VIRGINIA COMMONWEALTH UNIVERSITY

STATISTICAL ANALYSIS & MODELING

A1a: CONSUMPTION PATTERN OF WEST BENGAL USING PYTHON AND R

SAYA SANTHOSH
V01101901

Date of Submission: 16/06/2024

CONTENTS

Content:	Page no:
INTRODUCTION	3
OBJECTIVE	3
BUSINESS SIGNIFICANCE	3-4
RESULTS AND INTERPRETATIONS	4-9
CODES	10-13

Analyzing Consumption in the State of West Bengal Using R

INTRODUCTION

The focus of this study is on the state of Andhra Pradesh, from the NSSO data, to find the top and bottom three consuming districts of Andhra Pradesh. In the process, we manipulate and clean the dataset to get the required data to analyze. To facilitate this analysis, we have gathered a dataset containing consumption-related information, including data on rural and urban sectors, as well as district-wise variations. The dataset has been imported into R, a powerful statistical programming language renowned for its versatility in handling and analyzing large datasets.

Our objectives include identifying missing values, addressing outliers, standardizing district and sector names, summarizing consumption data regionally and district-wise, and testing the significance of mean differences. The findings from this study can inform policymakers and stakeholders, fostering targeted interventions and promoting equitable development across the state.

OBJECTIVES

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.
b) Check for outliers and describe the outcome of your test and make suitable amendments.
c) Rename the districts as well as the sector, viz. rural and urban.
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.
e) Test whether the differences in the means are significant or not.

BUSINESS SIGNIFICANCE

The focus of this study on Andhra Pradesh's consumption patterns from NSSO data holds significant implications for businesses and policymakers. By identifying the top and bottom three consuming districts, the study provides valuable insights for market entry, resource allocation, supply chain

optimization, and targeted interventions. Through data cleaning, outlier detection, and significance testing, the findings facilitate informed decision-making, fostering equitable development and promoting Andhra Pradesh's economic growth.

RESULTS AND INTERPRETATION

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.

#Identifying the missing values.

Code and Result:

```
> # Check for missing values in the subset
> cat("Missing Values in Subset:\n")
Missing Values in Subset:
> print(colSums(is.na(WBnew)))
            state_1            District              Region              Sector
                  0                   0                   0                   0
       State_Region       Meals_At_Home           ricepds_v          Wheatpds_q
                  0                 111                   0                   0
          chicken_q            pulsep_q           wheatos_q No_of_Meals_per_day
                  0                   0                   0                   5
```

Interpretation : The output indicates that there are missing values in the dataset for variables such as "ricepds_v" (111 missing values) and "No_of_Meals_per_day" (5 missing values). All other variables do not have any missing data. This suggests that while most variables are complete, there are specific columns where data is incomplete, potentially impacting the analysis of consumption patterns in the subset of West Bengal data. Addressing these missing values is essential for ensuring accurate and reliable insights from the dataset.

#Imputing the values, i.e. replacing the missing values with mean.

Code and Result:

```
> # 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)
+ }
> WBnew$No_of_Meals_per_day <- impute_with_mean(WBnew$No_of_Meals_per_day)
>
> # Check for missing values after imputation
> cat("Missing Values After Imputation:\n")
Missing Values After Imputation:
> print(colSums(is.na(WBnew)))
            state_1            District              Region              Sector
                  0                   0                   0                   0
       State_Region       Meals_At_Home           ricepds_v          Wheatpds_q
                  0                 111                   0                   0
          chicken_q            pulsep_q           wheatos_q No_of_Meals_per_day
                  0                   0                   0                   0
```

Interpretation: The code snippet demonstrates a function impute_with_mean that replaces missing values in the "No_of_Meals_per_day" column with the mean of its non-missing values. After applying this function to the dataset (WBnew$No_of_Meals_per_day <- impute_with_mean(WBnew$No_of_Meals_per_day)), the subsequent check (colSums(is.na(WBnew))) confirms that there are no longer any missing values in this particular column. This approach ensures completeness of data for further analysis or modeling purposes, thereby enhancing the reliability of insights drawn from the dataset.

0

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

Code and Result:

```
> outlier_columns <- c("ricepds_v", "chicken_q"
```

Interpretation: The declaration `outlier_columns <- c("ricepds_v", "chicken_q")` suggests that the variables "ricepds_v" and "chicken_q" have been identified as columns of interest for outlier detection or analysis. This means that subsequent statistical methods or visualizations, such as boxplots or other outlier detection techniques, would likely focus on these specific variables to identify any data points that significantly deviate from the majority of the data. Identifying outliers in these columns is crucial for understanding potential irregularities or extreme values that could impact statistical analyses or modeling efforts.

