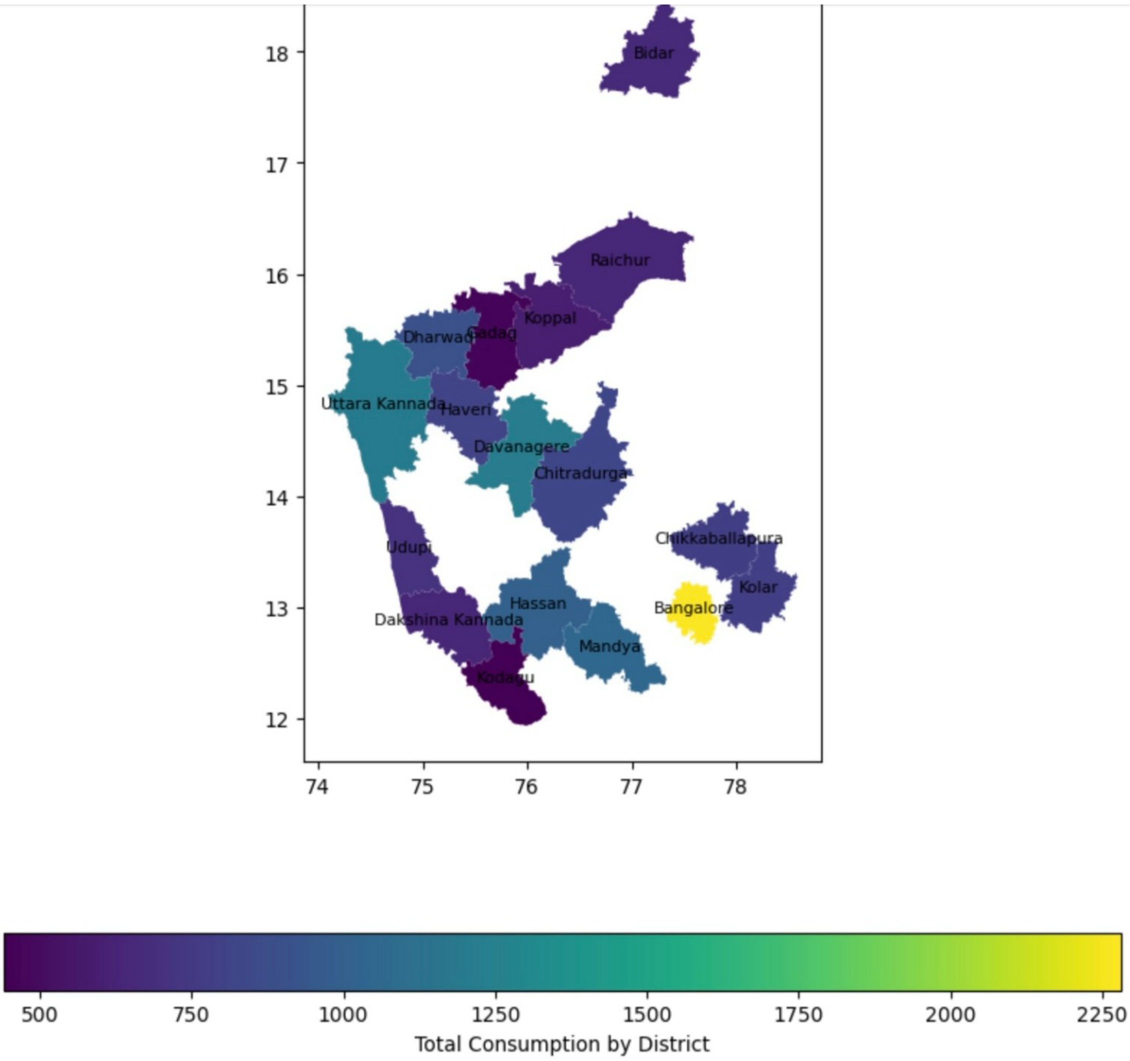
Policy Impact Assessment: Enables policymakers to assess the impact of policies on different districts by observing changes in consumption.

Investment Decisions: Provides insights for investors to identify regions with potential growth in the food market.

