# Set the working directory and verify it

setwd(C:\\Users\\Ahad\\OneDrive\\Pictures\\Documents\\Desktop\\scma)

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 Chhatisgarh

df <- data %>%

filter(state == 22)

# 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

CHTSDnew <- 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(CHTSDnew)))

# 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)

}

CHTSDnew$Meals\_At\_Home <- impute\_with\_mean(CHTSDnew$Meals\_At\_Home)

CHTSDnew$No\_of\_Meals\_per\_day <- impute\_with\_mean(CHTSDnew$No\_of\_Meals\_per\_day)

# Check for missing values after imputation

cat("Missing Values After Imputation:\n")

print(colSums(is.na(CHTSDnew)))

# 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) {

CHTSDnew <- remove\_outliers(CHTSDnew, col)

}

# Summarize consumption

CHTSDnew$total\_consumption <- rowSums(CHTSDnew[, 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 <- CHTSDnew %>%

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("11" = "Raipur", "10" = "Durg", "2" = "Surguja",

"16" = "Dantewada", "17" = "Narayanpur", "18" = "Bijapur")

sector\_mapping <- c("2" = "URBAN", "1" = "RURAL")

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

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

CHTSDnew$District <- ifelse(CHTSDnew$District %in% names(district\_mapping), district\_mapping[CHTSDnew$District], CHTSDnew$District)

CHTSDnew$Sector <- ifelse(CHTSDnew$Sector %in% names(sector\_mapping), sector\_mapping[CHTSDnew$Sector], CHTSDnew$Sector)

# Test for differences in mean consumption between urban and rural

rural <- CHTSDnew %>%

filter(Sector == "RURAL") %>%

select(total\_consumption)

urban <- CHTSDnew %>%

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"))

}

write.csv(CHTSDnew, "CHTSDnew.csv", row.names = FALSE)