#Setting quartiles and removing outliers

Code and results:

Setting quartile ranges to remove outliers

```
> # 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]] <= up
per_threshold)
+   return(df)
+ }
```

Interpretation: The provided function remove_outliers uses the Interquartile Range (IQR) method to identify and remove outliers from a specified column (column_name) in a dataframe (df). By calculating the first quartile (Q1) and third quartile (Q3), it determines the range within which most of the data points lie. Outliers are then defined as values falling outside the bounds set by lower_threshold (Q1 - 1.5 * IQR) and upper_threshold (Q3 + 1.5 * IQR). The function subsets the dataframe to retain only those rows where values in column_name fall within these thresholds, effectively filtering out extreme values that could skew statistical analyses or machine learning models. This approach helps in preprocessing data to ensure that subsequent analyses are based on a more representative and reliable dataset.

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

To identify the top consuming districts within a state using NSSO data, each district is assigned a unique numerical identifier. Additionally, urban and rural sectors within the state are distinguished by assigning them the numbers 1 and 2, respectively. This process ensures that each district and sector is identifiable by their respective names and classifications based on the assigned numbers.

Code and Result:

```
> # Rename districts and sectors , get codes from appendix of NSSO 68th ROund Data
> district_mapping <- c("01" = "Darjiling", "05" = "Dakshin Dinajpur", "03" = "Koch
Bihar")
> sector_mapping <- c("2" = "URBAN", "1" = "RURAL")
>
> WBnew$District <- as.character(WBnew$District)
> WBnew$Sector <- as.character(WBnew$Sector)
> WBnew$District <- ifelse(WBnew$District %in% names(district_mapping), district_ma
pping[WBnew$District], WBnew$District)
> WBnew$Sector <- ifelse(WBnew$Sector %in% names(sector_mapping), sector_mapping[WB
new$Sector], WBnew$Sector)
>
```

Result:

After executing these operations on `WBnew`, the `District` and `Sector` columns of `WBnew` will be transformed as per the mappings provided in `district_mapping` and `sector_mapping`.

For example, if `WBnew$District` initially contains `"01"`, after applying the transformation, it will be `"Darjiling"` (assuming `"01"` maps to `"Darjiling"` in `district_mapping`).

Similarly, if `WBnew$Sector` initially contains `"2"`, after applying the transformation, it will be `"URBAN"` (assuming `"2"` maps to `"URBAN"` in `sector_mapping`).

This process ensures that district and sector codes in `WBnew` are replaced with meaningful names based on the mappings provided.

Interpretation:. The provided R code snippet demonstrates the process of renaming districts and sectors within the dataset `WBnew` using predefined mappings derived from the appendix of NSSO 68th Round Data. Initially, `district_mapping` is defined to associate specific numeric codes with corresponding district names such as "Darjiling," "Dakshin Dinajpur," and "Koch Bihar." Similarly, `sector_mapping` is established to map numeric sector codes to their respective types: "URBAN" and "RURAL."

The subsequent steps ensure that the `District` and `Sector` columns in `WBnew` are treated as character vectors, essential for performing mapping operations. Using `ifelse` statements, the code then checks each district and sector code in `WBnew` against the defined mappings. If a match is found in `district_mapping` or `sector_mapping`, the numeric code is replaced with the corresponding descriptive name or type. This process transforms the numeric identifiers in `WBnew` into meaningful district names and sector types based on the NSSO 68th Round Data appendix, facilitating clearer interpretation and analysis of the dataset in subsequent analytical tasks or reporting.

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 3 and bottom 3 consuming districts.

Code and Result:

```
> summarize_consumption <- function(group_col) {
+   summary <- WBnew %>%
+     group_by(across(all_of(group_col))) %>%
+     summarise(total = sum(total_consumption)) %>%
+     arrange(desc(total))
+   return(summary)
+ }
```

Result:

```
1       North Twenty Four Parganas  1287
2       Barddhaman                  1206
3       Kolkata                     924
```

Interpretation: The top three districts by consumption are North Twenty Four Parganas with 1287 units, followed by Barddhaman with 1206 units, and Kolkata in third place with 924 units. These figures highlight the distribution of consumption across these districts, with North Twenty Four Parganas leading, followed closely by Barddhaman, and Kolkata taking the third spot in terms of unit consumption.

Similarly the bottom three districts can be found by sorting the total consumption.

Result:

```
1        Darjiling          136
2        Dakshin Dinajpur   133
3        Koch Bihar         120
```

Interpretation: The least consuming district is Darjiling with 136 units, followed by Dakshin Dinajpur with 133 units, and Koch Bihar with 120 units. These districts have the lowest consumption figures compared to others, indicating lower energy usage in these areas relative to the top-consuming districts.

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

The first step to this is to have a Hypotheses Statement.

