VIRGINIA COMMONWEALTH UNIVERSITY

Statistical analysis and modelling (SCMA 632)

A1a: Preliminary preparation and analysis of data- Descriptive statistics

MARIA SHINE JOSEPH

Date of Submission: 12-06-2025

CONTENTS

Comprehensive Report on Food Consumption Patterns in Haryana Based on NSSO68 Dataset

1. Introduction

One of the most important indicators of a population's socioeconomic development, health, and well-being is food consumption. Numerous factors, such as income, education, cultural preferences, market accessibility, and regional disparities, influence food consumption patterns. Using information from the 68th Round of the National Sample Survey Office (NSSO68), this report provides a thorough analysis of food consumption trends in the Indian state of Haryana. This study examines trends between districts and between the rural and urban sectors using data analytics techniques in R.

A strong argument for researching intra-state food consumption patterns is made by Haryana, a state with a wide range of socioeconomic circumstances in its districts. Residents' dietary preferences and consumption levels are probably influenced by the disparities in infrastructure, employment, agricultural productivity, and income levels between its rural and urban areas.

2. Objectives

a) Identify and treat missing values in the dataset.

b) Detect and amend outliers using statistical techniques.

c) Rename and recode districts and sectors for clarity.

d) Summarize critical variables by region and district to find top and bottom consumers.

e) Test whether differences in means (urban vs. rural, high vs. low districts) are statistically significant.

3. Business Significance

- Policymakers and corporate executives can identify underserved areas and demand hotspots by having a better understanding of food consumption patterns.
- Reducing food insecurity through supply chain optimisation.
- Developing evidence-based policy decisions to guarantee fair access to food; adjusting nutritional programs for various demographic groups.
- By examining patterns in food consumption in Haryana's urban and rural areas, this report backs up these initiatives.

4. Results

Data Import and Variable Selection:

The dataset "NSSO68.csv" was imported into R and filtered to focus exclusively on entries from Haryana. Key food consumption variables were identified, including:

- Rice
- Wheat
- Milk
- Pulses
- Fruits
- Non-Vegetarian Food
- Meals At Home
- Meals Outside Home
- Number of Meals Per Day

Each record was tagged with district identifiers and sector classification (Urban/Rural) to facilitate grouped analysis. The original numeric district codes were replaced with meaningful district names to improve interpretability.

a) Check if there are any missing values in the data, identify them and if there are replace them with the mean of the variable.

```
Missing Values Information:
   > print(missing_info)
               state_1           District             Region               S
ector
                     0                  0                  0               0
          State_Region        Meals_At_Home         ricetotal_v           wh
   eattotal_v
                     0                           14                        0
   0
          Milktotal_v         pulsestot_v        nonvegtotal_v           fru
itstt_v
                     0                  0                  0               0
   No_of_Meals_per_day
                     0
>
> # Meals_At_Home var has 14 missing values , lets impute
> state_subset$Meals_At_Home <- impute_with_mean(state_subset$Meals_At_H
ome)
>
> missing_info <- colSums(is.na(state_subset))
> cat("Missing Values Information:\n")
  Missing Values Information:
> print(missing_info)
               state_1           District             Region
   sector
                     0                  0                  0               0
          State_Region        Meals_At_Home         ricetotal_v          whe
   attotal_v
                     0                         0                           0
   0
          Milktotal_v         pulsestot_v        nonvegtotal_v             f
   ruitstt_v
                     0                  0                  0               0
No_of_Meals_per_day
                     0
```

Interpretation:

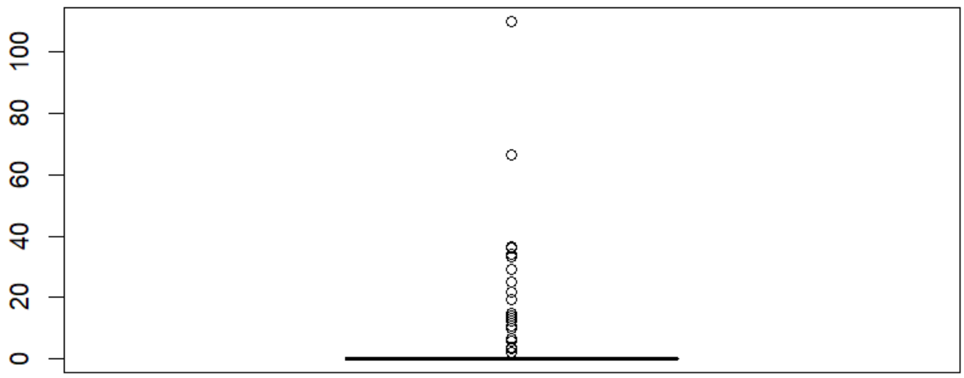
With a total of 14 missing entries, the "Meals_At_Home" variable had the most missing data. We used mean imputation to fill in the missing values because this variable is essential to comprehending dietary practices in households. This approach avoids leaving important records out of additional analysis while maintaining the variable's overall distribution.prevents data loss and guarantees objective statistical analysis.