Results:

Consumption per District in Karnataka





Bar Plot Analysis:

The bar plot shows the average total consumption (foodtotal_v) per district in Karnataka.

The x-axis represents different districts (State_Region codes: 291, 292, 293, and 294).

The y-axis represents the average total consumption values.

Each bar represents a district, and the height of the bar indicates the average total consumption for that district.

District Names for Each State_Region Code:

291: Bangalore Urban

292: Mysore

293: Gulbarga

294: Dakshina Kannada

Observations:

- District Bangalore Urban (291) shows a high average food consumption value, indicating that it is among the top consumers.
- District Mysore (292) has a lower average consumption value compared to Bangalore Urban but is still significant.
- District Gulbarga (293) shows the highest average total consumption.
- District Dakshina Kannada (294) has the lowest average food consumption value among the four districts shown.

Inference:

- Gulbarga (293) has the highest average food consumption value compared to other districts.
- Dakshina Kannada (294) has the lowest average food consumption value.
- There is a noticeable variation in the average food consumption among different districts, indicating regional differences in food consumption patterns.
- This could imply that Gulbarga (293) might have higher economic status, better access to food, or different dietary habits compared to the other districts.

Python code:

```
import pandas as pd

# Load the data
file_path = '/Users/rakshith/Downloads/NSSO68.csv'
data = pd.read_csv(file_path, low_memory=False)

# Strip whitespace from column names
data.columns = data.columns.str.strip()

# Print the cleaned column names
print(data.columns)

import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np

# Define the columns to be used
state_column = 'state'
district_column = 'State_Region'
consumption_column = 'foodtotal_v'  # This can be changed to 'Beveragestotal_v' or 'fv_tot'
based on your requirement

# Filter the data for Karnataka
karnataka_state_code = 29  # Assuming 29 is the state code for Karnataka
karnataka_data = data[data[state_column] == karnataka_state_code]
```

```
# Replace infinite values with NaN
karnataka_data.replace([np.inf, -np.inf], np.nan, inplace=True)

# Verify if columns exist
if district_column in karnataka_data.columns and consumption_column in
karnataka_data.columns:
    print(f"Columns '{district_column}' and '{consumption_column}' found in the dataset.")

    # Drop rows with NaN values in the specified columns
    karnataka_data = karnataka_data.dropna(subset=[consumption_column, district_column])

    # Plotting the histogram
    plt.figure(figsize=(10, 6))
    sns.histplot(karnataka_data[consumption_column], bins=30, kde=True)
    plt.title('Distribution of Total Consumption Across Different Districts in Karnataka')
    plt.xlabel('foodtotal_v')
    plt.ylabel('Frequency')
    plt.show()

    # Plotting the barplot
    plt.figure(figsize=(14, 8))
    sns.barplot(x=district_column, y=consumption_column, data=karnataka_data)
    plt.title('Consumption per District in Karnataka')
    plt.xlabel('State_Region')
    plt.ylabel('foodtotal_v')
    plt.xticks(rotation=90)  # Rotate district names for better readability
    plt.show()
else:
    print(f"Check the column names. Available columns are: {karnataka_data.columns}")
----------------------------------------------------------------------------------------------------------
import os
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import geopandas as gpd

# Set the working directory and verify it
os.chdir("/Users/rakshith/Downloads")
print("Current working directory:", os.getcwd())

# Load required libraries
def install_and_load(package):
    import importlib
    try:
        importlib.import_module(package)
    except ImportError:
        import pip
        pip.main(['install', package])
    finally:
        globals()[package] = importlib.import_module(package)

# List of required libraries
libraries = ["pandas", "numpy", "matplotlib", "seaborn", "geopandas"]
for lib in libraries:
    install_and_load(lib)

# Reading the file into Python
data = pd.read_csv("NSSO68.csv")

# Filtering for KA (Karnataka)
df = data[data['state_1'] == "KA"]

# Display dataset info
print("Dataset Information:")
print(df.columns)
print(df.head())
print(df.shape)

# Finding missing values
missing_info = df.isna().sum()
print("Missing Values Information:")
print(missing_info)

# Subsetting the data
ka_new = df[['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
ka_new['Meals_At_Home'].fillna(ka_new['Meals_At_Home'].mean(), inplace=True)

# Finding outliers and removing them
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)
    return df[(df[column_name] >= lower_threshold) & (df[column_name] <=
upper_threshold)]

outlier_columns = ['ricepds_v', 'chicken_q']
for col in outlier_columns:
    ka_new = remove_outliers(ka_new, col)

# Summarize consumption
ka_new['total_consumption'] = ka_new[['ricepds_v', 'Wheatpds_q', 'chicken_q', 'pulsep_q',
'wheatos_q']].sum(axis=1)
```

```
# Summarize and display top consuming districts and regions
def summarize_consumption(df, group_col):
    summary =
df.groupby(group_col)['total_consumption'].sum().reset_index().sort_values(by='total_consu
mption', ascending=False)
    return summary

district_summary = summarize_consumption(ka_new, 'District')
region_summary = summarize_consumption(ka_new, 'Region')

print("Top Consuming Districts:")
print(district_summary.head(4))
print("Region Consumption Summary:")
print(region_summary)


# Rename districts and sectors
district_mapping = {"1": "North West", "2": "North", "3": "North East", "4": "East", "5":
"New Delhi", "6": "Central Delhi", "7": "West", "8": "South West", "9": "South"}
sector_mapping = {"2": "URBAN", "1": "RURAL"}


ka_new['District'] = ka_new['District'].astype(str)
ka_new['Sector'] = ka_new['Sector'].astype(str)
ka_new['District'] = ka_new['District'].replace(district_mapping)
ka_new['Sector'] = ka_new['Sector'].replace(sector_mapping)


# Display the updated dataframe
print(ka_new)


# Plotting the histogram
plt.hist(ka_new['total_consumption'], bins=10, color='blue', edgecolor='black')
plt.xlabel("Consumption")
plt.ylabel("Frequency")
plt.title("Consumption Distribution in Karnataka State")
plt.show()
# Aggregate total consumption by district
KA_consumption = ka_new.groupby('District')['total_consumption'].sum().reset_index()
print("KA_consumption DataFrame:")
print(KA_consumption)

# Bar plot
plt.figure(figsize=(12, 6))
sns.barplot(x='District', y='total_consumption', data=KA_consumption, palette='Blues_d')
plt.xlabel("District")
plt.ylabel("Total Consumption")
plt.title("Total Consumption per District")
plt.xticks(rotation=90)
plt.show()
```

