

Statistical Computing Final Project

Sammarieo Brown

2023-05-05

Contents

1	Date Pre-processing & Preparation	3
1.1	Data Wrangling	3
2	Demographic Analysis	5
2.1	5
3	Key Variable Analysis	8
4	Inferential Analysis	13
4.1	Goal 1: To determine if there is a difference in per capita water consumption based on location (area).	13
4.2	Check for Normality	13
4.3	Normality by Groups	13
4.4	multiple comparisons test	16
4.5	Comparison Plot	17
4.6	Test: Independent Sample t-test	18
4.7	Independent Sample t-test	19
4.8	Goal 3: To determine if there is a relationship between per capita water consumption and household size (add control – area, toilet, kitchen)	20
4.9	multiple linear regression model	20
4.10	Diagnostic Plots	21
5	Advanced Data Analysis	21
5.1	Filter the dataset to only include Location and Per Capita water Consumption	22
5.2	Create the dissimilarity matrix using Gower’s distance	22
5.3	Perform hierarchical clustering	22
5.4	Determine the optimal number of clusters	23
5.5	Visualize the dendrogram	24
5.6	Visualize the clusters in a 2D plot	25

5.7	Create a cluster summary	25
5.8	Visualize the cluster summary	25
6	Composite Index	27
6.1	Recode the variables	27
6.2	Calculate the sum of the recoded variables	27
6.3	Cronbach Alpha analysis	28
7	Inferential Analysis of Composite Index	31
7.1	Goal 5 Is there a relationship between shared facilities index and per capita water consumption?	31
7.2	Diagnostic Plots	31
0.0.1	Package Management -> Importing the necessary packages that will be used in this project	

```
suppressPackageStartupMessages({
  library(corrplot)
  library(cluster)
  library(dendextend)
  library(dplyr)
  library(factoextra)
  library(ggplot2)
  library(ggpubr)
  library(ggstatsplot)
  library(ggrepel)
  library(gtsummary)
  library(haven)
  library(huxtable)
  library(kableExtra)
  library(knitr)
  library(performance)
  library(psych)
  library(rstatix)
  library(readr)
  library(rempsyc)
  library(Rtsne)
  library(see)
  library(sjPlot)
  library(sjlabelled)
  library(sjmisc)
  library(tibble)
  library(tidyr)
  library(tidyverse)
  library(tinytex)
  library(gridExtra)
  library(flextable)
```

```
} )
```

1 Date Pre-processing & Preparation

1. Import dataset (SLC_2007.sav)
2. Rename column headers to be more descriptive.
3. convert the .sav file to a .csv file

```
dataframe <- read_sav("F:/DataSpell/Statistical Computing Project/dataset/SLC_2007.sav")

# rename column names to a more descriptive column name

colnames(dataframe)[1:45] <- c("HH_Num", "Final_Weight", "Water_Bill", "HH_Size_All", "HH_Size_Mem", "P

# convert all the column names to lower case
colnames(dataframe) <- tolower(colnames(dataframe))

# converting the SLC_2007.sav to csv

converted_df <- write.table(x=dataframe,file="F:/DataSpell/Statistical Computing Project/dataset/SLC_20
```

1.1 Data Wrangling

This process of the Data Science lifecycle involves cleaning, transforming and restructuring the raw data to make it suitable for analysis.

```
# Set the seed and take 90% of the random sample

set.seed(710)

# import dataset
SLC_2007.Data <- read.csv("F:/DataSpell/Statistical Computing Project/dataset/SLC_2007.csv")

# random sample
n <- nrow(SLC_2007.Data)
sample_size <- round(0.9 * n) # calculate the desired sample size as 90% of the total number of rows

SLC_2007.Sample <- SLC_2007.Data[sample(seq_len(n), size = sample_size, replace = FALSE),] # take a ran

# create a sub dataset with the demographic variables
SLC_2007.Subset <- select(SLC_2007.Sample,
                          area_code,type_dwelling,
                          type_toilet,toilet_shared,kitchen_shared,own_dwelling,
                          water_source,hh_size_all,hh_size_mem,
                          per_cap_con_all,water_bill,water_source, water_bill_latest, water_source_shar
                          water_meter

)
```

```
# rename all the elements of the rows in our subset to labels.
```

```
SLC_2007.Subset <- SLC_2007.Subset %>%  
  rename(  
    area_code = area_code,  
    type_dwelling = type_dwelling,  
    type_toilet = type_toilet,  
    toilet_shared = toilet_shared,  
    kitchen_shared = kitchen_shared,  
    own_dwelling = own_dwelling,  
    water_source = water_source,  
    water_source_shared = water_source_shared,  
    water_meter = water_meter,  
  ) %>%  
  mutate(  
    area_code = case_when(  
      area_code == 1 ~ "KMA",  
      area_code == 2 ~ "Other Town",  
      area_code == 3 ~ "Rural",  
      TRUE ~ as.character(area_code) # keep original value if not matched  
    ),  
    type_dwelling = case_when(  
      type_dwelling == 1 ~ "SEPARATE HOUSE DETACHED",  
      type_dwelling == 2 ~ "SEMI-DETACHED HOUSE",  
      type_dwelling == 3 ~ "PARTS OF A HOUSE",  
      type_dwelling == 4 ~ "APARTMENT BUILDING",  
      type_dwelling == 5 ~ "TOWNHOUSE",  
      type_dwelling == 6 ~ "IMPROVISED HOUSING UNIT",  
      type_dwelling == 7 ~ "PARTS OF COMMERCIAL BUILDING",  
      type_dwelling == 8 ~ "OTHER (SPECIFY)",  
      TRUE ~ as.character(type_dwelling) # keep original value if not matched  
    ),  
    type_toilet = case_when(  
      type_toilet == 1 ~ "W.C. LINKED TO SEWER",  
      type_toilet == 2 ~ "W.C. NOT LINKED",  
      type_toilet == 3 ~ "PIT",  
      type_toilet == 4 ~ "OTHER",  
      type_toilet == 5 ~ "NONE",  
      TRUE ~ as.character(type_toilet) # keep original value if not matched  
    ),  
    toilet_shared = case_when(  
      toilet_shared == 1 ~ "EXCLUSIVE USE",  
      toilet_shared == 2 ~ "SHARED",  
      TRUE ~ as.character(toilet_shared) # keep original value if not matched  
    ),  
    kitchen_shared = case_when(  
      kitchen_shared == 1 ~ "EXCLUSIVE USE",  
      kitchen_shared == 2 ~ "SHARED",  
      kitchen_shared == 3 ~ "NONE",  
      TRUE ~ as.character(kitchen_shared) # keep original value if not matched  
    ),  
  )
```

```

own_dwelling = case_when(
  own_dwelling == 1 ~ "YES",
  own_dwelling == 2 ~ "NO",
  TRUE ~ as.character(own_dwelling) # keep original value if not matched
),
water_source = case_when(
  water_source == 1 ~ "Indoor tap/pipe",
  water_source == 2 ~ "Outside private",
  water_source == 3 ~ "Public standpipe",
  water_source == 4 ~ "Well",
  water_source == 5 ~ "River, Lake, Spring, Pond",
  water_source == 6 ~ "Rainwater (Tank)",
  water_source == 7 ~ "Trucked water (NWC)",
  water_source == 8 ~ "Bottled Water",
  water_source == 9 ~ "Other (Specify)",

  TRUE ~ as.character(water_source) # keep original value if not matched
),
water_source_shared = case_when(
  water_source_shared == 1 ~ "YES",
  water_source_shared == 2 ~ "NO",
  TRUE ~ as.character(water_source_shared) # keep original value if not matched
),
water_meter = case_when(
  water_meter == 1 ~ "Group",
  water_meter == 2 ~ "Individual",
  water_meter == 3 ~ "No Meter",
  TRUE ~ as.character(water_meter) # keep original value if not matched
),

)

```

2 Demographic Analysis

2.1

```

SLC_2007.Subset2 <- SLC_2007.Subset %>%
  rename("Area Code" = area_code)

table1 <- SLC_2007.Subset2 %>%
  select("Area Code") %>%
  filter(!(`Area Code` %in% c(4, 5))) %>% # exclude Area Codes 4 and 5
  tbl_summary(
    missing = "no"
  ) %>%
  add_n() %>%
  modify_header(label = "**Characteristic**") %>%
  bold_labels() %>%
  as_kable_extra() %>% # Convert to kableExtra table
  kable_styling(latex_options = "hold_position", position = "center") # Center the table in the P

```

table1

Characteristic	N	N = 5,863
Area Code	5,863	
KMA		919 (16%)
Other Town		1,208 (21%)
Rural		3,736 (64%)

¹ n (%)

```
SLC_2007.Subset2 <- SLC_2007.Subset %>%
  rename("Type of Toilet" = type_toilet)
table2 <- SLC_2007.Subset2 %>%
  select("Type of Toilet") %>%
  tbl_summary(
    missing = "no"
  ) %>%
  add_n() %>% # add column with total number of non-missing observations
  modify_header(label = "**Characteristic**") %>% # update the column header
  bold_labels() %>%
  as_kable_extra() %>% # Convert to kableExtra table
  kable_styling(latex_options = "hold_position", position = "center") # Center the table in the P
```

table2

Characteristic	N	N = 6,278
Type of Toilet	6,255	
NONE		110 (1.8%)
OTHER		7 (0.1%)
PIT		2,931 (47%)
W.C. LINKED TO SEWER		957 (15%)
W.C. NOT LINKED		2,250 (36%)

¹ n (%)

```
# Load necessary libraries
```

```
# Create a sub dataset with the demographic variables
```

```
SLC_2007.Subset_demographics <- select(SLC_2007.Subset,
  water_source_shared,
  kitchen_shared, toilet_shared, area_code)
```

```
# Prepare the dataset
```

```
SLC_2007.Demographics <- select(SLC_2007.Subset_demographics,
  water_source_shared,
  kitchen_shared, toilet_shared, area_code)
```

```

# Filter out area codes 4 and 5
SLC_2007.Demographics_filtered <- SLC_2007.Demographics %>%
  filter(area_code != 4 & area_code != 5)

# Create summary statistics table
summary_table <- SLC_2007.Demographics_filtered %>%
  tbl_summary(
    by = area_code,
    type = list(
      water_source_shared = "categorical",
      kitchen_shared = "categorical",
      toilet_shared = "categorical"
    ),
    statistic = list(
      water_source_shared ~ "{n} ({p}%)",
      kitchen_shared ~ "{n} ({p}%)",
      toilet_shared ~ "{n} ({p}%"
    ),
    missing = "no",
    label = list(
      area_code ~ "Area Code",
      water_source_shared ~ "Water Source Shared",
      kitchen_shared ~ "Kitchen Shared",
      toilet_shared ~ "Toilet Shared"
    )
  )%>%
  add_n() %>% # add column with total number of non-missing observations
  modify_header(label = "**Characteristic**") %>% # update the column header
  bold_labels()%>%
  as_kable_extra() %>% # Convert to kableExtra table
  kable_styling(latex_options = "hold_position", position = "center") # Center the table in the PDF out,

summary_table

```

Characteristic	N	KMA, N = 919	Other Town, N = 1,208	Rural, N = 3,736
Water Source Shared	1,431			
NO		45 (73%)	160 (86%)	1,076 (91%)
YES		17 (27%)	27 (14%)	106 (9.0%)
Kitchen Shared	5,839			
EXCLUSIVE USE		641 (70%)	980 (81%)	3,141 (84%)
NONE		88 (9.6%)	113 (9.4%)	367 (9.9%)
SHARED		187 (20%)	110 (9.1%)	212 (5.7%)
Toilet Shared	5,709			
EXCLUSIVE USE		586 (64%)	912 (77%)	3,017 (83%)
SHARED		324 (36%)	266 (23%)	604 (17%)

¹ n (%)

```

SLC_2007.Subset2 <- SLC_2007.Subset %>%
  rename("Area Code" = area_code)

table3 <- SLC_2007.Subset2 %>%

```

Characteristic	N	KMA, N = 919	Other Town, N = 1,208	Rural, N = 3,736
per_cap_con_all	5,863			
Mean		131,854.05	105,834.19	76,277.71
SD		137,062.67	110,473.95	91,693.03

```
select("Area Code", per_cap_con_all) %>%
filter(!(`Area Code` %in% c(4, 5))) %>% # exclude Area Codes 4 and 5
group_by(`Area Code`) %>%
tbl_summary(
  by = `Area Code`,
  type = all_continuous() ~ "continuous2",
  statistic = list(all_continuous() ~ c("{mean}", "{sd}")),
  digits = all_continuous() ~ c(2, 2),
  missing = "no"
) %>%
add_n() %>%
modify_header(label = "**Characteristic**") %>%
bold_labels()%>%
as_kable_extra() %>% # Convert to kableExtra table
kable_styling( position = "center") # Center the table in the PDF output
```

table3

3 Key Variable Analysis

```
SLC_2007.Subset2 <- SLC_2007.Subset %>%
  rename("Type of Toilet" = type_toilet)

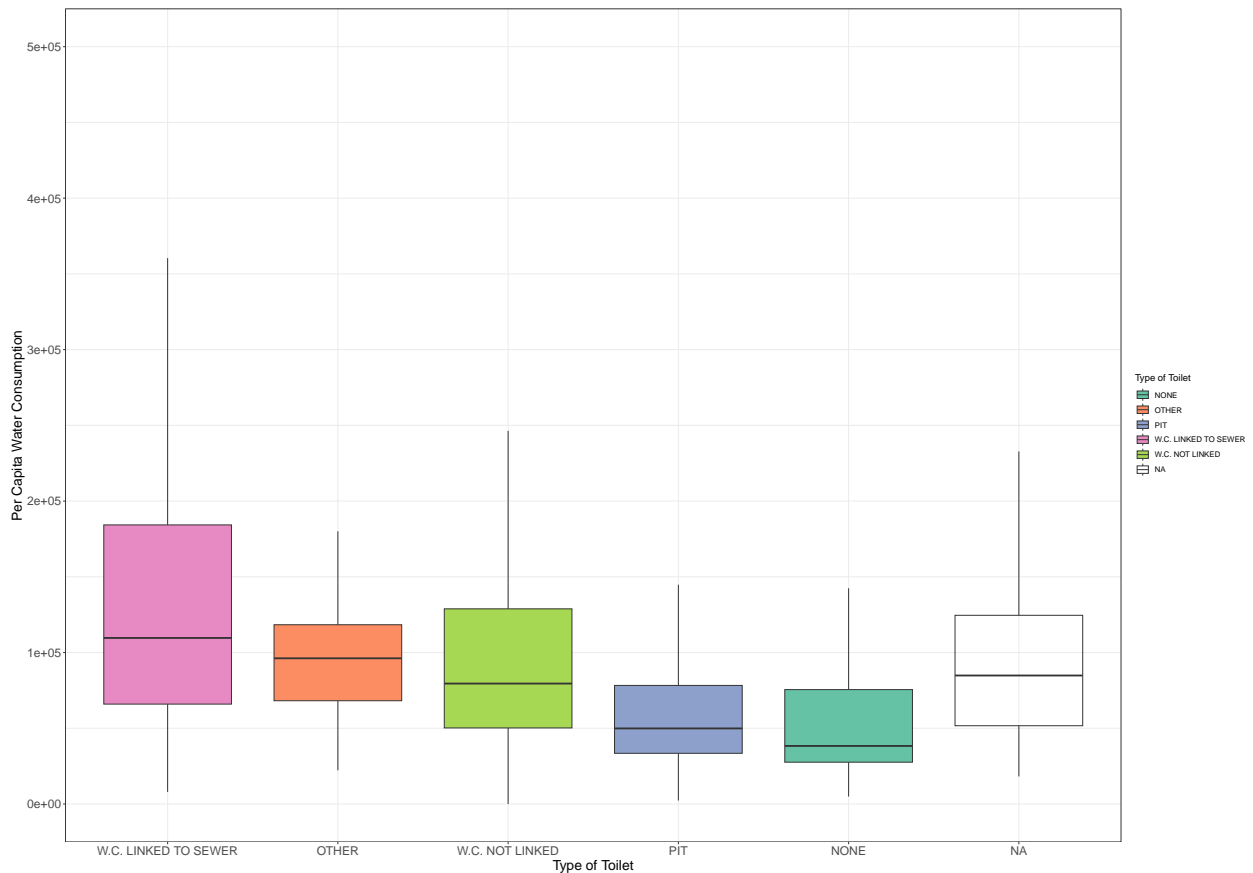
# Calculate median per capita water consumption for each Type of Toilet
medians <- SLC_2007.Subset2 %>%
  group_by(`Type of Toilet`) %>%
  summarise(Median = median(per_cap_con_all, na.rm = TRUE))

# Merge calculated medians back into the main data frame
SLC_2007.Subset2 <- SLC_2007.Subset2 %>%
  left_join(medians, by = "Type of Toilet")

# Create a boxplot with reordered Type of Toilet on the x-axis
boxplot_colored_labeled_sorted <- ggplot(SLC_2007.Subset2, aes(x = reorder(`Type of Toilet`, -Median), y = per_cap_con_all)) +
  geom_boxplot(outlier.shape = NA, coef = 1.5) + # Remove outliers by setting outlier.shape to NA and coef to 1.5
  coord_cartesian(ylim = c(0, 5e+05)) + # Set y-axis limits to 0 and 5e+05
  scale_fill_brewer(palette = "Set2") + # Apply a color theme from the ColorBrewer palette
  theme_bw() + # Use a black and white theme for the plot
  labs(
    x = "Type of Toilet",
    y = "Per Capita Water Consumption") +
  theme(axis.text = element_text(size = 14), # Increase the font size of the axis text to 14
        axis.title = element_text(size = 16)) # Increase the font size of the axis titles to 16
```

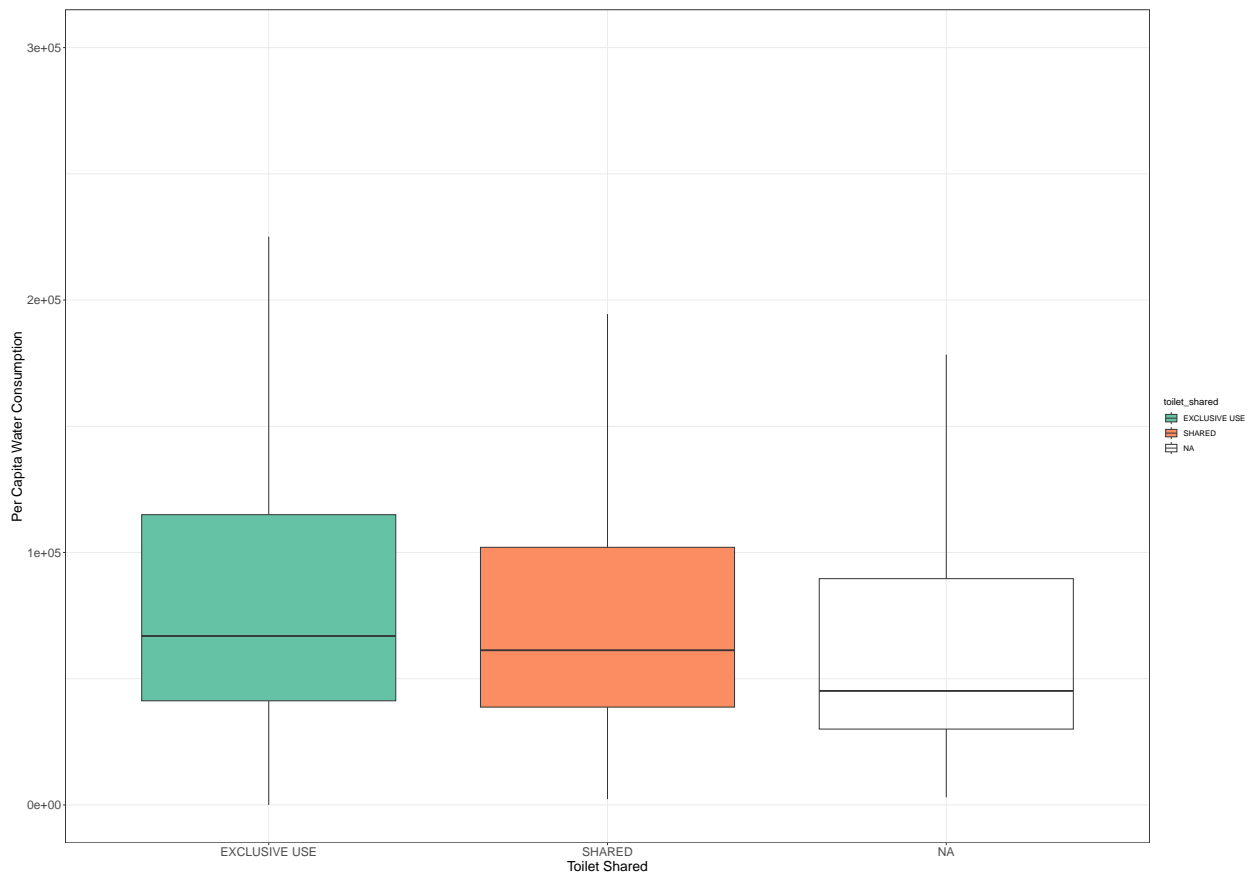


```
# Print the sorted boxplot
boxplot_colored_labeled_sorted
```



```
# Create a boxplot of toilet_shared and per_cap_con_all
boxplot_colored <- ggplot(SLC_2007.Subset, aes(x = toilet_shared, y = per_cap_con_all, fill = toilet_sh
  geom_boxplot(outlier.shape = NA, coef = 1.5) + # Remove outliers by setting outlier.shape to NA and c
  coord_cartesian(ylim = c(0, 3e+05)) + # Set y-axis limits to 0 and 5e+05
  scale_fill_brewer(palette = "Set2") + # Apply a color theme from the ColorBrewer palette
  theme_bw() + # Use a black and white theme for the plot
  labs(
    x = "Toilet Shared",
    y = "Per Capita Water Consumption") +
  theme(axis.text = element_text(size = 14), # Increase the font size of the axis text to 14
        axis.title = element_text(size = 16)) # Increase the font size of the axis titles to 16

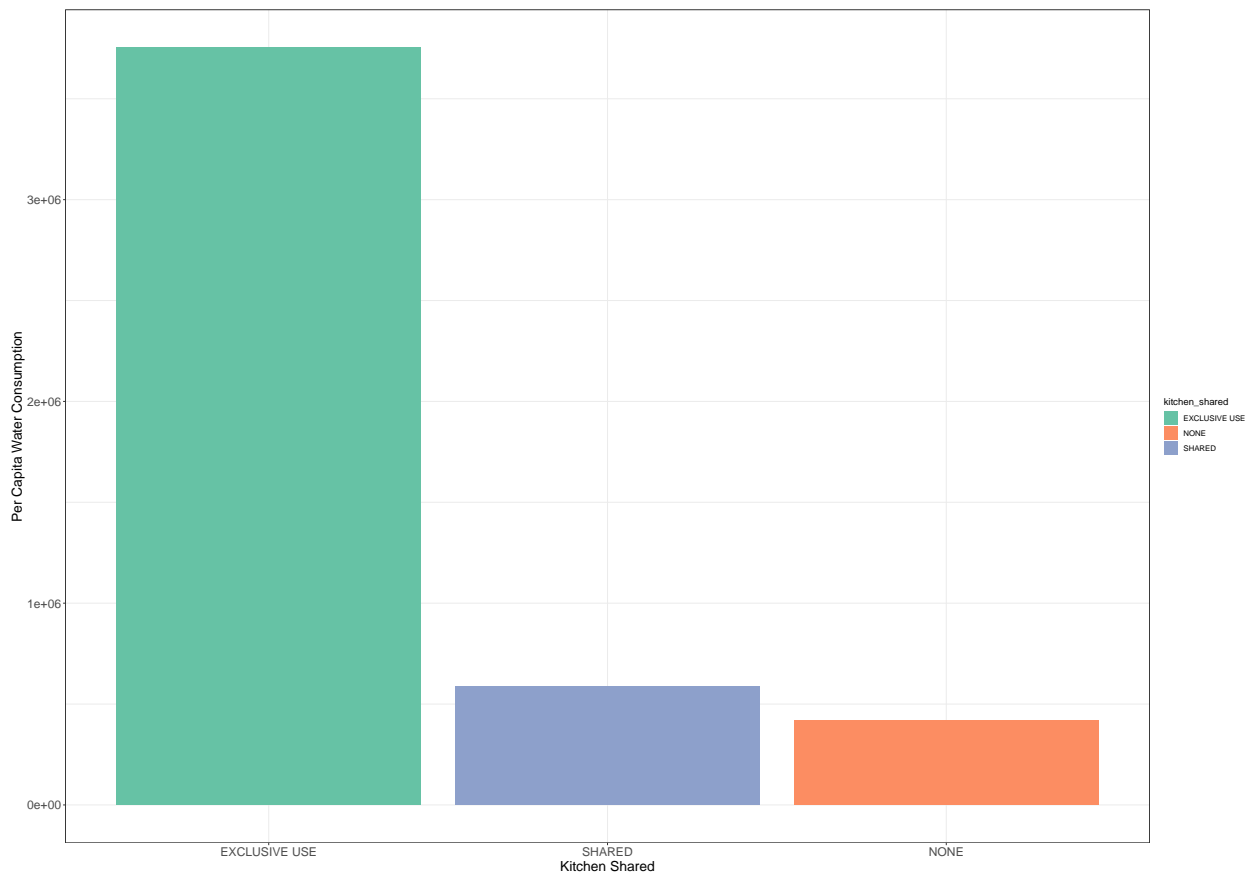
# Print the boxplot
boxplot_colored
```



```
# Calculate the percentages for each kitchen_shared category, removing NA values
SLC_2007.Subset4 <- SLC_2007.Subset %>%
  filter(!is.na(kitchen_shared)) %>%
  select(kitchen_shared, per_cap_con_all) %>% # Include 'per_cap_con_all' in the dataframe
  count(kitchen_shared, per_cap_con_all) %>% # Add 'per_cap_con_all' in the count function
  mutate(percentage = n / sum(n) * 100) %>%
  arrange(desc(percentage))

# Create a bar chart of kitchen_shared and per_cap_con_all, with percentage labels
bar_chart_colored <- ggplot(SLC_2007.Subset4, aes(x = reorder(kitchen_shared, -per_cap_con_all), y = per_cap_con_all)) +
  geom_bar(stat = "identity", position = position_dodge(width = 0.9)) + # Use the identity statistic to
  scale_fill_brewer(palette = "Set2") + # Apply a color theme from the ColorBrewer palette
  theme_bw() + # Use a black and white theme for the plot
  labs(
    x = "Kitchen Shared",
    y = "Per Capita Water Consumption") +
  theme(axis.text = element_text(size = 14), # Increase the font size of the axis text to 14
        axis.title = element_text(size = 16)) # