VIRGINIA COMMONWEALTH UNIVERSITY

STATISTICAL ANALYSIS & MODELING

A1a: CONSUMPTION PATTERN OF UTTAR PRADESH USING PYTHON AND R

JESIN KANDATHY JOY
V01110163

Date of Submission: 16/06/2024

CONTENTS

INTRODUCTION

This report presents an analysis of socioeconomic data for Uttar Pradesh using the dataset "NSSO68.csv". The analysis employs R and Python, focusing on key steps such as handling missing values, detecting and managing outliers, and renaming categorical variables for consistency.

The core of this analysis involves summarizing critical variables region-wise and district-wise to uncover consumption patterns and economic disparities. By identifying the top and bottom three districts based on consumption metrics, the areas with the highest and lowest economic activity are highlighted. Additionally, hypothesis testing is conducted to assess significant differences in means across various segments, providing deeper insights into the socio-economic conditions of Uttar Pradesh. Through these comprehensive steps, the report aims to offer valuable insights into the state's economic landscape.

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

Understanding Uttar Pradesh's socioeconomic landscape through detailed analysis of NSSO data is crucial for stakeholders across various sectors. By examining consumption patterns and economic disparities within the state, this study offers actionable insights for businesses and policymakers alike.

Identifying consumption trends across different districts enables businesses to tailor their marketing strategies and distribution networks more effectively. For instance, insights gained from the top and bottom performing districts can guide companies in allocating resources and identifying growth opportunities in high-potential markets.

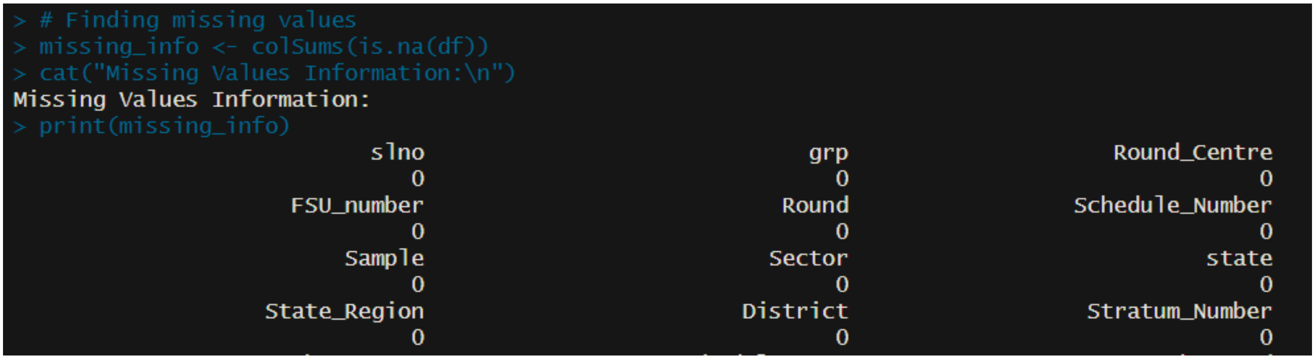
Policymakers can utilize these findings to prioritize development initiatives and allocate resources where they are most needed. By addressing economic disparities and promoting balanced regional growth, policymakers can foster inclusive development and improve the overall economic resilience of Uttar Pradesh.

This analysis not only sheds light on consumption dynamics but also serves as a valuable tool for strategic decision-making, enhancing business efficiency and driving sustainable socioeconomic development across Uttar Pradesh.

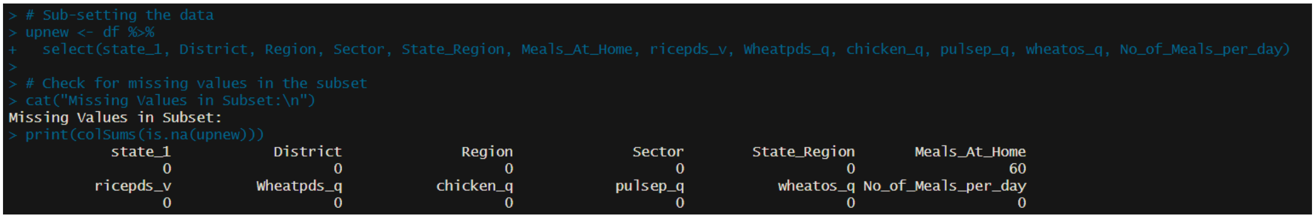
RESULTS AND INTERPRETATION

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.

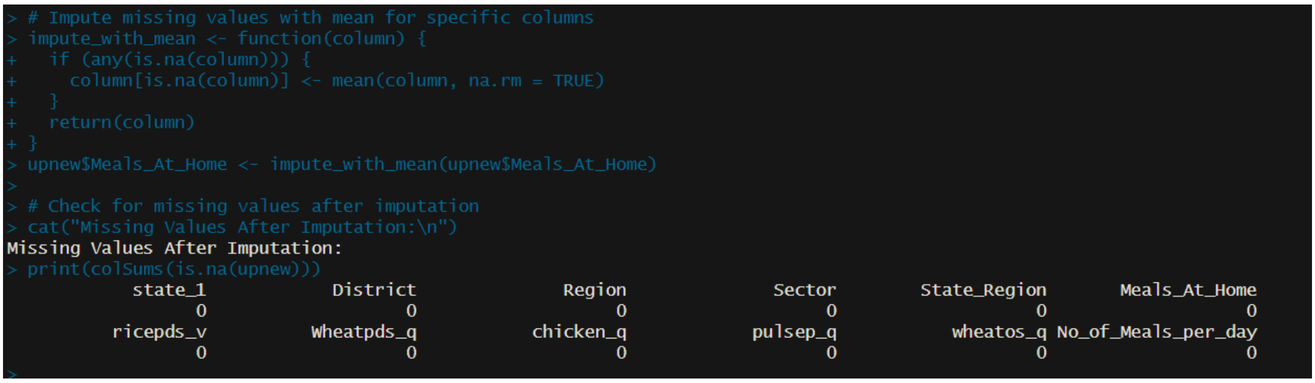
Sample code in R:



```
> # Finding missing values
> missing_info <- colSums(is.na(df))
> cat("Missing Values Information:\n")
Missing Values Information:
> print(missing_info)
                      slno                        grp               Round_Centre
                         0                          0                          0
                FSU_number                      Round            Schedule_Number
                         0                          0                          0
                    Sample                     Sector                      state
                         0                          0                          0
              State_Region                   District             Stratum_Number
                         0                          0                          0
```



```
> # Sub-setting the data
> upnew <- 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")
Missing Values in Subset:
> print(colSums(is.na(upnew)))
          state_1            District              Region              Sector        State_Region       Meals_At_Home
                0                   0                   0                   0                   0                  60
        ricepds_v          Wheatpds_q           chicken_q            pulsep_q           wheatos_q No_of_Meals_per_day
                0                   0                   0                   0                   0                   0
```



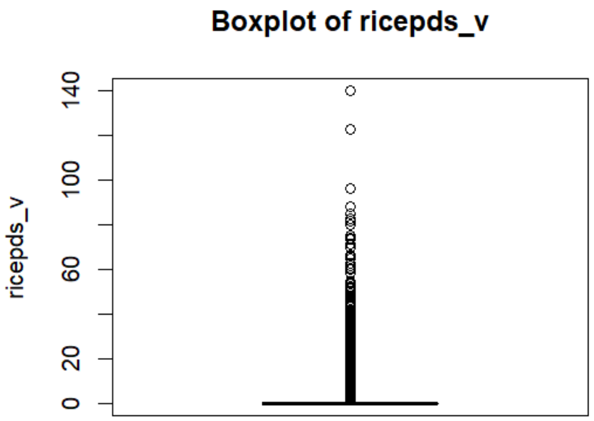
```
> # Impute missing values with mean for specific columns
> impute_with_mean <- function(column) {
+   if (any(is.na(column))) {
+     column[is.na(column)] <- mean(column, na.rm = TRUE)
+   }
+   return(column)
+ }
> upnew$Meals_At_Home <- impute_with_mean(upnew$Meals_At_Home)
>
> # Check for missing values after imputation
> cat("Missing Values After Imputation:\n")
Missing Values After Imputation:
> print(colSums(is.na(upnew)))
          state_1            District              Region              Sector        State_Region       Meals_At_Home
                0                   0                   0                   0                   0                   0
        ricepds_v          Wheatpds_q           chicken_q            pulsep_q           wheatos_q No_of_Meals_per_day
                0                   0                   0                   0                   0                   0
>
```

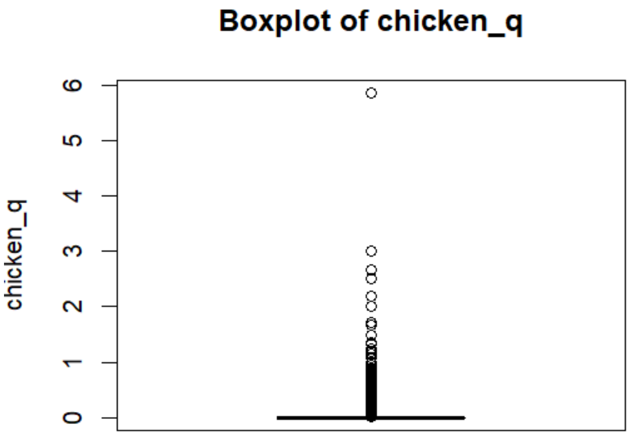
Interpretation:

1. The given code identifies and counts missing values in each column (df), providing an overview of data completeness.
2. Then it selects specific columns of interest (upnew) from the dataset (df), focusing on relevant data for further analysis.
3. And then verifies if there are any missing values in the selected subset (upnew), ensuring data completeness for analysis.
4. The code uses the mean to fill in missing values in the Meals_At_Home column of upnew, improving dataset completeness.
5. It confirms that all missing values in the upnew subset have been filled with the mean, ensuring data integrity.

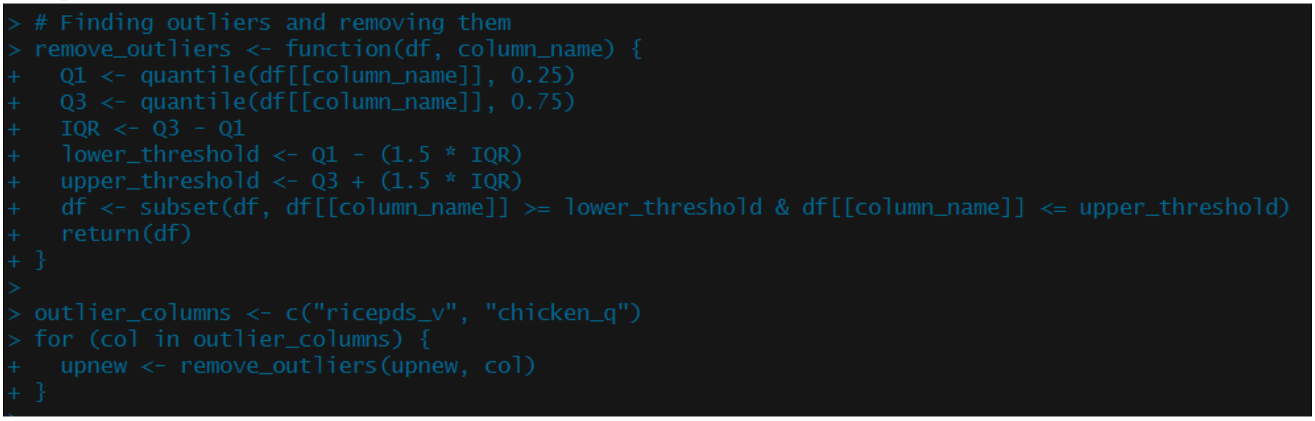
In general, the missing data is managed by first identifying gaps in the dataset, selecting relevant columns for analysis, and using statistical imputation to ensure data completeness before proceeding with further analysis.

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





Sample code in R:



```
> # 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) {
+   upnew <- remove_outliers(upnew, col)
+ }
```

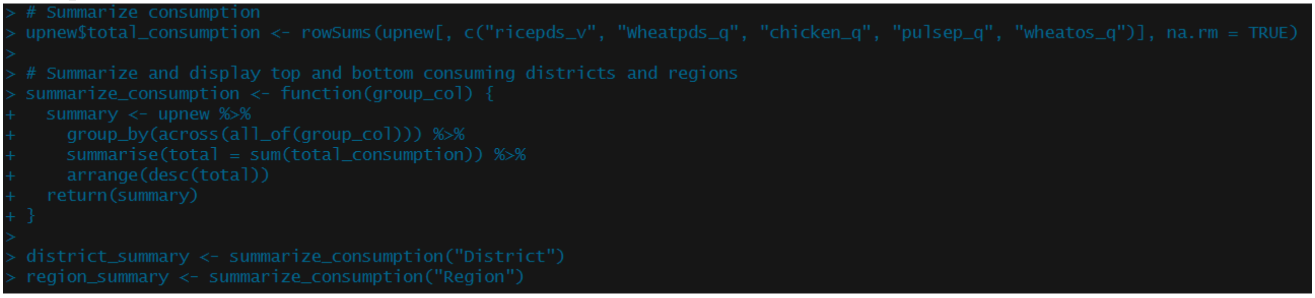
Interpretation:

Outliers are values that significantly differ from other observations in the dataset. Boxplots provide insights into the median, spread, and also the outliers of the data. In the given code, the function `remove_outliers` identifies outliers in specified columns (`ricepds_v` and `chicken_q`) using the Interquartile Range (IQR) method. After identifying outliers, the function removes these observations from `upnew` to ensure statistical robustness in subsequent analyses.

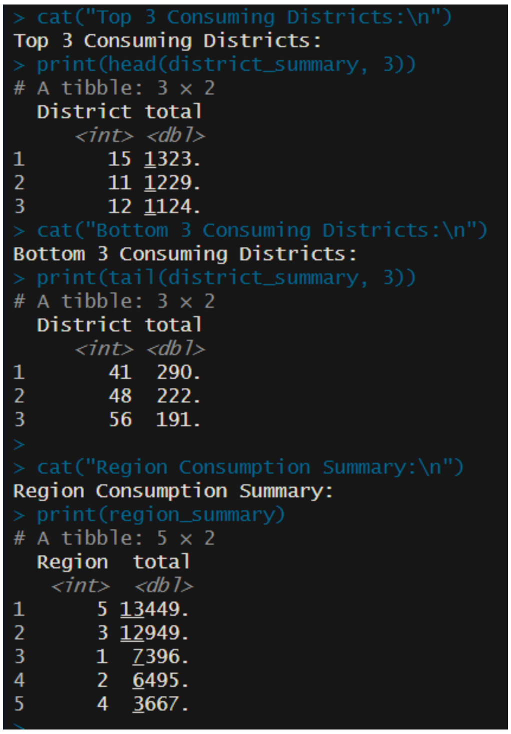
So generally, we use boxplots to visually assess the distributions of the required columns. Then we may employ a function to detect and remove outliers from these variables, ensuring that the dataset is more reliable for further statistical analyses by eliminating extreme data points that could skew results.

c) 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.

Sample code in R:



```
> # Summarize consumption
> upnew$total_consumption <- rowSums(upnew[, 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 <- upnew %>%
+     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")
```



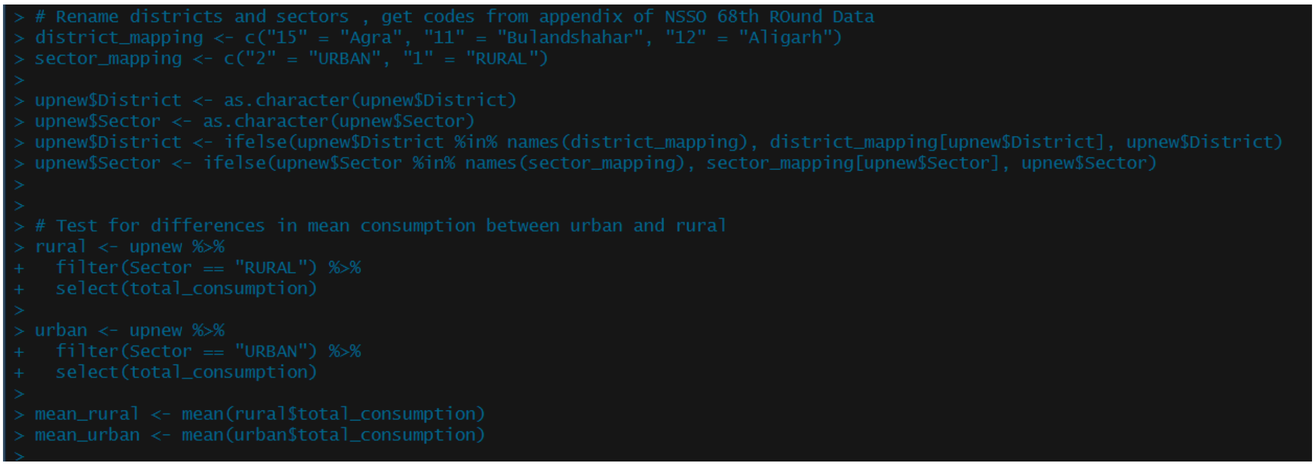
```
> cat("Top 3 Consuming Districts:\n")
Top 3 Consuming Districts:
> print(head(district_summary, 3))
# A tibble: 3 × 2
  District total
     <int> <dbl>
1       15 1323.
2       11 1229.
3       12 1124.
> cat("Bottom 3 Consuming Districts:\n")
Bottom 3 Consuming Districts:
> print(tail(district_summary, 3))
# A tibble: 3 × 2
  District total
     <int> <dbl>
1       41  290.
2       48  222.
3       56  191.
>
> cat("Region Consumption Summary:\n")
Region Consumption Summary:
> print(region_summary)
# A tibble: 5 × 2
  Region  total
   <int>  <dbl>
1      5 13449.
2      3 12949.
3      1  7396.
4      2  6495.
5      4  3667.
```

Interpretation:
The given code calculates `total_consumption` by summing specific columns (`ricepds_v`, `Wheatpds_q`, `chicken_q`, `pulsep_q`, `wheatos_q`) across rows in the dataset `upnew`. Then the function `summarize_consumption` groups the dataset `upnew` by either `District` or `Region`, computes the total consumption for each group, and arranges the results in descending order based on total consumption. It calculates and displays the top 3 consuming districts (`District`) and the bottom 3 consuming districts. It also summarizes Region consumption, displaying each region's total consumption.

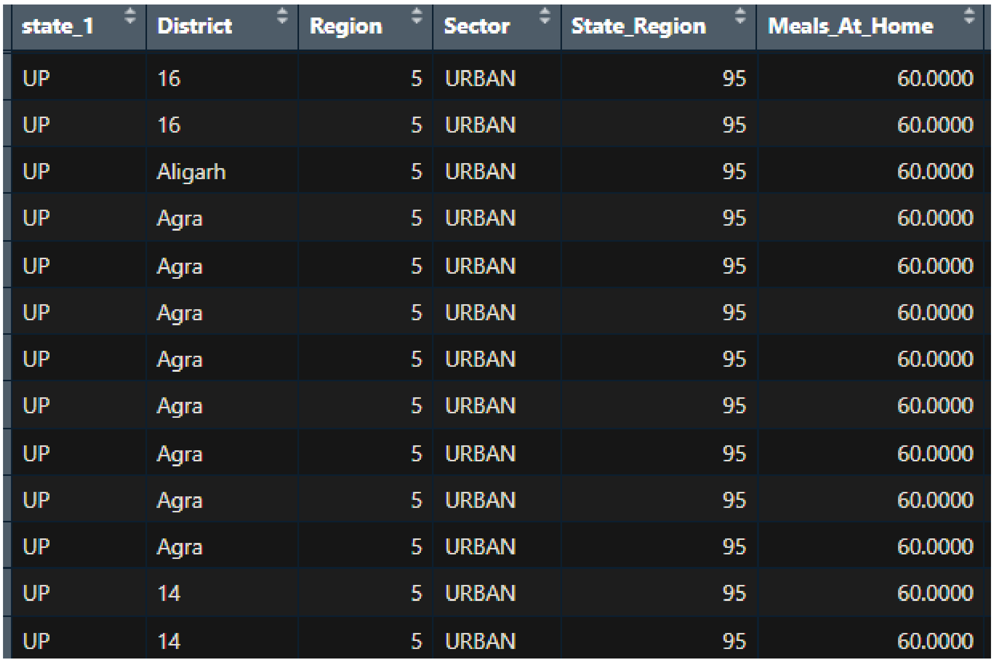
Here we analyze consumption patterns by summarizing total consumption across specified food items. It provides insights into which districts and regions consume the most and least, helping to identify consumption trends and patterns within the dataset.