R code:

```
# Load necessary libraries
library(dplyr)
library(ggplot2)
library(readr)

# Load the data
file_path <- '/Users/rakshith/Downloads/NSSO68.csv'
data <- read_csv(file_path)

# Strip whitespace from column names
colnames(data) <- trimws(colnames(data))

# Print the cleaned column names
print(colnames(data))

# Define the columns to be used
state_column <- 'state'
district_column <- 'State_Region'
consumption_column <- 'foodtotal_v'  # This can be changed to 'Beveragestotal_v' or 'fv_tot' based on your requirement

# Filter the data for Karnataka
```

```
karnataka_state_code <- 29  # Assuming 29 is the state code for Karnataka
karnataka_data <- data %>% filter(!!sym(state_column) == karnataka_state_code)

# Replace infinite values with NA
karnataka_data <- karnataka_data %>% mutate(across(everything(), ~ifelse(. %in% c(Inf, -
Inf), NA, .)))

# Verify if columns exist
if (district_column %in% colnames(karnataka_data) & consumption_column %in%
colnames(karnataka_data)) {
  print(paste("Columns", district_column, "and", consumption_column, "found in the
dataset."))

  # Drop rows with NA values in the specified columns
  karnataka_data <- karnataka_data %>% drop_na(!!sym(consumption_column),
!!sym(district_column))

  # Plotting the histogram
  ggplot(karnataka_data, aes(x = !!sym(consumption_column))) +
    geom_histogram(bins = 30, fill = "blue", alpha = 0.7, color = "black") +
    geom_density(alpha = 0.2, fill = "red") +
    labs(title = 'Distribution of Total Consumption Across Different Districts in Karnataka',
         x = consumption_column,
         y = 'Frequency') +
    theme_minimal()

  # Plotting the barplot
  ggplot(karnataka_data, aes(x = as.factor(!!sym(district_column)), y =
!!sym(consumption_column))) +
    geom_bar(stat = "summary", fun = "mean", fill = "skyblue", color = "black") +
    labs(title = 'Consumption per District in Karnataka',
         x = 'State_Region',
         y = consumption_column) +
    theme_minimal() +
    theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))
} else {
  print(paste("Check the column names. Available columns are:",
paste(colnames(karnataka_data), collapse = ", ")))
}
------------------------------------------------------------------------------------------------
# Set the working directory and verify it
setwd("/Users/rakshith/Downloads")
print(paste("Current working directory:", getwd()))

# Load required libraries
install_and_load <- function(package) {
  if (!require(package, character.only = TRUE)) {
    install.packages(package, dependencies = TRUE)
    library(package, character.only = TRUE)
  }
}

# List of required libraries
libraries <- c("dplyr", "ggplot2", "sf")
lapply(libraries, install_and_load)

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

# Filtering for KA (Karnataka)
df <- subset(data, state_1 == "KA")

# Display dataset info
print("Dataset Information:")
print(colnames(df))
print(head(df))
print(dim(df))
```