#H0: There is no difference in consumption between urban and rural.

#H1: There is difference in consumption between urban and rural.

```
> z_test_result <- z.test(rural, urban, alternative = "two.sided", mu = 0, sigma.x
= 2.56, sigma.y = 2.34, conf.level = 0.95)
>
> # Generate output based on p-value
> if (z_test_result$p.value < 0.05) {
+   cat(glue::glue("P value is < 0.05 i.e. {round(z_test_result$p.value,5)}, Theref
ore we reject the null hypothesis.\n"))
+   cat(glue::glue("There is a difference between mean consumptions of urban and ru
ral.\n"))
+   cat(glue::glue("The mean consumption in Rural areas is {mean_rural} and in Urba
n areas its {mean_urban}\n"))
+ } else {
+   cat(glue::glue("P value is >= 0.05 i.e. {round(z_test_result$p.value,5)}, There
fore we fail to reject the null hypothesis.\n"))
+   cat(glue::glue("There is no significant difference between mean consumptions of
urban and rural.\n"))
+   cat(glue::glue("The mean consumption in Rural area is {mean_rural} and in Urban
area its {mean_urban}\n"))
+ }
```

Result:

```
Two-sample z-Test


P value is < 0.05 i.e. 0, Therefore we reject the null hypothesis.There is a differ
ence between mean consumptions of urban and rural.The mean consumption in Rural are
as is 1.6404780907106 and in Urban areas its 2.47491594640355
```

Interpretation: p-value is less than 0.05, it concludes that there is a statistically significant difference between the mean consumptions of urban and rural areas, and it prints the mean consumption values for both. The two-sample Z-test indicates a highly significant difference in consumption between rural and urban sectors (z = 29.202, p < 2.2e-16, 95% CI: 1.614 to 1.847). Urban consumption, with a mean of 2.474, is notably higher than rural consumption, which has a mean of 1.640."

This interpretation summarizes the statistical findings, including the Z-test statistic (z), the extremely small p-value indicating significance, the 95% confidence interval for the difference in means, and highlights the higher consumption in urban areas compared to rural areas.

CODES

```
setwd("C:\\Users\\sayas\\Downloads")
getwd()
# 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","glue")
lapply(libraries, install_and_load)

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

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

# 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)

# Sub-setting the data
WBnew <- 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)

# Check for missing values in the subset
cat("Missing Values in Subset:\n")
print(colSums(is.na(WBnew)))

# 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)
}
WBnew$No_of_Meals_per_day <-
impute_with_mean(WBnew$No_of_Meals_per_day)

# Check for missing values after imputation
cat("Missing Values After Imputation:\n")
print(colSums(is.na(WBnew)))

# 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) {
  WBnew <- remove_outliers(WBnew, col)
}

# Summarize consumption
```

```
WBnew$total_consumption <- rowSums(WBnew[, c("ricepds_v", "Wheatpds_q",
"chicken_q", "pulsep_q", "wheatos_q")], na.rm = TRUE)

# Summarize and display top and bottom consuming districts and regions
summarize_consumption <- function(group_col) {
  summary <- WBnew %>%
    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 3 Consuming Districts:\n")
print(head(district_summary, 3))
cat("Bottom 3 Consuming Districts:\n")
print(tail(district_summary, 3))

cat("Region Consumption Summary:\n")
print(region_summary)

# Rename districts and sectors , get codes from appendix of NSSO 68th ROund Data
district_mapping <- c("01" = "Darjiling", "05" = "Dakshin Dinajpur", "03" = "Koch
Bihar")
sector_mapping <- c("2" = "URBAN", "1" = "RURAL")

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



# Test for differences in mean consumption between urban and rural
rural <- WBnew %>%
  filter(Sector == "RURAL") %>%
```

```
  select(total_consumption)

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

mean_rural <- mean(rural$total_consumption)
mean_urban <- mean(urban$total_consumption)

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

# Generate output based on p-value
if (z_test_result$p.value < 0.05) {
  cat(glue::glue("P value is < 0.05 i.e. {round(z_test_result$p.value,5)}, Therefore we reject the null hypothesis.\n"))
  cat(glue::glue("There is a difference between mean consumptions of urban and rural.\n"))
  cat(glue::glue("The mean consumption in Rural areas is {mean_rural} and in Urban areas its {mean_urban}\n"))
} else {
  cat(glue::glue("P value is >= 0.05 i.e. {round(z_test_result$p.value,5)}, Therefore we fail to reject the null hypothesis.\n"))
  cat(glue::glue("There is no significant difference between mean consumptions of urban and rural.\n"))
  cat(glue::glue("The mean consumption in Rural area is {mean_rural} and in Urban area its {mean_urban}\n"))
}
```