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

Sample code in R:



```
> # Rename districts and sectors , get codes from appendix of NSSO 68th ROund Data
> district_mapping <- c("15" = "Agra", "11" = "Bulandshahar", "12" = "Aligarh")
> sector_mapping <- c("2" = "URBAN", "1" = "RURAL")
>
> upnew$District <- as.character(upnew$District)
> upnew$Sector <- as.character(upnew$Sector)
> upnew$District <- ifelse(upnew$District %in% names(district_mapping), district_mapping[upnew$District], upnew$District)
> upnew$Sector <- ifelse(upnew$Sector %in% names(sector_mapping), sector_mapping[upnew$Sector], upnew$Sector)
>
>
> # Test for differences in mean consumption between urban and rural
> rural <- upnew %>%
+   filter(Sector == "RURAL") %>%
+   select(total_consumption)
>
> urban <- upnew %>%
+   filter(Sector == "URBAN") %>%
+   select(total_consumption)
>
> mean_rural <- mean(rural$total_consumption)
> mean_urban <- mean(urban$total_consumption)
>
```



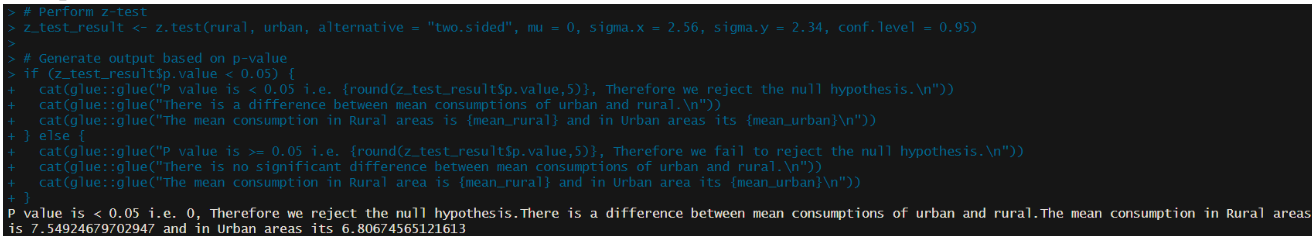
state_1	District	Region	Sector	State_Region	Meals_At_Home
UP	16	5	URBAN	95	60.0000
UP	16	5	URBAN	95	60.0000
UP	Aligarh	5	URBAN	95	60.0000
UP	Agra	5	URBAN	95	60.0000
UP	Agra	5	URBAN	95	60.0000
UP	Agra	5	URBAN	95	60.0000
UP	Agra	5	URBAN	95	60.0000
UP	Agra	5	URBAN	95	60.0000
UP	Agra	5	URBAN	95	60.0000
UP	Agra	5	URBAN	95	60.0000
UP	Agra	5	URBAN	95	60.0000
UP	14	5	URBAN	95	60.0000
UP	14	5	URBAN	95	60.0000

Interpretation:
In the given code, the districts identified by codes "15", "11", and "12" are renamed to "Agra", "Bulandshahar", and "Aligarh", respectively. These mappings are derived from the NSSO 68th Round Data Appendix. Similarly, sectors represented by codes "2" and "1" are renamed to "URBAN" and "RURAL", respectively.

Generally, we try to standardize and humanize district and sector labels in the dataset by mapping numeric codes to meaningful names sourced from the NSSO 68th Round Data Appendix. It ensures clarity and understanding of the data by replacing numeric identifiers with recognizable names, facilitating easier interpretation and analysis.

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

Sample code in R:



```
> # 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"))
+ }
P value is < 0.05 i.e. 0, Therefore we reject the null hypothesis.There is a difference between mean consumptions of urban and rural.The mean consumption in Rural areas
is 7.54924679702947 and in Urban areas its 6.80674565121613
```

Interpretation:
Here we conduct a two-sample z-test to determine if there is a statistically significant difference in mean consumption between rural and urban areas.

Parameters for the z-test include:

- `mu = 0`: Null hypothesis assuming no difference in means.
- `alternative = "two.sided"`: Testing for differences in means in both directions.
- `sigma.x` and `sigma.y`: Standard deviations of `rural` and `urban` data
- `conf.level = 0.95`: Confidence level set to 95%.

Interpretation Based on p-value:

- If `p.value` from the z-test is less than 0.05 (`< 0.05`), reject the null hypothesis.
- If `p.value` is greater than or equal to 0.05 (`>= 0.05`), fail to reject the null hypothesis.

We perform a hypothesis test to assess if there is a significant difference in mean consumption between urban and rural areas based on the `total_consumption` variable in the dataset. In this case we reject the null hypothesis. The test results indicate a statistically significant difference (`p < 0.05`), suggesting that mean consumption levels differ between these two sectors. Specifically, rural areas have a higher mean consumption (`7.549`) compared to urban areas (`6.807`). This analysis provides insights into consumption patterns that may be influenced by geographic factors like urbanization and economic conditions.