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

# Subsetting the data
ka_new <- 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
ka_new$Meals_At_Home[is.na(ka_new$Meals_At_Home)] <-
mean(ka_new$Meals_At_Home, na.rm = TRUE)

# Finding outliers and removing them
remove_outliers <- function(df, column_name) {
  Q1 <- quantile(df[[column_name]], 0.25, na.rm = TRUE)
  Q3 <- quantile(df[[column_name]], 0.75, na.rm = TRUE)
  IQR <- Q3 - Q1
  lower_threshold <- Q1 - 1.5 * IQR
  upper_threshold <- Q3 + 1.5 * IQR
  df[df[[column_name]] >= lower_threshold & df[[column_name]] <= upper_threshold, ]
}

outlier_columns <- c('ricepds_v', 'chicken_q')
for (col in outlier_columns) {
  ka_new <- remove_outliers(ka_new, col)
}

# Summarize consumption
ka_new$total_consumption <- rowSums(ka_new[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(df, group_col) {
  summary <- df %>% group_by(!!sym(group_col)) %>% summarize(total_consumption =
sum(total_consumption)) %>% arrange(desc(total_consumption))
  return(summary)
}

district_summary <- summarize_consumption(ka_new, 'District')
region_summary <- summarize_consumption(ka_new, 'Region')

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

# Rename districts and sectors
district_mapping <- c("1" = "North West", "2" = "North", "3" = "North East", "4" = "East",
"5" = "New Delhi", "6" = "Central Delhi", "7" = "West", "8" = "South West", "9" = "South")
sector_mapping <- c("2" = "URBAN", "1" = "RURAL")

ka_new$District <- as.character(ka_new$District)
ka_new$Sector <- as.character(ka_new$Sector)
ka_new$District <- recode(ka_new$District, !!!district_mapping)
ka_new$Sector <- recode(ka_new$Sector, !!!sector_mapping)

# Display the updated dataframe
print(ka_new)

# Plotting the histogram
ggplot(ka_new, aes(x = total_consumption)) +
  geom_histogram(bins = 10, fill = 'blue', color = 'black') +
  labs(x = "Consumption", y = "Frequency", title = "Consumption Distribution in Karnataka
State")

# Aggregate total consumption by district
```

```
KA_consumption <- ka_new %>% group_by(District) %>% summarize(total_consumption = sum(total_consumption))
print("KA_consumption DataFrame:")
print(KA_consumption)

# Bar plot
ggplot(KA_consumption, aes(x = District, y = total_consumption)) +
  geom_bar(stat = "identity", fill = 'blue') +
  labs(x = "District", y = "Total Consumption", title = "Total Consumption per District") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
```