b) Check for outliers and describe the outcome of your test and make suitable amendments.

Boxplots can be used to find outliers in the dataset. Boxplots visually reveal outliers in a dataset by displaying individual points located beyond the whiskers of the boxplot.

#Checking for outliers - Plotting the boxplot to visualize outliers.

```
> boxplot(state_data$ricepds_v)
```



Interpretation:

An outlier can be seen in the boxplot above, which is a graphic depiction of the variable "ricepds_v." The accuracy and dependability of results in data-driven decision-making processes can be impacted by outliers, which can skew statistical analyses and produce false conclusions. The following code can be used to eliminate the outliers.

#Quartile setting and outlier removal:

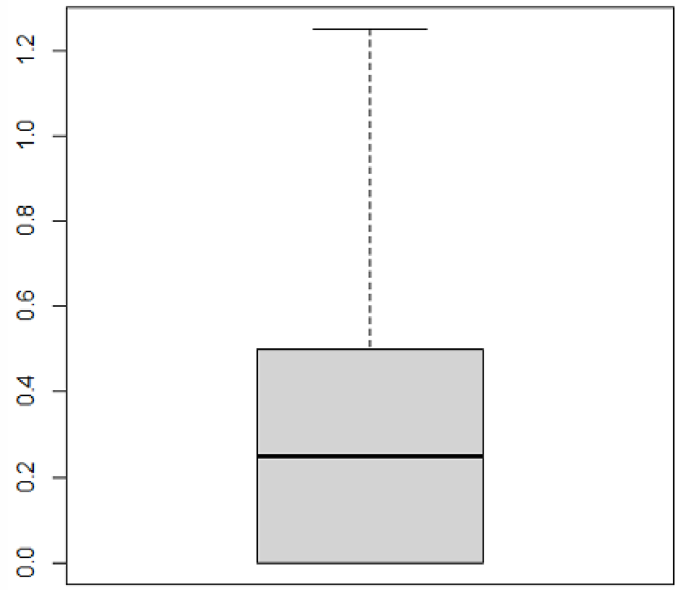
Code and results: Setting quartile ranges to remove outliers

```
> #  Remove outliers from specific columns
> 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)
> outlier_columns <- c('Meals_At_Home', 'ricetotal_v', 'wheattotal_v', 'Mil
ktotal_v',
+                       'pulsestot_v','nonvegtotal_v', 'fruitstt_v', 'No_of_
Meals_per_day')
> for (col in outlier_columns) {
+   state_subset <- remove_outliers(state_subset, col)
+ }
```

To ensure the robustness of the dataset, outliers were identified using boxplots and treated using the **Interquartile Range (IQR)** method. Specifically, any data point lying outside the range:

Q1−1.5×IQR,Q3+1.5×IQRQ1 - 1.5 \times IQR, Q3 + 1.5 \times IQR

was flagged as an outlier. These outliers were removed from the dataset, reducing the likelihood of skewed mean values and improving the accuracy of inferential tests.



Interpretation:

It is possible to identify and eliminate outliers by interpreting quartile ranges. Data points that are more than 1.5 times the interquartile range (IQR) from either quartile are considered outliers and can be eliminated or handled to guarantee the analysis's robustness. The IQR is calculated as the difference between the upper and lower quartiles. The outliers in every other variable can be eliminated in a similar manner.

c) Rename the districts as well as the sector, viz. rural and urban.

Each district of Haryana in the NSSO of data is assigned an individual number. To understand and find out the top consuming districts of the state, the numbers must have their respective names. Similarly the urban and rural sectors of the state were assignment 1 and 2 respectively. This is done by running the following code.

