A1-R.R

ujjwa

2025-06-12

```
# --- Project Setup and Package Management ---

# Set the base directory for the project
BASE <- "C:\\Users\\ujjwa\\Documents\\VCU\\Pre-Course\\SCMA632\\Assignments\\A1\\R"
setwd(BASE) # Set the working directory to the specified base path
getwd() # Verify the current working directory
```

```
## [1] "C:/Users/ujjwa/Documents/VCU/Pre-Course/SCMA632/Assignments/A1/R"
```

```
# Define a function to install packages if they are not already installed
install <- function(pkg) {
  if (!require(pkg, character.only = T)) { # Check if the package is loaded
    # Install with dependencies if not loaded
    install.packages(pkg, dependencies = T, quiet = T, verbose = F)
  }
}

# Define a function to load packages
load <- function(pkg) {
  library(pkg, character.only = T, quietly = T, verbose = F) # Load the specified package
}

# List of required packages for data manipulation, visualization, and statistical tests
pkgs <- c("dplyr", "readr", "readxl", "tidyr", "ggplot2", "BSDA")
lapply(pkgs, install) # Apply the install function to all packages
```

```
## Loading required package: dplyr

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union

## Loading required package: readr
```

```
## Loading required package: readxl

## Loading required package: tidyr

## Loading required package: ggplot2

## Loading required package: BSDA

## Loading required package: lattice

##
## Attaching package: 'BSDA'

## The following object is masked from 'package:datasets':
##
##     Orange

## [[1]]
## NULL
##
## [[2]]
## NULL
##
## [[3]]
## NULL
##
## [[4]]
## NULL
##
## [[5]]
## NULL
##
## [[6]]
## NULL
```

```
lapply(pkgs, load) # Apply the load function to all packages
```

```
## [[1]]
##  [1] "BSDA"      "lattice"   "ggplot2"   "tidyr"     "readxl"    "readr"
##  [7] "dplyr"     "stats"     "graphics"  "grDevices" "utils"     "datasets"
## [13] "methods"   "base"
##
## [[2]]
##  [1] "BSDA"      "lattice"   "ggplot2"   "tidyr"     "readxl"    "readr"
##  [7] "dplyr"     "stats"     "graphics"  "grDevices" "utils"     "datasets"
## [13] "methods"   "base"
##
## [[3]]
##  [1] "BSDA"      "lattice"   "ggplot2"   "tidyr"     "readxl"    "readr"
##  [7] "dplyr"     "stats"     "graphics"  "grDevices" "utils"     "datasets"
## [13] "methods"   "base"
```

```
##
## [[4]]
##  [1] "BSDA"      "lattice"   "ggplot2"   "tidyr"     "readxl"    "readr"
##  [7] "dplyr"     "stats"     "graphics"  "grDevices" "utils"     "datasets"
## [13] "methods"   "base"
##
## [[5]]
##  [1] "BSDA"      "lattice"   "ggplot2"   "tidyr"     "readxl"    "readr"
##  [7] "dplyr"     "stats"     "graphics"  "grDevices" "utils"     "datasets"
## [13] "methods"   "base"
##
## [[6]]
##  [1] "BSDA"      "lattice"   "ggplot2"   "tidyr"     "readxl"    "readr"
##  [7] "dplyr"     "stats"     "graphics"  "grDevices" "utils"     "datasets"
## [13] "methods"   "base"
```

```
# --- Data Loading and Initial Filtering ---

# Load the main dataset from a CSV file
data <- read.csv('./datasets/NSSO68.csv')
# Filter the data to include only records for Meghalaya state (state code "17")
data_meghalaya <- data %>% filter(state == "17")

# Remove the original large dataset to free up memory
rm(data)

# Print the column names of the filtered Meghalaya dataset
print(names(data_meghalaya))
```

```
##   [1] "slno"                          "grp"
##   [3] "Round_Centre"                  "FSU_number"
##   [5] "Round"                         "Schedule_Number"
##   [7] "Sample"                        "Sector"
##   [9] "state"                         "State_Region"
##  [11] "District"                      "Stratum_Number"
##  [13] "Sub_Stratum"                   "Schedule_type"
##  [15] "Sub_Round"                     "Sub_Sample"
##  [17] "FOD_Sub_Region"                "Hamlet_Group_Sub_Block"
##  [19] "t"                             "X_Stage_Stratum"
##  [21] "HHS_No"                        "Level"
##  [23] "Filler"                        "hhdsz"
##  [25] "NIC_2008"                      "NCO_2004"
##  [27] "HH_type"                       "Religion"
##  [29] "Social_Group"                  "Whether_owns_any_land"
##  [31] "Type_of_land_owned"            "Land_Owned"
##  [33] "Land_Leased_in"                "Otherwise_possessed"
##  [35] "Land_Leased_out"               "Land_Total_possessed"
##  [37] "During_July_June_Cultivated"   "During_July_June_Irrigated"
##  [39] "NSS"                           "NSC"
##  [41] "MLT"                           "land_tt"
##  [43] "Cooking_code"                  "Lighting_code"
##  [45] "Dwelling_unit_code"            "Regular_salary_earner"
##  [47] "Perform_Ceremony"              "Meals_seved_to_non_hhld_members"
##  [49] "Possess_ration_card"           "Type_of_ration_card"
```

```
##  [51] "MPCE_URP"                 "MPCE_MRP"
##  [53] "Person_Srl_No"            "Relation"
##  [55] "Sex"                      "Age"
##  [57] "Marital_Status"           "Education"
##  [59] "Days_Stayed_away"         "No_of_Meals_per_day"
##  [61] "Meals_School"             "Meals_Employer"
##  [63] "Meals_Others"             "Meals_Payment"
##  [65] "Meals_At_Home"            "Item_Code"
##  [67] "Source_Code"              "ricepds_q"
##  [69] "riceos_q"                 "ricetotal_q"
##  [71] "chira_q"                  "khoi_q"
##  [73] "muri_q"                   "ricepro_q"
##  [75] "riceGT_q"                 "Wheatpds_q"
##  [77] "wheatos_q"                "wheattotal_q"
##  [79] "maida_q"                  "suji_q"
##  [81] "sewai_q"                  "bread_q"
##  [83] "wheatp_q"                 "wheatGT_q"
##  [85] "jowarp_q"                 "bajrap_q"
##  [87] "maizep_q"                 "barleyp_q"
##  [89] "milletp_q"                "ragip_q"
##  [91] "cerealot_q"               "cerealtot_q"
##  [93] "cerealsub_q"              "cerealstt_q"
##  [95] "arhar_q"                  "gramdal_q"
##  [97] "gramwholep_q"             "gramGT_q"
##  [99] "moong_q"                  "masur_q"
## [101] "urd_q"                    "peasdal_q"
## [103] "khesari_q"                "otpulse_q"
## [105] "gramp_q"                  "besan_q"
## [107] "pulsep_q"                 "pulsestot_q"
## [109] "pulsestt_q"               "soyabean_q"
## [111] "milk_q"                   "babyfood_q"
## [113] "milkcond_q"               "curd_q"
## [115] "ghee_q"                   "butter_q"
## [117] "icecream_q"               "otmilkp_q"
## [119] "Milktotal_q"              "milkprott_q"
## [121] "vanas_q"                  "musoil_q"
## [123] "gnoil_q"                  "cocooil_q"
## [125] "edioilothr_q"             "edibletotal_q"
## [127] "ediblest_q"               "eggsno_q"
## [129] "fishprawn_q"              "goatmeat_q"
## [131] "beef_q"                   "pork_q"
## [133] "chicken_q"                "othrbirds_q"
## [135] "nonvegtotal_q"            "emftt_q"
## [137] "potato_q"                 "onion_q"
## [139] "tamato_q"                 "brinjal_q"
## [141] "radish_q"                 "carrot_q"
## [143] "palak_q"                  "chillig_q"
## [145] "bhindi_q"                 "parwal_q"
## [147] "cauli_q"                  "cabbage_q"
## [149] "pumpkin_q"                "peas_q"
## [151] "fbeans_q"                 "lemonno_q"
## [153] "otveg_q"                  "vegtt_q"
## [155] "bananano_q"               "jackfruit_q"
## [157] "watermel_q"               "pineaplno_q"
```

```
## [159] "cocono_q"                "cocogno_q"
## [161] "guava_q"                 "sighara_q"
## [163] "orangeno_q"              "papayar_q"
## [165] "mango_q"                 "kharbooz_q"
## [167] "pears_q"                 "berries_q"
## [169] "leechi_q"                "apple_q"
## [171] "grapes_q"                "otfruits_q"
## [173] "fruitstt_q"              "fruitt_total"
## [175] "cocodf_q"                "gnutdf_q"
## [177] "datesdf_q"               "cashewdf_q"
## [179] "walnutdf_q"              "otnutsdf_q"
## [181] "kishmish_q"              "otherdf_q"
## [183] "dryfruitstotal_q"        "dftt_q"
## [185] "sugarpds_q"              "sugaros_q"
## [187] "sugarst_q"               "gur_q"
## [189] "misri_q"                 "honey_q"
## [191] "sugartotal_q"            "sugartt_q"
## [193] "salt_q"                  "ginger_q"
## [195] "garlic_q"                "jeera_q"
## [197] "dhania_q"                "turnmeric_q"
## [199] "blackpepper_q"           "drychilly_q"
## [201] "tamarind_q"              "currypowder_q"
## [203] "oilseeds_q"              "spicesothr_q"
## [205] "spicetot_q"              "spicestotal_q"
## [207] "teacupno_q"              "tealeaf_q"
## [209] "teatotal_q"              "cofeeno_q"
## [211] "coffeepwdr_q"            "cofeetotal_q"
## [213] "ice_q"                   "coldbvrg_q"
## [215] "juice_q"                 "othrbevrg_q"
## [217] "bevergest_q"             "Biscuits_q"
## [219] "preparedsweet_q"         "pickle_q"
## [221] "sauce_jam_q"             "Othrprocessed_q"
## [223] "Beveragestotal_q"        "ricepds_v"
## [225] "riceos_v"                "ricetotal_v"
## [227] "chira_v"                 "khoi_v"
## [229] "muri_v"                  "ricepro_v"
## [231] "riceGT_v"                "Wheatpds_v"
## [233] "wheatos_v"               "wheattotal_v"
## [235] "maida_v"                 "suji_v"
## [237] "sewai_v"                 "bread_v"
## [239] "wheatp_v"                "wheatGT_v"
## [241] "jowarp_v"                "bajrap_v"
## [243] "maizep_v"                "barleyp_v"
## [245] "milletp_v"               "ragip_v"
## [247] "cerealot_v"              "cerealtot_v"
## [249] "cerealsub_v"             "cerealstt_v"
## [251] "arhar_v"                 "gramdal_v"
## [253] "gramwholep_v"            "gramGT_v"
## [255] "moong_v"                 "masur_v"
## [257] "urd_v"                   "peasdal_v"
## [259] "khesari_v"               "otpulse_v"
## [261] "gramp_v"                 "besan_v"
## [263] "pulsep_v"                "pulsestot_v"
## [265] "pulsestt_v"              "soyabean_v"
```

```
## [267] "milk_v"                "babyfood_v"
## [269] "milkcond_v"            "curd_v"
## [271] "ghee_v"                "butter_v"
## [273] "icecream_v"            "otmilkp_v"
## [275] "Milktotal_v"           "milkprott_v"
## [277] "vanas_v"               "musoil_v"
## [279] "gnoil_v"               "cocooil_v"
## [281] "edioilothr_v"          "edibletotal_v"
## [283] "ediblest_v"            "eggsno_v"
## [285] "fishprawn_v"           "goatmeat_v"
## [287] "beef_v"                "pork_v"
## [289] "chicken_v"             "othrbirds_v"
## [291] "nonvegtotal_v"         "emftt_v"
## [293] "potato_v"              "onion_v"
## [295] "tamato_v"              "brinjal_v"
## [297] "radish_v"              "carrot_v"
## [299] "palak_v"               "chillig_v"
## [301] "bhindi_v"              "parwal_v"
## [303] "cauli_v"               "cabbage_v"
## [305] "pumpkin_v"             "peas_v"
## [307] "fbeans_v"              "lemonno_v"
## [309] "otveg_v"               "vegtt_v"
## [311] "bananano_v"            "jackfruit_v"
## [313] "watermel_v"            "pineaplno_v"
## [315] "cocono_v"              "cocogno_v"
## [317] "guava_v"               "sighara_v"
## [319] "orangeno_v"            "papayar_v"
## [321] "mango_v"               "kharbooz_v"
## [323] "pears_v"               "berries_v"
## [325] "leechi_v"              "apple_v"
## [327] "grapes_v"              "otfruits_v"
## [329] "fruitstt_v"            "cocodf_v"
## [331] "gnutdf_v"              "datesdf_v"
## [333] "cashewdf_v"            "walnutdf_v"
## [335] "otnutsdf_v"            "kishmish_v"
## [337] "otherdf_v"             "dryfruitstotal_v"
## [339] "dftt_v"                "sugarpds_v"
## [341] "sugaros_v"             "sugarst_v"
## [343] "gur_v"                 "misri_v"
## [345] "honey_v"               "sugartotal_v"
## [347] "sugartt_v"             "salt_v"
## [349] "ginger_v"              "garlic_v"
## [351] "jeera_v"               "dhania_v"
## [353] "turnmeric_v"           "blackpepper_v"
## [355] "drychilly_v"           "tamarind_v"
## [357] "currypowder_v"         "oilseeds_v"
## [359] "spicesothr_v"          "spicetot_v"
## [361] "spicestotal_v"         "teacupno_v"
## [363] "tealeaf_v"             "teatotal_v"
## [365] "cofeeno_v"             "coffeepwdr_v"
## [367] "cofeetotal_v"          "ice_v"
## [369] "coldbvrg_v"            "juice_v"
## [371] "othrbevrg_v"           "bevergest_v"
## [373] "Biscuits_v"            "preparedsweet_v"
```

```
## [375] "pickle_v"                    "sauce_jam_v"
## [377] "Othrprocessed_v"             "Beveragestotal_v"
## [379] "foodtotal_v"                 "foodtotal_q"
## [381] "state_1"                     "Region"
## [383] "fruits_df_tt_v"              "fv_tot"
```

```r
# Print the first few rows of the Meghalaya dataset for a quick preview
print(head(data_meghalaya))
```

```
##    slno      grp Round_Centre FSU_number Round Schedule_Number Sample Sector
## 1 17686 4.46E+31            1      44610    68              10      1      2
## 2 17687 4.46E+31            1      44610    68              10      1      2
## 3 17688 4.46E+31            1      44610    68              10      1      2
## 4 17689 4.46E+31            1      44610    68              10      1      2
## 5 17690 4.46E+31            1      44610    68              10      1      2
## 6 17691 4.46E+31            1      44610    68              10      1      2
##   state State_Region District Stratum_Number Sub_Stratum Schedule_type
## 1    17          171        6              6           6             1
## 2    17          171        6              6           6             1
## 3    17          171        6              6           6             1
## 4    17          171        6              6           6             1
## 5    17          171        6              6           6             1
## 6    17          171        6              6           6             1
##   Sub_Round Sub_Sample FOD_Sub_Region Hamlet_Group_Sub_Block        t
## 1         3          2           1710                      1 1.01e+13
## 2         3          2           1710                      1 1.02e+13
## 3         3          2           1710                      1 2.01e+13
## 4         3          2           1710                      1 2.02e+13
## 5         3          2           1710                      1 2.03e+13
## 6         3          2           1710                      1 2.04e+13
##   X_Stage_Stratum HHS_No Level Filler hhdsz NIC_2008 NCO_2004 HH_type Religion
## 1               1      1     5      0     3    84119      419       2        3
## 2               1      2     5      0     5    32111      121       1        2
## 3               2      1     5      0     4    47221      341       2        3
## 4               2      2     5      0     3    47110      121       1        1
## 5               2      3     5      0     3    47594      121       1        2
## 6               2      4     5      0     6    85212      232       3        1
##   Social_Group Whether_owns_any_land Type_of_land_owned Land_Owned
## 1            1                     1                 1          4
## 2            9                     2                NA         NA
## 3            1                     2                NA         NA
## 4            9                     2                NA         NA
## 5            9                     2                NA         NA
## 6            9                     2                NA         NA
##   Land_Leased_in Otherwise_possessed Land_Leased_out Land_Total_possessed
## 1             NA                  NA             NA                    4
## 2             NA                  NA             NA                   NA
## 3              3                  NA             NA                    3
## 4              3                  NA             NA                    3
## 5              3                  NA             NA                    3
## 6              4                  NA             NA                    4
##   During_July_June_Cultivated During_July_June_Irrigated NSS NSC    MLT land_tt
## 1                          NA                         NA   2   4 121200    0.04
## 2                          NA                         NA   2   4 121200      NA
```

```
## 3                          NA                          NA   2   4  44188    0.03
## 4                          NA                          NA   2   4  44188    0.03
## 5                          NA                          NA   2   4  44188    0.03
## 6                          NA                          NA   2   4  44188    0.04
##   Cooking_code Lighting_code Dwelling_unit_code Regular_salary_earner
## 1            3             5                 1                     1
## 2            3             5                 2                     2
## 3            3             5                 2                     1
## 4            3             5                 2                     2
## 5            3             5                 2                     2
## 6            3             5                 2                     2
##   Perform_Ceremony Meals_seved_to_non_hhld_members Possess_ration_card
## 1                2                               0                   1
## 2                2                              NA                   1
## 3                2                               0                   1
## 4                2                               0                   1
## 5                2                               0                   1
## 6                2                               0                   1
##   Type_of_ration_card MPCE_URP MPCE_MRP Person_Srl_No Relation Sex Age
## 1                   3  3512.33  3186.12             1        1   2  47
## 2                   3  4178.00  3938.99             1        1   1  56
## 3                   3  2426.00  2370.18             1        1   1  70
## 4                   3  2095.00  2203.08             1        1   1  58
## 5                   3  2260.67  2372.08             1        1   1  42
## 6                   3  2146.50  2024.83             1        1   2  45
##   Marital_Status Education Days_Stayed_away No_of_Meals_per_day Meals_School
## 1              3        10                0                   2           0
## 2              2        10                0                   2           0
## 3              2         5                0                   2           0
## 4              2        10                0                   2           0
## 5              2        12                0                   2           0
## 6              3        12                0                   2           0
##   Meals_Employer Meals_Others Meals_Payment Meals_At_Home Item_Code Source_Code
## 1              0            0             0           60       101           1
## 2              0            0             0           60       101           1
## 3              0            0             0           60       101           1
## 4              0            0             0           60       101           1
## 5              0            0             0           60       101           1
## 6              0            0             0           60       101           1
##   ricepds_q riceos_q ricetotal_q chira_q khoi_q muri_q ricepro_q  riceGT_q
## 1  4.000000 6.000000   10.000000       0      0      0         0 10.000000
## 2  4.000000 3.600000    7.600000       0      0      0         0  7.600000
## 3  3.000000 5.000000    8.000000       0      0      0         0  8.000000
## 4  3.333333 1.666667    5.000000       0      0      0         0  5.000000
## 5  3.000000 6.000000    9.000000       0      0      0         0  9.000000
## 6  3.333333 4.250000    7.583333       0      0      0         0  7.583333
##   Wheatpds_q wheatos_q wheattotal_q   maida_q suji_q sewai_q    bread_q
## 1          0 1.6666667    1.6666667 0.0000000      0     0.0 0.66666667
## 2          0 1.2000000    1.2000000 0.7600000      0     0.8 0.80000000
## 3          0 0.7500000    0.7500000 0.0000000      0     0.0 0.25000000
## 4          0 4.0000000    4.0000000 1.0000000      0     0.0 0.00000000
## 5          0 0.6666667    0.6666667 0.0000000      0     0.0 1.00000000
## 6          0 0.5000000    0.5000000 0.2666667      0     0.0 0.08333333
##   wheatp_q wheatGT_q jowarp_q bajrap_q maizep_q barleyp_q milletp_q ragip_q
```

```