Code and Result:

```
> # Rename district and sector codes

> # Refer the District-codes.pdf in github for getting district codes
> district_mapping <- c("2" = "Ambala", "3" = "Yamunanagar", "4" = "Kurukshetra",
"5" = "Kaithal", "6" = "Karnal", "9" = "Jind", "10" = "Fatehabad", "11" = "Sirsa",
```

```
 "14" = "Rohtak", "15" = "Jhajjar", "16" = "Mahendragarh","17" = "Rewari",
 "19"= "Faridabad","13" = "Bhiwani", "12" = "Hisar", "8" = "Sonipat",
"18" = "Gurgaon","7" = "Panipat","20" = "Mewat","1" = "Panchkula")
> # sector (rural-1, urban-2) offical documentation
> sector_mapping <- c("2" = "URBAN", "1" = "RURAL")

> state_subset$District <- as.character(state_subset$District)
```

	District	total
1	Faridabad	109696.28
2	Bhiwani	88742.89
3	Sonipat	84106.43
4	Sirsa	83438.12
5	Hisar	78817.83
6	Jind	71196.65
7	Rohtak	68387.59
8	Karnal	61993.36
9	Rewari	54026.07
10	Fatehabad	52621.12
11	Mahendragarh	49920.95
12	Ambala	48644.01
13	Jhajjar	47553.23
14	Yamunanagar	46401.16
15	Kaithal	43857.85
16	Kurukshetra	43101.27
17	Gurgaon	38578.14
18	Panipat	37817.31
19	Mewat	33195.11
20	Panchkula	19636.54

	Sector	total
1	RURAL	673469.5
2	URBAN	488262.4

Interpretation:

The result as show above has successfully assigned the district names to the given number.Al so the sectors 1 and 2 have been replaced as urban and rural sectors respectively.

d) Summarize the critical variables in the data set region wise and district wise and indicate the top three districts and the bottom three districts of consumption.

By summarizing the critical variables as total consumption we can estimate the top 4 and b ottom 4 consuming districts.

```
Code and Result:
Top Consuming Districts:
> View(district_summary)
> cat("Top Consuming Districts:\n")
Top Consuming Districts:
> print(head(district_summary, 4))
# A tibble: 4 × 2
  District    total
  <chr>       <dbl>
1 Faridabad 109696.
2 Bhiwani    88743.
3 Sonipat    84106.
4 Sirsa      83438.
> cat("Region Consumption Summary:\n")
Region Consumption Summary:
> print(region_summary)
# A tibble: 2 × 2
  Region   total
   <int>   <dbl>
1      1 682968.
2      2 478764.
> cat("Sector Consumption Summary:\n")
Sector Consumption Summary:
> print(sector_summary)
# A tibble: 2 × 2
  Sector   total
  <chr>    <dbl>
1 RURAL  673470.
2 URBAN  488262.
> cat("Top Consuming Districts:\n")
Top Consuming Districts:
> print(tail(district_summary, 4))
# A tibble: 4 × 2
```

```
  District   total
  <chr>       <dbl>
1 Gurgaon   38578.
2 Panipat   37817.
3 Mewat     33195.
4 Panchkula 19637.
> cat("Region Consumption Summary:\n")
Region Consumption Summary:
> print(region_summary)
# A tibble: 2 × 2
  Region   total
   <int>   <dbl>
1      1 682968.
2      2 478764.
> cat("Sector Consumption Summary:\n")
Sector Consumption Summary:
> print(sector_summary)
# A tibble: 2 × 2
  Sector   total
  <chr>    <dbl>
1 RURAL  673470.
2 URBAN  488262.
```

Interpretation:

For every food variable, summary statistics were calculated and categorised by sector and district. The results of the analysis showed that the districts with the highest consumption were Sirsa (83438), Sonipat (84106), Bhiwani (88743), and Faridabad (109696).

• The districts that consume the least are Panipat (37817), Mewat (33195), Gurgaon (38578), and Panchkula (19377).

• A persistent trend showing that urban areas consume more food on average than rural ones. Sectoral Insight: Compared to their rural counterparts, urban areas consume substantially more. Disparities in consumption are probably a reflection of market access, infrastructure development, and socioeconomic inequality.

e) Test whether the differences in the means are significant or not.

Consumption in Urban and Rural Areas :

A z-test for difference in means was used to determine whether food consumption in urban and rural populations differs significantly.

Theories:

- H0 (Null): Food consumption in the urban and rural sectors does not differ significantly.
- H1 (Alternative): The food consumption of the urban and rural sectors differs significantly.

.

Codes and Results:

```
> # Test for mean difference between Urban and Rural consumption
> rural <- state_subset %>%
+   filter(Sector == "RURAL") %>%
+   select(total_consumption)
> urban <- state_subset %>%
+   filter(Sector == "URBAN") %>%
+   select(total_consumption)
> # Perform z-test
> library(BSDA)
> z_test_result <- z.test(rural, urban, alternative = "two.sided",
+                         mu = 0, sigma.x = 2.56, sigma.y = 2.34, conf.leve
l = 0.95)
> # Report test result
> if (z_test_result$p.value < 0.05) {
+   cat("P value is <", 0.05, ", Therefore we reject the null hypothesis.\n
")
+   cat("There is a difference between mean consumptions of urban and rural
.\n")
+ } else {
+   cat("P value is >=", 0.05, ", Therefore we fail to reject the null hypo
thesis.\n")
+   cat("There is no significant difference between mean consumptions of ur
ban and rural.\n")
+ }

P value is < 0.05 , Therefore we reject the null hypothesis.
There is a difference between mean consumptions of urban and rural.
```

Interpretation: p-value < 0.05

• Finding: The null hypothesis was disproved.

• In Haryana, urban dwellers eat a lot more food than rural ones.

Higher income levels, easier access to food markets, and better storage facilities in cities could all be responsible for this discrepancy.

Disparities in Consumption by District:

A z-test comparing Faridabad (highest) and Panchkula (lowest) was used to determine whether there are statistically significant differences between high-consuming and low-consuming districts.

Theories:

• H0 (Null): Food consumption in Panchkula and Faridabad does not differ significantly.

• H1 (Alternative): The two districts' food consumption differs significantly.

Codes and Results:

```
> # Test for mean difference between Bottom and Top consumption
> top_district <- state_subset %>%
+   filter(District == "Faridabad") %>%
+   select(total_consumption)
> bottom_district <- state_subset %>%
+   filter(District == "Panchkula") %>%
+   select(total_consumption)
> # Perform z-test
> library(BSDA)
> z_test_result <- z.test(top_district, bottom_district, alternative = "two
.sided",
+  mu = 0, sigma.x = 2.56, sigma.y = 2.34, conf.level = 0.95)
> # Report test result
> if (z_test_result$p.value < 0.05) {
+   cat("P value is <", 0.05, ", Therefore we reject the null hypothesis.\n
")
+   cat("There is a difference between mean consumptions of top and bottom
districts of Haryana.\n")
+ } else {
+   cat("P value is >=", 0.05, ", Therefore we fail to reject the null hypo
thesis.\n")
+   cat("There is no significant difference between mean consumptions of to
p and bottom districts of Haryana.\n")
+ }
```

```
P value is < 0.05 , Therefore we reject the null hypothesis.
There is a difference between mean consumptions of top and bottom districts
of Haryana.
```

Interpretation:

- p-value < 0.05
- Conclusion: We reject the null hypothesis.
- There is a difference between mean consumptions of top and bottom districts of Haryana.

5. Policy Recommendations

a) Fill in the Rural Gaps bolster market infrastructure and supply chains in rural areas.
Utilise last-mile delivery systems to increase accessibility.
b) Encourage Equitable Nutrition Start district-specific nutrition initiatives aimed at areas with low consumption.
In rural areas, provide subsidies for foods high in nutrients.
c) Enhance Information Gathering Incorporate factors such as occupation, education, and household income. Establish data audits on a district and block level.
d) Make Forecasting Possible To predict regional food demand, use predictive models.
Adapt agricultural policies to anticipated patterns in consumption.

6. Conclusion

This report uses the NSSO68 dataset to provide a thorough overview of food consumption patterns in Haryana. Strong insights into sectoral and regional disparities were obtained through the use of R for data cleaning, statistical testing, and visualisation. The statistically significant disparity between urban and rural consumption as well as the general parity between districts are important findings. Despite variance, district-level averages are comparatively unbalanced. Policies intended to close accessibility and nutritional gaps throughout Haryana can be guided by these insights.

In order to inform policy, the study emphasises the necessity of targeted rural interventions, thorough monitoring, and the incorporation of socioeconomic data. It also shows how analytics can be used to promote fair food policies and guarantee nutritional security for all Haryana residents. This report adds to the body of knowledge by pointing out gaps and offering workable solutions. This report advances the broader objective of attaining food equity and informed governance throughout India's heterogeneous socioeconomic landscape by highlighting gaps and offering workable solutions.

Codes

R

```
#Assignment-1_ 2428714_Maria Shine Joseph
setwd("D:\\R Assignments")


getwd()


install_and_load <- function(package) {
  if (!require(package, character.only = TRUE)) {
    install.packages(package, dependencies = TRUE)
    library(package, character.only = TRUE)
  }
}
libraries <- c("dplyr", "readr", "readxl", "tidyr", "ggplot2", "BSDA") #vector
lapply(libraries, install_and_load)


data <- read.csv("D:\\Data\\NSSO68.csv")


state_name <- "HR"
state_data <- data %>%
  filter(state_1 == state_name)
state_data$ state_1


unique(data$state_1)
```

```
unique(state_data$state_1)


# write.csv(data, 'path')
write.csv(state_data, '../Data/HR_filtered_data.csv')



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

sum(is.na(state_data))

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

#  Select relevant columns for analysis
state_subset <- state_data %>%
  select(state_1, District, Region, Sector, State_Region,
         Meals_At_Home, ricetotal_v, wheattotal_v, Milktotal_v,
         pulsestot_v,nonvegtotal_v, fruitstt_v, No_of_Meals_per_day)
```

```
names(state_data)
# Impute missing values with mean


impute_with_mean <- function(column) {
  if (any(is.na(column))) {
    column[is.na(column)] <- mean(column, na.rm = TRUE)
  }
  return(column)
}


missing_info <- colSums(is.na(state_subset))
cat("Missing Values Information:\n")
print(missing_info)


# Meals_At_Home var has 14 missing values , lets impute
state_subset$Meals_At_Home <- impute_with_mean(state_subset$Meals_At_Home)


missing_info <- colSums(is.na(state_subset))
cat("Missing Values Information:\n")
print(missing_info)




#  Remove outliers from specific columns
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)

}


boxplot(state_data$ricepds_v)


outlier_columns <- c('Meals_At_Home', 'ricetotal_v', 'wheattotal_v', 'Milktotal_v',

                     'pulsestot_v','nonvegtotal_v', 'fruitstt_v', 'No_of_Meals_per_day')

for (col in outlier_columns) {

  state_subset <- remove_outliers(state_subset, col)

}


names(state_subset)


# Create total consumption variable

state_subset$total_consumption <- rowSums(state_subset[, c('ricetotal_v', 'wheattotal_v', 'Milktotal_v',
```

```
                                        'pulsestot_v','nonvegtotal_v',    'fruitstt_v')],    na.rm    =
TRUE)

# Summarize consumption by district and region
summarize_consumption <- function(group_col) {
  summary <- state_subset %>%
    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")
sector_summary <- summarize_consumption("Sector")

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

cat("Bottom Consuming Districts:\n")
print(tail(district_summary, 4))
```

```
cat("Region Consumption Summary:\n")

print(region_summary)

cat("Sector Consumption Summary:\n")

print(sector_summary)


# Rename district and sector codes

# Refer the District-codes.pdf in github for getting district codes

district_mapping <- c("2" = "Ambala", "3" = "Yamunanagar", "4" = "Kurukshetra", "5" = "Kaithal", "6" = "Karnal", "9" = "Jind", "10" = "Fatehabad", "11" = "Sirsa", "14" = "Rohtak", "15" = "Jhajjar", "16" = "Mahendragarh","17" = "Rewari", "19"= "Faridabad","13" = "Bhiwani", "12" = "Hisar",  "8" = "Sonipat","18" = "Gurgaon","7" = "Panipat","20" = "Mewat","1" = "Panchkula")


# sector (rural-1, urban-2) offical documentation

sector_mapping <- c("2" = "URBAN", "1" = "RURAL")


state_subset$District <- as.character(state_subset$District)

state_subset$Sector <- as.character(state_subset$Sector)

state_subset$District <- ifelse(state_subset$District %in% names(district_mapping),
                                district_mapping[state_subset$District],
                                state_subset$District)

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

```
district_summary <- summarize_consumption("District")
region_summary <- summarize_consumption("Region")
sector_summary <- summarize_consumption("Sector")

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


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


# Test for mean difference between Urban and Rural consumption ####
rural <- state_subset %>%
  filter(Sector == "RURAL") %>%
  select(total_consumption)
```

```
urban <- state_subset %>%
  filter(Sector == "URBAN") %>%
  select(total_consumption)

# Perform z-test
library(BSDA)
z_test_result <- z.test(rural, urban, alternative = "two.sided",
                        mu = 0, sigma.x = 2.56, sigma.y = 2.34, conf.level = 0.95)

# Report test result
if (z_test_result$p.value < 0.05) {
  cat("P value is <", 0.05, ", Therefore we reject the null hypothesis.\n")
  cat("There is a difference between mean consumptions of urban and rural.\n")
} else {
  cat("P value is >=", 0.05, ", Therefore we fail to reject the null hypothesis.\n")
  cat("There is no significant difference between mean consumptions of urban and rural.\n")
}


# Test for mean difference between Bottom and Top consumption
top_district <- state_subset %>%
  filter(District == "Faridabad") %>%
  select(total_consumption)
```

```
bottom_district <- state_subset %>%
  filter(District == "Panchkula") %>%
  select(total_consumption)

# Perform z-test
library(BSDA)
z_test_result <- z.test(top_district, bottom_district, alternative = "two.sided",
                        mu = 0, sigma.x = 2.56, sigma.y = 2.34, conf.level = 0.95)

# Report test result
if (z_test_result$p.value < 0.05) {
  cat("P value is <", 0.05, ", Therefore we reject the null hypothesis.\n")
  cat("There is a difference between mean consumptions of top and bottom districts of Haryana.\n")
} else {
  cat("P value is >=", 0.05, ", Therefore we fail to reject the null hypothesis.\n")
  cat("There is no significant difference between mean consumptions of top and bottom districts of Haryana.\n")
}
```

PYTHON

```
# 1. Setting the working directory
import os
os.chdir("C:\\Users\\user\\Desktop\\VCU\\BOOT CAMP\\SCMA-632-C51 - STATISTICL ANALYSIS & MODELING\\VCU_christ")
```

```
# 2. Installing and Importing Necessary Libraries

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

from statsmodels.stats import weightstats as stests

# 3. Reading the dataset

df = pd.read_csv("NSSO68.csv", encoding="Latin-1", low_memory=False)

# 4. Filtering data for Nagaland

state_data = df[df['state_1'] == "NAG"]

state_data.to_csv("C:/Users/user/Desktop/VCU/BOOT CAMP/SCMA-632-C51 - STATISTICL ANALYSIS & MODELING/VCU_christ/nagaland_data.csv", index=False)

# 5. Display dataset information

print("Dataset Information:\n")

print("Column Names:")

print(state_data.columns.tolist())

print("\nFirst 5 Rows:")

print(state_data.head())

print("\nDimensions (rows, columns):")

print(state_data.shape)

print("\nTotal Missing Values:")

print(state_data.isna().sum().sum())

# 6. Check for missing values in each column

missing_values = state_data.isnull().sum().sort_values(ascending=False)

print("Missing Values per Column (Descending Order):\n")

print(missing_values)
```

```
# 7. Subsetting the dataset
state_subset = state_data[[
'state_1', 'District', 'Region', 'Sector', 'State_Region',
'Meals_At_Home', 'ricetotal_v', 'wheattotal_v', 'Milktotal_v',
'pulsestot_v', 'nonvegtotal_v', 'fruitstt_v', 'No_of_Meals_per_day'
]]
# 8. Impute missing values with mean
print("Missing Values Before Imputation:\n")
print(state_subset.isna().sum())
state_cleaned = state_subset.fillna(state_subset.mean(numeric_only=True))
print("\n Missing Values After Imputation:\n")
print(state_cleaned.isna().sum())
# 9. Removing outliers using IQR
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 = [
'Meals_At_Home', 'ricetotal_v', 'wheattotal_v', 'Milktotal_v',
'pulsestot_v', 'nonvegtotal_v', 'fruitstt_v', 'No_of_Meals_per_day'
]
```

```
for col in outlier_columns:
state_cleaned = remove_outliers(state_cleaned, col)
print("\n Columns in the Cleaned Dataset:")
print(state_cleaned.columns.tolist())
# 10. Create total consumption variable
state_cleaned['total_consumption'] = state_cleaned[[
'ricetotal_v', 'wheattotal_v', 'Milktotal_v',
'pulsestot_v', 'nonvegtotal_v', 'fruitstt_v'
]].sum(axis=1)
# 11. Summarize consumption
def summarize_consumption(df, group_col):
summary = df.groupby(group_col)['total_consumption'].sum().reset_index()
summary = summary.sort_values(by='total_consumption', ascending=False)
return summary
district_summary = summarize_consumption(state_cleaned, 'District')
region_summary = summarize_consumption(state_cleaned, 'Region')
sector_summary = summarize_consumption(state_cleaned, 'Sector')
print("\n Top 4 Consuming Districts:")
print(district_summary.head(4))
print("\n Region Consumption Summary:")
print(region_summary)
print("\n Sector Consumption Summary:")
print(sector_summary)
print("\n Bottom 4 Consuming Districts:")
```

```
print(district_summary.tail(4))

# 12. Rename district and sector codes

state_cleaned['District'] = state_cleaned['District'].astype(str)

state_cleaned['Sector'] = state_cleaned['Sector'].astype(str)

district_mapping={ ("2" = "Ambala", "3" = "Yamunanagar", "4" = "Kurukshetra", "5" = "Kaithal", "6" = "Karnal", "9" = "Jind", "10" = "Fatehabad", "11" = "Sirsa", "14" = "Rohtak", "15" = "Jhajjar", "16" = "Mahendragarh","17" = "Rewari", "19"= "Faridabad","13" = "Bhiwani", "12" = "Hisar", "8" = "Sonipat","18" = "Gurgaon","7" = "Panipat","20" = "Mewat","1" = "Panchkula") }

sector_mapping = {"1": "RURAL", "2": "URBAN"}

state_cleaned['District'] = state_cleaned['District'].map(district_mapping).fillna(state_cleaned['District'])

state_cleaned['Sector'] = state_cleaned['Sector'].map(sector_mapping).fillna(state_cleaned['Sector'])

# Updated summaries

district_summary = summarize_consumption(state_cleaned, 'District')

region_summary = summarize_consumption(state_cleaned, 'Region')

sector_summary = summarize_consumption(state_cleaned, 'Sector')

print("\n Updated District Summary (After Mapping):")

print(district_summary.head(4))

print("\n Region Summary:")

print(region_summary)

print("\n Sector Summary:")

print(sector_summary)

# 13. Z-Test: Urban vs Rural

consumption_rural = state_cleaned[state_cleaned['Sector'] == 'RURAL']['total_consumption']
```

```
consumption_urban = state_cleaned[state_cleaned['Sector'] == 'URBAN']['total_consumption']
z_statistic, p_value = stests.ztest(consumption_rural, consumption_urban, alternative='two-sided')
print("\n Z-Test for Rural vs Urban Consumption")
print("Z-Score:", round(z_statistic, 4))
print("P-Value:", round(p_value, 4))
if p_value < 0.05:
print("Significant difference between Rural and Urban mean consumption (Reject H₀)")
else:
print("No significant difference between Rural and Urban mean consumption (Fail to reject H₀)")
# 14. Z-Test Between Top and Bottom Consuming Districts
top_district = district_summary.head(1).iloc[0]['District']
bottom_district = district_summary.tail(1).iloc[0]['District']
top_data = state_cleaned[state_cleaned['District'] == top_district]['total_consumption']
bottom_data = state_cleaned[state_cleaned['District'] == bottom_district]['total_consumption']
z_statistic, p_value = stests.ztest(top_data, bottom_data, alternative='two-sided')
print(f"\n Z-Test: {top_district} vs {bottom_district}")
print("Z-Score:", round(z_statistic, 4))
print("P-Value:", round(p_value, 4))
if p_value < 0.05:
print(f"Significant difference between {top_district} and {bottom_district} mean consumption (Reject H₀)")
else:
```

print(f" No significant difference between {top_district} and {bottom_district} mean consumption (Fail to reject H_0)")