## 1        0  2.333333        0        0        0         0         0       0
## 2        0  3.560000        0        0        0         0         0       0
## 3        0  1.000000        0        0        0         0         0       0
## 4        0  5.000000        0        0        0         0         0       0
## 5        0  1.666667        0        0        0         0         0       0
## 6        0  0.850000        0        0        0         0         0       0
##   cerealot_q cerealtot_q cerealsub_q cerealstt_q   arhar_q gramdal_q
## 1          0   12.333333           0   12.333333 0.0000000 0.0000000
## 2          0   11.160000           0   11.160000 0.0000000 0.4000000
## 3          0    9.000000           0    9.000000 0.0000000 0.0000000
## 4          0   10.000000           0   10.000000 0.0000000 0.0000000
## 5          0   10.666667           0   10.666667 0.1666667 0.1666667
## 6          0    8.433333           0    8.433333 0.3333333 0.0000000
##   gramwholep_q  gramGT_q   moong_q   masur_q     urd_q peasdal_q khesari_q
## 1            0 0.0000000 1.0000000 0.6666667 0.0000000         0         0
## 2            0 0.4000000 0.2000000 0.5200000 0.0000000         0         0
## 3            0 0.0000000 0.2500000 0.5000000 0.0000000         0         0
## 4            0 0.0000000 0.3333333 0.6666667 0.3333333         0         0
## 5            0 0.1666667 0.0000000 0.3333333 0.0000000         0         0
## 6            0 0.0000000 0.1666667 0.3333333 0.1666667         0         0
##   otpulse_q gramp_q   besan_q pulsep_q pulsestot_q pulsestt_q soyabean_q milk_q
## 1         0       0 0.6666667        0   2.3333333  2.3333333         NA   5.20
## 2         0       0 0.0000000        0   1.1200000  1.1200000         NA   6.24
## 3         0       0 0.1250000        0   0.8750000  0.8750000         NA   7.80
## 4         0       0 0.0000000        0   1.3333333  1.3333333         NA   5.20
## 5         0       0 0.0000000        0   0.6666667  0.6666667         NA   2.08
## 6         0       0 0.0000000        0   1.0000000  1.0000000         NA   5.20
##   babyfood_q milkcond_q curd_q ghee_q butter_q icecream_q otmilkp_q Milktotal_q
## 1          0          0      0   0.05      0.1          0         0           0
## 2          0          0      0   0.00      0.1          0         0           0
## 3          0          0      0   0.25      0.0          0         0           0
## 4          0          0      0   0.00      0.0          0         0           0
## 5          0          0      0   0.00      0.0          0         0           0
## 6          0          0      0   0.00      0.0          0         0           0
##   milkprott_q vanas_q  musoil_q gnoil_q cocooil_q edioilothr_q edibletotal_q
## 1        5.35       0 0.6666667       0         0            0     1.0000000
## 2        6.34       0 0.8000000       0         0            0     1.2000000
## 3        8.05       0 0.5000000       0         0            0     1.0000000
## 4        5.20       0 0.3333333       0         0            0     0.6666667
## 5        2.08       0 0.6666667       0         0            0     1.0000000
## 6        5.20       0 0.3333333       0         0            0     0.5000000
##   ediblest_q  eggsno_q fishprawn_q goatmeat_q    beef_q pork_q chicken_q
## 1  0.6666667 0.0002750   0.6666667        0.0 1.0000000      0 0.3333333
## 2  0.8000000 0.0006600   1.0000000        0.6 0.4500000      0 0.0000000
## 3  0.5000000 0.0004125   0.0000000        0.5 0.0000000      0 0.0000000
## 4  0.3333333 0.0005500   0.3333333        0.0 0.0000000      0 0.6666667
## 5  0.6666667 0.0000000   0.1666667        0.0 0.3333333      0 0.0000000
## 6  0.3333333 0.0001375   0.1666667        0.0 0.0000000      0 0.1666667
##   othrbirds_q nonvegtotal_q   emftt_q  potato_q   onion_q  tamato_q brinjal_q
## 1           0             0 2.0002750 1.0000000 0.6666667 1.0000000 0.3333333
## 2           0             0 2.0506600 1.0000000 0.8000000 1.2000000 0.0000000
## 3           0             0 0.5004125 1.2500000 0.5000000 0.7500000 0.2500000
## 4           0             0 1.0005500 1.0000000 0.6666667 0.3333333 0.3333333
## 5           0             0 0.5000000 0.3333333 0.6666667 1.0000000 0.6666667
```

```
## 6          0             0 0.3334708 1.0000000 0.3333333 0.5000000 0.0000000
##    radish_q  carrot_q   palak_q  chillig_q bhindi_q parwal_q   cauli_q
## 1 0.0000000 0.0000000 1.0000000 0.16666667        0        0 0.6666667
## 2 0.4000000 0.6000000 0.8000000 0.00000000        0        0 0.4000000
## 3 0.0000000 0.2500000 0.5000000 0.12500000        0        0 0.7500000
## 4 0.0000000 0.3333333 0.6666667 0.10000000        0        0 0.3333333
## 5 0.0000000 0.3333333 0.0000000 0.00000000        0        0 0.6666667
## 6 0.1666667 0.3333333 0.5000000 0.08333333        0        0 0.3333333
##   cabbage_q pumpkin_q    peas_q  fbeans_q lemonno_q otveg_q  vegtt_q
## 1 0.0000000         0 0.3333333 0.3333333     1e-04       1 6.500100
## 2 0.6000000         0 0.4000000 0.8000000     0e+00       0 7.000000
## 3 0.5000000         0 0.0000000 0.2500000     0e+00       0 5.125000
## 4 0.3333333         0 0.3333333 0.3333333     0e+00       0 4.766667
## 5 0.0000000         0 0.0000000 0.3333333     0e+00       0 4.000000
## 6 0.3333333         0 0.0000000 0.3333333     0e+00       0 3.916667
##    bananano_q jackfruit_q watermel_q pineaplno_q cocono_q cocogno_q guava_q
## 1 0.000600000           0          0    0.001500        0         0       0
## 2 0.000280000           0          0    0.001500        0         0       0
## 3 0.000300000           0          0    0.001125        0         0       0
## 4 0.000266667           0          0    0.000500        0         0       0
## 5 0.000400000           0          0    0.001500        0         0       0
## 6 0.000200000           0          0    0.000500        0         0       0
##   sighara_q orangeno_q papayar_q mango_q kharbooz_q pears_q berries_q leechi_q
## 1         0          0 0.0000000       0          0       0         0        0
## 2         0          0 0.6000000       0          0       0         0        0
## 3         0          0 0.0000000       0          0       0         0        0
## 4         0          0 0.0000000       0          0       0         0        0
## 5         0          0 0.6666667       0          0       0         0        0
## 6         0          0 0.0000000       0          0       0         0        0
##     apple_q  grapes_q otfruits_q fruitstt_q fruitt_total cocodf_q gnutdf_q
## 1 0.0000000 0.3333333          0          0     89.00000        0      0.0
## 2 0.2000000 0.2000000          0          0    113.40000        0      0.2
## 3 0.0000000 0.0000000          0          0     39.87500        0      0.0
## 4 0.3333333 0.0000000          0          0     66.00000        0      0.0
## 5 0.0000000 0.0000000          0          0     55.66667        0      0.0
## 6 0.0000000 0.2166667          0          0     35.16667        0      0.0
##   datesdf_q cashewdf_q walnutdf_q otnutsdf_q kishmish_q otherdf_q
## 1         0          0  0.0000000          0          0         0
## 2         0          0  0.1000000          0          0         0
## 3         0          0  0.0000000          0          0         0
## 4         0          0  0.0000000          0          0         0
## 5         0          0  0.1666667          0          0         0
## 6         0          0  0.0000000          0          0         0
##   dryfruitstotal_q    dftt_q sugarpds_q sugaros_q sugarst_q      gur_q
## 1                0 0.0000000  0.5000000 0.0000000 0.5000000 0.06666667
## 2                0 0.3000000  0.0000000 1.0000000 1.0000000 0.00000000
## 3                0 0.0000000  0.5000000 0.0000000 0.5000000 0.10000000
## 4                0 0.0000000  0.0000000 0.3333333 0.3333333 0.06666667
## 5                0 0.1666667  0.2666667 0.4333333 0.7000000 0.00000000
## 6                0 0.0000000  0.3333333 0.3333333 0.6666667 0.25000000
##      misri_q honey_q sugartotal_q sugartt_q    salt_q    ginger_q garlic_q
## 1 0.00000000       0    0.6666667 0.5666667 0.1000000 0.000050000 1.00e-04
## 2 0.00000000       0    1.3200000 1.0000000 0.3200000 0.000100000 1.60e-04
## 3 0.00000000       0    0.7000000 0.6000000 0.1000000 0.000037500 6.25e-05
```

```
## 4 0.00000000       0    0.6000000 0.4000000 0.2000000 0.000083300 8.33e-05
## 5 0.00000000       0    0.9666667 0.7000000 0.2666667 0.000166667 8.33e-05
## 6 0.03333333       0    1.0833333 0.9500000 0.1333333 0.000050000 5.00e-05
##    jeera_q dhania_q turnmeric_q blackpepper_q drychilly_q tamarind_q
## 1 3.33e-05 0.00e+00    5.00e-05      3.33e-05    0.00e+00   0.00e+00
## 2 3.00e-05 0.00e+00    2.00e-05      2.00e-05    0.00e+00   0.00e+00
## 3 2.50e-05 3.75e-05    0.00e+00      0.00e+00    2.50e-05   0.00e+00
## 4 0.00e+00 0.00e+00    1.67e-05      1.67e-05    8.33e-05   0.00e+00
## 5 0.00e+00 0.00e+00    3.33e-05      0.00e+00    5.00e-05   0.00e+00
## 6 8.33e-06 1.67e-05    8.33e-06      0.00e+00    4.17e-05   8.33e-06
##   currypowder_q oilseeds_q spicesothr_q  spicetot_q spicestotal_q teacupno_q
## 1             0          0     5.00e-05 0.000316667   0.000316667          0
## 2             0          0     3.00e-05 0.000370000   0.000360000          0
## 3             0          0     3.75e-05 0.000225000   0.000225000          0
## 4             0          0     0.00e+00 0.000283333   0.000283333          0
## 5             0          0     0.00e+00 0.000333333   0.000333333          0
## 6             0          0     2.50e-05 0.000225000   0.000208333          0
##   tealeaf_q teatotal_q cofeeno_q coffeepwdr_q cofeetotal_q ice_q coldbvrg_q
## 1         0          0         0            0            0     0          0
## 2         0          0         0            0            0     0          0
## 3         0          0         0            0            0     0          0
## 4         0          0         0            0            0     0          0
## 5         0          0         0            0            0     0          0
## 6         0          0         0            0            0     0          0
##   juice_q othrbevrg_q bevergest_q Biscuits_q preparedsweet_q pickle_q
## 1       0           0           0          0              0 3.33e-05
## 2       0           0           0          0              0 5.00e-05
## 3       0           0           0          0              0 6.25e-05
## 4       0           0           0          0              0 5.00e-05
## 5       0           0           0          0              0 0.00e+00
## 6       0           0           0          0              0 0.00e+00
##   sauce_jam_q Othrprocessed_q Beveragestotal_q ricepds_v riceos_v ricetotal_v
## 1           0               0        3.33e-05  44.00000 140.6667   184.66667
## 2           0               0        5.00e-05  44.00000  90.0000   134.00000
## 3           0               0        6.25e-05  33.00000 110.0000   143.00000
## 4           0               0        5.00e-05  36.66667  40.0000    76.66667
## 5           0               0        0.00e+00  33.00000 150.0000   183.00000
## 6           0               0        0.00e+00  35.00000  85.0000   120.00000
##   chira_v khoi_v muri_v ricepro_v  riceGT_v Wheatpds_v wheatos_v wheattotal_v
## 1       0      0      0         0 184.66667          0 23.333333    23.333333
## 2       0      0      0         0 134.00000          0 17.600000    17.600000
## 3       0      0      0         0 143.00000          0 12.000000    12.000000
## 4       0      0      0         0  76.66667          0 64.000000    64.000000
## 5       0      0      0         0 183.00000          0 10.666667    10.666667
## 6       0      0      0         0 120.00000          0  6.666667     6.666667
##    maida_v suji_v sewai_v   bread_v wheatp_v wheatGT_v jowarp_v bajrap_v
## 1  0.00000      0     0.0 43.333333        0  66.66667        0        0
## 2 18.40000      0    16.8 32.000000        0  84.80000        0        0
## 3  0.00000      0     0.0 18.750000        0  30.75000        0        0
## 4 19.33333      0     0.0  0.000000        0  83.33333        0        0
## 5  0.00000      0     0.0 43.333333        0  54.00000        0        0
## 6  7.00000      0     0.0  6.666667        0  20.33333        0        0
##   maizep_v barleyp_v milletp_v ragip_v cerealot_v cerealtot_v cerealsub_v
## 1        0         0         0       0          0    251.3333           0
```

```
## 2        0         0         0       0          0    218.8000           0
## 3        0         0         0       0          0    173.7500           0
## 4        0         0         0       0          0    160.0000           0
## 5        0         0         0       0          0    237.0000           0
## 6        0         0         0       0          0    140.3333           0
##   cerealstt_v   arhar_v gramdal_v gramwholep_v  gramGT_v   moong_v  masur_v
## 1    251.3333  0.000000  0.000000            0  0.000000 41.333333 36.00000
## 2    218.8000  0.000000 15.600000            0 15.600000 12.600000 27.20000
## 3    173.7500  0.000000  0.000000            0  0.000000 13.500000 22.50000
## 4    160.0000  0.000000  0.000000            0  0.000000 15.333333 29.33333
## 5    237.0000  9.333333  8.666667            0  8.666667  0.000000 16.00000
## 6    140.3333 14.666667  0.000000            0  0.000000  6.666667 19.33333
##       urd_v peasdal_v khesari_v otpulse_v gramp_v  besan_v pulsep_v pulsestot_v
## 1  0.000000         0         0         0       0 29.33333        0   106.66667
## 2  0.000000         0         0         0       0  0.00000        0    55.40000
## 3  0.000000         0         0         0       0  6.00000        0    42.00000
## 4 17.333333         0         0         0       0  0.00000        0    62.00000
## 5  0.000000         0         0         0       0  0.00000        0    34.00000
## 6  8.666667         0         0         0       0  0.00000        0    49.33333
##   pulsestt_v soyabean_v   milk_v babyfood_v milkcond_v curd_v ghee_v butter_v
## 1  106.66667         NA 130.0000          0          0      0      9       30
## 2   55.40000         NA 184.0800          0          0      0      0       30
## 3   42.00000         NA 195.0000          0          0      0     30        0
## 4   62.00000         NA 104.0000          0          0      0      0        0
## 5   34.00000         NA  52.0000          0          0      0      0        0
## 6   49.33333         NA 112.6667          0          0      0      0        0
##   icecream_v otmilkp_v Milktotal_v milkprott_v vanas_v musoil_v gnoil_v
## 1          0         0    164.0000    169.0000       0       60       0
## 2          0         0    207.0000    214.0800       0       72       0
## 3          0         0    217.5000    225.0000       0       45       0
## 4          0         0    100.0000    104.0000       0       30       0
## 5          0         0     50.0000     52.0000       0       60       0
## 6          0         0    108.3333    112.6667       0       30       0
##   cocooil_v edioilothr_v edibletotal_v ediblest_v eggsno_v fishprawn_v
## 1         0            0      88.33333         60     1.10   113.33333
## 2         0            0     106.00000         72     2.75   130.00000
## 3         0            0      85.00000         45     1.65     0.00000
## 4         0            0      61.66667         30     2.20    46.66667
## 5         0            0      91.66667         60     0.00    20.00000
## 6         0            0      45.83333         30     0.55    23.33333
##   goatmeat_v    beef_v pork_v chicken_v othrbirds_v nonvegtotal_v   emftt_v
## 1          0 160.00000      0  66.66667           0             0 341.10000
## 2        150  72.00000      0   0.00000           0             0 354.75000
## 3        130   0.00000      0   0.00000           0             0 131.65000
## 4          0   0.00000      0 120.00000           0             0 168.86667
## 5          0  53.33333      0   0.00000           0             0  73.33333
## 6          0   0.00000      0  33.33333           0             0  57.21667
##    potato_v  onion_v  tamato_v brinjal_v radish_v carrot_v  palak_v chillig_v
## 1 15.000000 10.00000 20.000000  6.666667      0.0     0.00 15.33333  6.666667
## 2 14.000000 12.00000 24.000000  0.000000     12.0     9.00 24.00000  0.000000
## 3 18.750000  7.50000 17.500000  7.500000      0.0     8.75 13.75000 10.000000
## 4 15.000000 13.33333  8.333333 11.666667      0.0     5.00 13.33333  5.000000
## 5  3.333333 10.00000 20.000000 20.000000      0.0     5.00  0.00000  0.000000
## 6 11.333333  5.00000  7.500000  0.000000      2.5     4.00 11.66667  4.166667
```

```
##   bhindi_v parwal_v   cauli_v cabbage_v pumpkin_v   peas_v fbeans_v lemonno_v
## 1        0        0 16.666667  0.000000         0 15.00000 13.33333       0.5
## 2        0        0 10.000000 12.000000         0 16.00000 28.00000       0.0
## 3        0        0 18.750000  7.500000         0  0.00000 10.00000       0.0
## 4        0        0  8.333333  6.666667         0 13.33333 16.66667       0.0
## 5        0        0 20.000000  0.000000         0  0.00000 13.33333       0.0
## 6        0        0  8.333333  6.666667         0  0.00000 11.66667       0.0
##   otveg_v   vegtt_v bananano_v jackfruit_v watermel_v pineaplno_v cocono_v
## 1      20 139.16667   4.000000           0          0      45.000        0
## 2       0 161.00000   1.800000           0          0      42.000        0
## 3       0 120.00000   1.750000           0          0      28.125        0
## 4       0 116.66667   1.833333           0          0      17.500        0
## 5       0  91.66667   2.333333           0          0      30.000        0
## 6       0  72.83333   1.166667           0          0      14.000        0
##   cocogno_v guava_v sighara_v orangeno_v papayar_v mango_v kharbooz_v pears_v
## 1         0       0         0          0   0.00000       0          0       0
## 2         0       0         0          0   5.60000       0          0       0
## 3         0       0         0          0   0.00000       0          0       0
## 4         0       0         0          0   0.00000       0          0       0
## 5         0       0         0          0  23.33333       0          0       0
## 6         0       0         0          0   0.00000       0          0       0
##   berries_v leechi_v  apple_v grapes_v otfruits_v fruitstt_v cocodf_v gnutdf_v
## 1         0        0  0.00000       40          0          0        0        0
## 2         0        0 36.00000       28          0          0        0       13
## 3         0        0  0.00000        0         10          0        0        0
## 4         0        0 46.66667        0          0          0        0        0
## 5         0        0  0.00000        0          0          0        0        0
## 6         0        0  0.00000       20          0          0        0        0
##   datesdf_v cashewdf_v walnutdf_v otnutsdf_v kishmish_v otherdf_v
## 1         0          0    0.00000          0          0         0
## 2         0          0   32.00000          0          0         0
## 3         0          0    0.00000          0          0         0
## 4         0          0    0.00000          0          0         0
## 5         0          0   56.33333          0          0         0
## 6         0          0    0.00000          0          0         0
##   dryfruitstotal_v   dftt_v sugarpds_v sugaros_v sugarst_v     gur_v misri_v
## 1                0  0.00000   8.000000   0.00000   8.00000  2.333333       0
## 2                0 45.00000   0.000000  36.00000  36.00000  0.000000       0
## 3                0  0.00000   6.000000   0.00000   6.00000  2.500000       0
## 4                0  0.00000   0.000000  18.66667  18.66667  2.000000       0
## 5                0 56.33333   4.000000  16.00000  20.00000  0.000000       0
## 6                0  0.00000   5.333333  12.00000  17.33333 18.000000       1
##   honey_v sugartotal_v sugartt_v   salt_v    ginger_v    garlic_v     jeera_v
## 1       0     11.66667  10.33333 1.333333 0.005000000 0.008333333 0.006666667
## 2       0     39.20000  36.00000 3.200000 0.011000000 0.021400000 0.004000000
## 3       0     10.00000   8.50000 1.500000 0.005000000 0.007500000 0.003250000
## 4       0     24.00000  20.66667 3.333333 0.008333333 0.006666667 0.000000000
## 5       0     23.66667  20.00000 3.666667 0.016666667 0.010000000 0.000000000
## 6       0     38.00000  36.33333 1.666667 0.004166667 0.003333333 0.001666667
##      dhania_v turnmeric_v blackpepper_v drychilly_v  tamarind_v currypowder_v
## 1 0.000000000 0.005000000   0.002666667 0.000000000 0.000000000             0
## 2 0.000000000 0.006000000   0.003000000 0.000000000 0.000000000             0
## 3 0.003750000 0.000000000   0.000000000 0.005000000 0.000000000             0
## 4 0.000000000 0.003333333   0.005000000 0.010000000 0.000000000             0
```

```
## 5 0.000000000 0.006666667   0.000000000 0.006666667 0.000000000             0
## 6 0.001666667 0.000833333   0.000000000 0.004166667 0.001666667             0
##   oilseeds_v spicesothr_v spicetot_v spicestotal_v teacupno_v tealeaf_v
## 1          0  0.002333333 0.03000000    0.03000000          0         0
## 2          0  0.004000000 0.05300000    0.04940000          0         0
## 3          0  0.005000000 0.02950000    0.02950000          0         0
## 4          0  0.000000000 0.03333333    0.03333333          0         0
## 5          0  0.000000000 0.04000000    0.04000000          0         0
## 6          0  0.002500000 0.02250000    0.02000000          0         0
##   teatotal_v cofeeno_v coffeepwdr_v cofeetotal_v ice_v coldbvrg_v juice_v
## 1          0         0            0            0     0          0       0
## 2          0         0            0            0     0          0       0
## 3          0         0            0            0     0          0       0
## 4          0         0            0            0     0          0       0
## 5          0         0            0            0     0          0       0
## 6          0         0            0            0     0          0       0
##   othrbevrg_v bevergest_v Biscuits_v preparedsweet_v    pickle_v sauce_jam_v
## 1           0           0   40.00000        0.000000 0.020000000           0
## 2           0           0   18.00000       12.000000 0.014000000           0
## 3           0           0    0.00000       12.500000 0.011250000           0
## 4           0           0   23.33333       20.000000 0.006666667           0
## 5           0           0   40.00000        3.333333 0.000000000           0
## 6           0           0   25.00000       33.333333 0.000000000           0
##   Othrprocessed_v Beveragestotal_v foodtotal_v foodtotal_q state_1 Region
## 1         0.00000         40.02000   1118.9833    29.85073     MEG      1
## 2         0.00000         30.01400   1190.2934    30.09107     MEG      1
## 3         0.00000         12.51125    759.9407    24.75070     MEG      1
## 4         0.00000         43.34000    708.9067    23.23422     MEG      1
## 5        43.33333         86.66667    714.7067    19.71367     MEG      1
## 6         0.00000         58.33333    558.7367    20.30035     MEG      1
##   fruits_df_tt_v   fv_tot
## 1       89.00000 228.1667
## 2      158.40000 319.4000
## 3       39.87500 159.8750
## 4       66.00000 182.6667
## 5      112.00000 203.6667
## 6       35.16667 108.0000
```

```
# Print the dimensions (number of rows and columns) of the Meghalaya dataset
print(dim(data_meghalaya))
```

```
## [1] 1259  384
```

```
# Save the filtered Meghalaya data to a new CSV file for future use
write.csv(data_meghalaya, './datasets/meghalaya_NSSO68.csv', row.names = FALSE) # Prevent writing row n

# Select a subset of relevant columns for focused analysis
data_meghalaya <- data_meghalaya %>% select(
  state,
  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
)

# --- Handling Missing Values (NA) ---

# Check which columns (if any) still have missing values (NA) before imputation
print(colSums(is.na(data_meghalaya)) > 0)
```

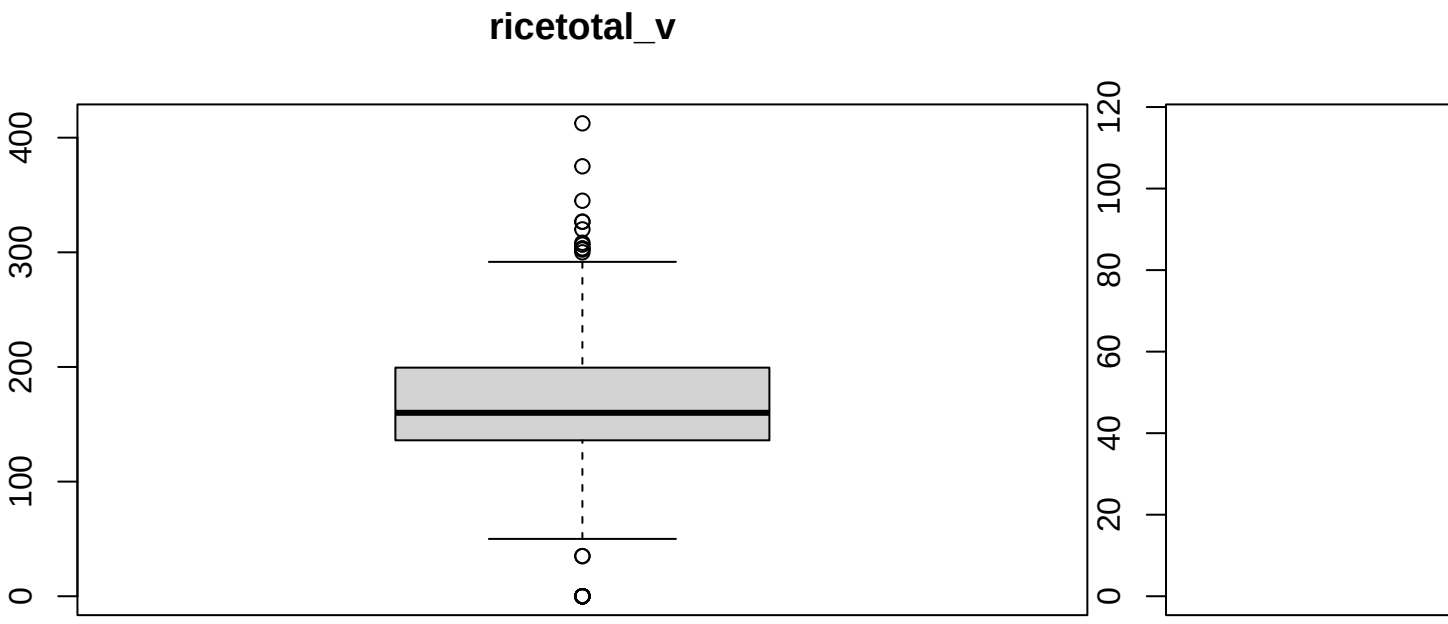
```
##               state             state_1            District              Region
##               FALSE               FALSE               FALSE               FALSE
##              Sector        State_Region       Meals_At_Home         ricetotal_v
##               FALSE               FALSE                TRUE               FALSE
##        wheattotal_v         Milktotal_v         pulsestot_v       nonvegtotal_v
##               FALSE               FALSE               FALSE               FALSE
##          fruitstt_v No_of_Meals_per_day
##               FALSE                TRUE
```

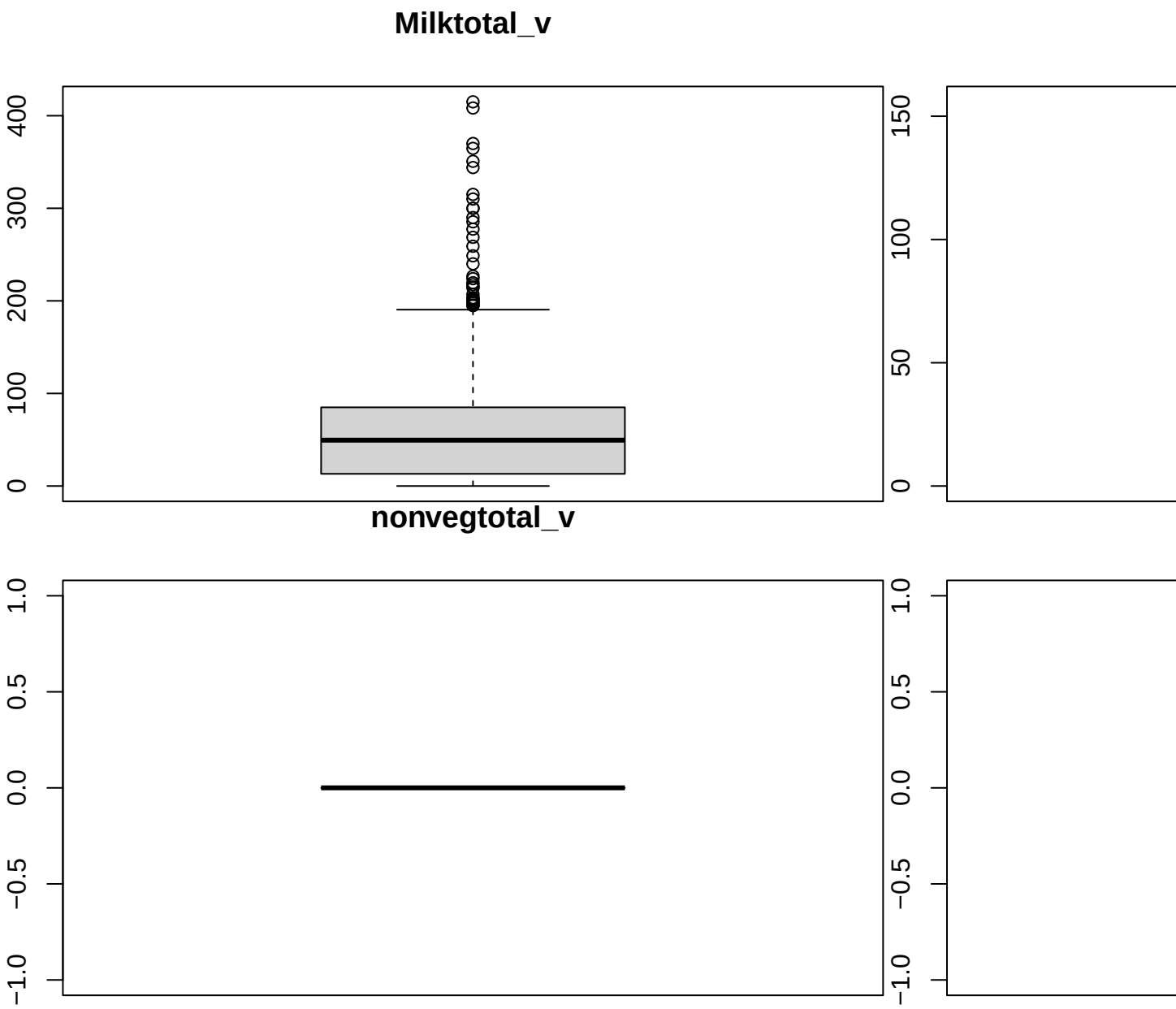
```
# Define a function to impute missing values with the mean of the column
impute <- function(col) {
  if(any(is.na(col))) { # Check if there are any NAs in the column
    col[is.na(col)] <- mean(col, na.rm = T) # Replace NAs with the mean, ignoring NAs in mean calculati
  }
  return(col) # Return the column with imputed values
}

# Apply the imputation function to specific columns identified to have NAs
data_meghalaya$Meals_At_Home <- impute(data_meghalaya$Meals_At_Home)
data_meghalaya$No_of_Meals_per_day <- impute(data_meghalaya$No_of_Meals_per_day)

# Re-check for missing values after imputation to confirm removal
print(colSums(is.na(data_meghalaya)) > 0)
```

```
##               state             state_1            District              Region
##               FALSE               FALSE               FALSE               FALSE
##              Sector        State_Region       Meals_At_Home         ricetotal_v
##               FALSE               FALSE               FALSE               FALSE
##        wheattotal_v         Milktotal_v         pulsestot_v       nonvegtotal_v
##               FALSE               FALSE               FALSE               FALSE
##          fruitstt_v No_of_Meals_per_day
##               FALSE               FALSE
```

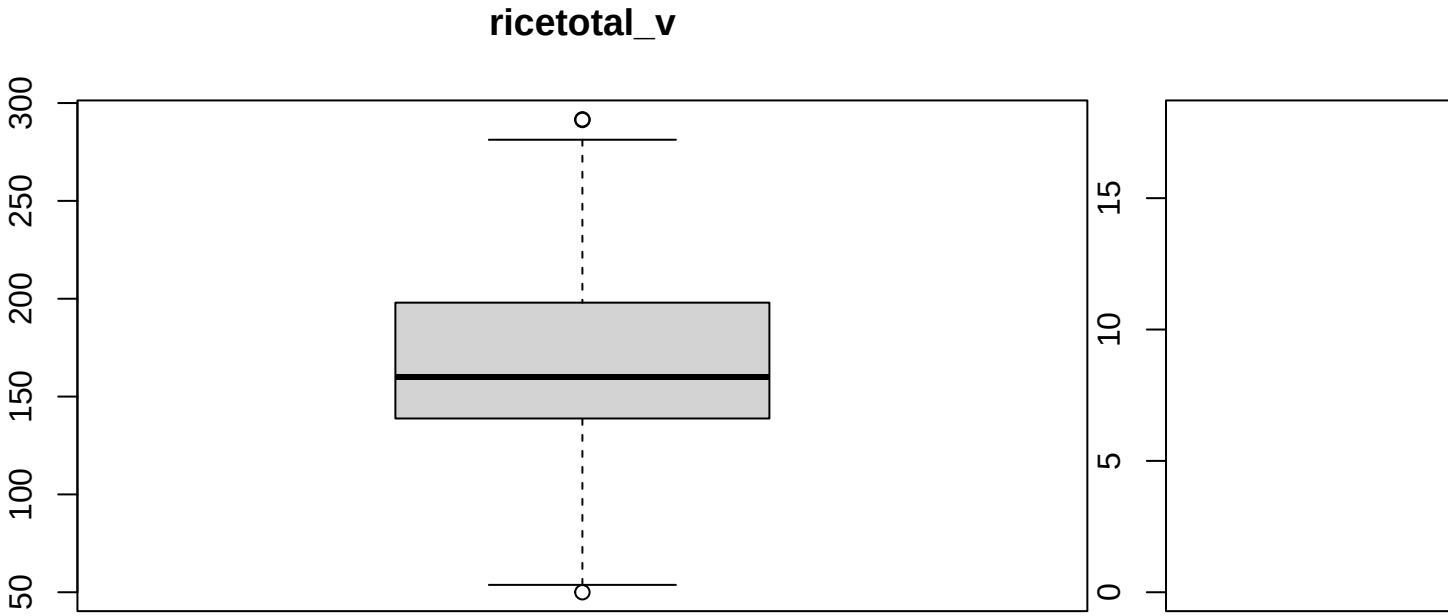
```
# --- Outlier Removal ---

# Define a function to remove outliers using the Interquartile Range (IQR) method
```

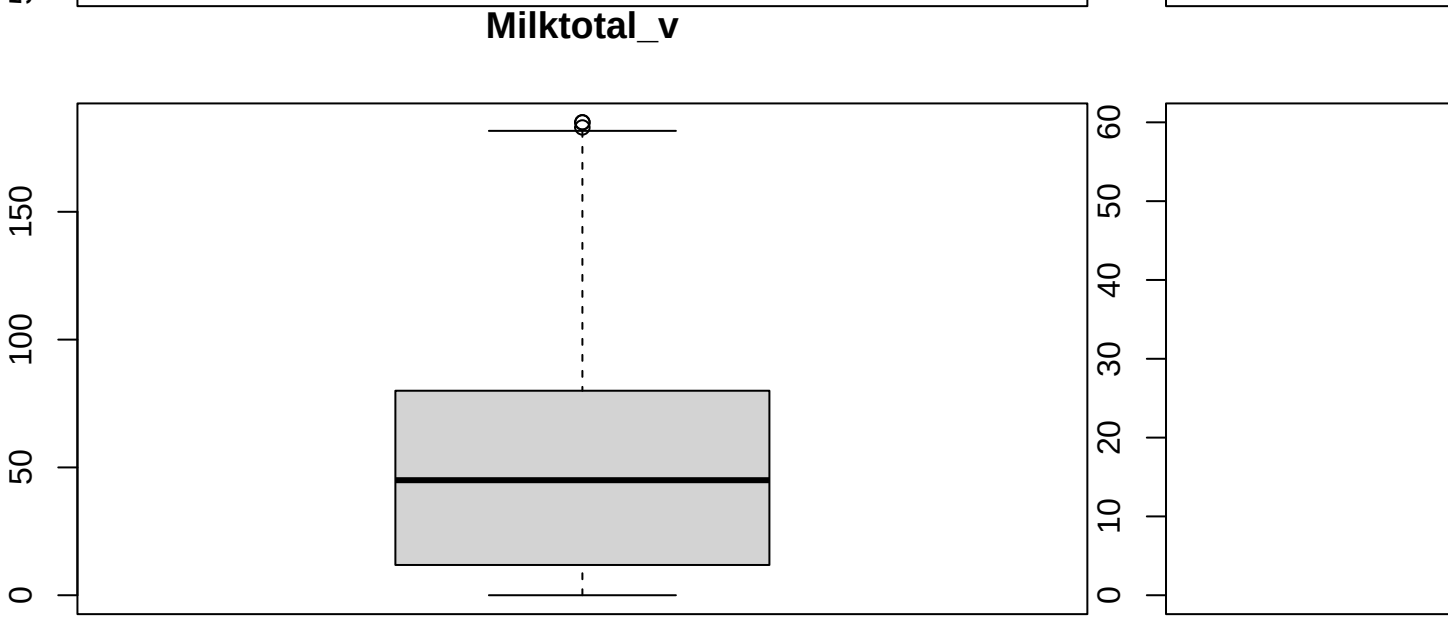
```r
remove_outliers <- function(df, col) {
  q1 <- quantile(df[, col], 0.25) # Calculate the first quartile (25th percentile)
  q3 <- quantile(df[, col], 0.75) # Calculate the third quartile (75th percentile)
  iqr <- q3 - q1 # Calculate the Interquartile Range
  lower_threshold <- q1 - (1.5 * iqr) # Define the lower bound for outlier detection
  upper_threshold <- q3 + (1.5 * iqr) # Define the upper bound for outlier detection
  # Subset the dataframe to include only rows where the column's value is within the defined thresholds
  df <- subset(df, df[, col] >= lower_threshold & df[, col] <= upper_threshold)
  return(df) # Return the dataframe with outliers removed from the specified column
}

# Columns to check for outliers (various consumption variables)
outlier_check_cols <- c("ricetotal_v", "wheattotal_v", "Milktotal_v", "pulsestot_v", "nonvegtotal_v", "
# Define a function to plot boxplots for outlier visualization
plot_outliers <- function(cols) {
  for (outlier_check_col in cols) {
    boxplot(data_meghalaya[, outlier_check_col], main = outlier_check_col) # Create a boxplot for each
  }
}

# Plot initial boxplots to visualize outliers before removal
plot_outliers(outlier_check_cols)
```

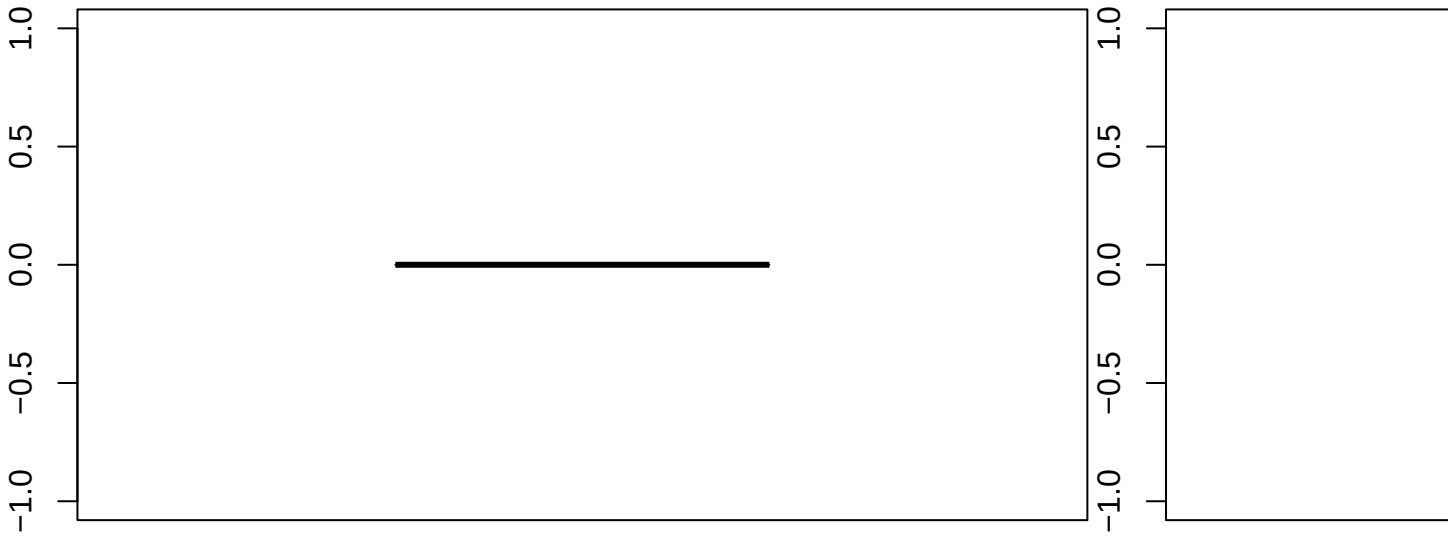





```r
# Columns to apply outlier removal to (a subset of the check columns)
outlier_cols <- c("ricetotal_v", "wheattotal_v", "Milktotal_v", "pulsestot_v")

# Loop through specified columns and apply the outlier removal function
for (outlier_col in outlier_cols) {
  data_meghalaya <- remove_outliers(data_meghalaya, outlier_col)
}

# Plot boxplots again to visualize the effect of outlier removal
plot_outliers(outlier_check_cols)
```


ricetotal_v
300
250
200
150
100
50
15
10
5
0


Milktotal_v
150
100
50
0
60
50
40
30
20
10
0




```
# --- Total Consumption Calculation and Summaries ---

# Define columns representing various consumption categories to be summed
consumption_cols <- c("ricetotal_v", "wheattotal_v", "Milktotal_v", "pulsestot_v", "nonvegtotal_v", "fr

# Calculate the total consumption for each row by summing up the specified consumption columns
data_meghalaya$total_consumption_v <- rowSums(data_meghalaya[, consumption_cols], na.rm = T)

# Define a function to summarize total consumption by a given grouping column
summarize_consumption <- function(col) {
  summary <- data_meghalaya %>%
    group_by(across(all_of(col))) %>% # Group data by the specified column
    summarize(total = sum(total_consumption_v)) %>% # Calculate the sum of total consumption for each g
    arrange(desc(total)) # Arrange the summary table in descending order of total consumption
  return(summary) # Return the summary table
}

# Summarize consumption by District, Region, and Sector
district_summary <- summarize_consumption("District")
region_summary <- summarize_consumption("Region")
sector_summary <- summarize_consumption("Sector")

# Print the top 3 and bottom 3 consuming districts based on the summary
cat("Top 3 Consuming Districts:\n")
```

```
## Top 3 Consuming Districts:
```

```
print(head(district_summary, 3))
```

```
## # A tibble: 3 x 2
##   District  total
##      <int>  <dbl>
```

```
## 1        6 70280.
## 2        1 53582.
## 3        4 36429.
```

```
cat("Bottom 3 Consuming Districts:\n")
```

```
## Bottom 3 Consuming Districts:
```

```
print(tail(district_summary, 3))
```

```
## # A tibble: 3 x 2
##   District  total
##      <int>  <dbl>
## 1        5 29470.
## 2        2 28229.
## 3        3 19920.
```

```
# Print the consumption summary for regions and sectors
cat("Region Consumption Summary:\n")
```

```
## Region Consumption Summary:
```

```
print(region_summary)
```

```
## # A tibble: 1 x 2
##   Region   total
##    <int>   <dbl>
## 1      1 269421.
```

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

```
## Sector Consumption Summary:
```

```
print(sector_summary)
```

```
## # A tibble: 2 x 2
##   Sector   total
##    <int>   <dbl>
## 1      1 184565.
## 2      2  84856.
```

```
# --- Data Transformation (Mapping Codes to Names) ---

# Create a mapping from numeric District codes to their full names
district_map <- c(
  "1" = 'West Garo Hills',
  "2" = 'East Garo Hills',
  "3" = 'South Garo Hills',
  "4" = 'West Khasi Hills',
```

```
  "5" = 'Ri Bhoi',
  "6" = 'East Khasi Hills',
  "7" = 'Jaintia Hills'
)

# Convert District column to character type before applying the map for consistency
data_meghalaya$District <- as.character(data_meghalaya$District)
# Replace numeric District codes with their corresponding names using the map
data_meghalaya$District <- ifelse(data_meghalaya$District %in% names(district_map), district_map[data_m

# Create a mapping from numeric Sector codes to descriptive names
sector_map <- c(
  "1" = "Rural",
  "2" = "Urban"
)

# Convert Sector column to character type before applying the map
data_meghalaya$Sector <- as.character(data_meghalaya$Sector)
# Replace numeric Sector codes with "Rural" or "Urban"
data_meghalaya$Sector <- ifelse(data_meghalaya$Sector %in% names(sector_map), sector_map[data_meghalaya

# Re-summarize consumption by District, Region, and Sector after mapping names
# This ensures summaries use the descriptive names rather than numeric codes
district_summary <- summarize_consumption("District")
region_summary <- summarize_consumption("Region")
sector_summary <- summarize_consumption("Sector")

# --- Z-test for Rural vs. Urban Consumption ---

# Filter total consumption data for rural areas
rural <- data_meghalaya %>%
  filter(Sector == "Rural") %>%
  select(total_consumption_v)

# Filter total consumption data for urban areas
urban <- data_meghalaya %>%
  filter(Sector == "Urban") %>%
  select(total_consumption_v)

# Calculate the mean total consumption for rural and urban areas, ignoring NAs
mean_rural <- mean(rural$total_consumption_v, na.rm = T)
mean_urban <- mean(urban$total_consumption_v, na.rm = T)

# Calculate the standard deviation for rural and urban consumption, ignoring NAs
sd_rural <- sd(rural$total_consumption_v, na.rm = T)
sd_urban <- sd(urban$total_consumption_v, na.rm = T)

# Perform a two-sample Z-test to compare mean consumptions between rural and urban areas
# Using calculated sample standard deviations as estimates for population standard deviations
z_test_result_sector <- z.test(
  x = rural$total_consumption_v, # Data for the first group (rural)
  y = urban$total_consumption_v, # Data for the second group (urban)
  alternative = "two.sided", # Test for a difference in either direction
```

```r
  mu = 0, # Null hypothesis: difference in means is 0
  sigma.x = sd_rural, # Standard deviation of the rural group
  sigma.y = sd_urban, # Standard deviation of the urban group
  conf.level = 0.95 # 95% confidence level
)

# Interpret the results of the Z-test for rural vs. urban consumption
cat("\n--- Z-test Results: Rural vs. Urban Consumption ---\n")
```

```
##
## --- Z-test Results: Rural vs. Urban Consumption ---
```

```r
if (z_test_result_sector$p.value < 0.05) {
  cat("P value is < 0.05 (", round(z_test_result_sector$p.value, 5), "). Therefore, we reject the null h
  cat("There is a significant difference between mean consumptions of urban and rural areas.\n")
  cat("The mean consumption in Rural areas is ", round(mean_rural, 2), " and in Urban areas it's ", roun
} else {
  cat("P value is >= 0.05 (", round(z_test_result_sector$p.value, 5), "). Therefore, we fail to reject t
  cat("There is no significant difference between mean consumptions of urban and rural areas.\n")
  cat("The mean consumption in Rural areas is ", round(mean_rural, 2), " and in Urban areas it's ", roun
}
```

```
## P value is < 0.05 ( 0 ). Therefore, we reject the null hypothesis.
## There is a significant difference between mean consumptions of urban and rural areas.
## The mean consumption in Rural areas is  228.99  and in Urban areas it's  271.97 .
```

```r
# --- Z-test for Top vs. Bottom Consuming Districts ---

# Get the name of the top consuming district from the summarized data
top_district_name <- head(district_summary, 1)$District
# Get the name of the bottom consuming district from the summarized data
bottom_district_name <- tail(district_summary, 1)$District

# Filter total consumption data for the top consuming district
top_district_data <- data_meghalaya %>%
  filter(District == top_district_name) %>%
  select(total_consumption_v)

# Filter total consumption data for the bottom consuming district
bottom_district_data <- data_meghalaya %>%
  filter(District == bottom_district_name) %>%
  select(total_consumption_v)

# Calculate the mean total consumption for the top and bottom districts, ignoring NAs
mean_top_district <- mean(top_district_data$total_consumption_v, na.rm = TRUE)
mean_bottom_district <- mean(bottom_district_data$total_consumption_v, na.rm = TRUE)

# Calculate the standard deviation for the top and bottom districts' consumption, ignoring NAs
sd_top_district <- sd(top_district_data$total_consumption_v, na.rm = TRUE)
sd_bottom_district <- sd(bottom_district_data$total_consumption_v, na.rm = TRUE)

# Perform a two-sample Z-test to compare mean consumptions between the top and bottom districts
```

```r
# Using calculated sample standard deviations as estimates for population standard deviations
z_test_result_district <- z.test(
  x = top_district_data$total_consumption_v, # Data for the top district
  y = bottom_district_data$total_consumption_v, # Data for the bottom district
  alternative = "two.sided", # Test for a difference in either direction
  mu = 0, # Null hypothesis: difference in means is 0
  sigma.x = sd_top_district, # Standard deviation for the top district
  sigma.y = sd_bottom_district, # Standard deviation for the bottom district
  conf.level = 0.95 # 95% confidence level
)

# Interpret the results of the Z-test for top vs. bottom districts
cat("\n--- Z-test Results: Top Consuming District (", top_district_name, ") vs. Bottom Consuming Distric
```

```
##
## --- Z-test Results: Top Consuming District ( East Khasi Hills ) vs. Bottom Consuming District ( South
```

```r
if (z_test_result_district$p.value < 0.05) {
  cat("P value is < 0.05 (", round(z_test_result_district$p.value, 5), ").\n")
  cat("Therefore, we reject the null hypothesis.\n")
  cat("There is a significant difference between mean consumptions of ", top_district_name, " and ", bot
  cat("The mean consumption in ", top_district_name, " is ", round(mean_top_district, 2), " and in ", bo
} else {
  cat("P value is >= 0.05 (", round(z_test_result_district$p.value, 5), ").\n")
  cat("Therefore, we fail to reject the null hypothesis.\n")
  cat("There is no significant difference between mean consumptions of ", top_district_name, " and ", bo
  cat("The mean consumption in ", top_district_name, " is ", round(mean_top_district, 2), " and in ", bo
}
```

```
## P value is >= 0.05 ( 0.37814 ).
## Therefore, we fail to reject the null hypothesis.
## There is no significant difference between mean consumptions of  East Khasi Hills  and  South Garo Hi
## The mean consumption in  East Khasi Hills  is  252.8  and in  South Garo Hills  it's  245.93 .
```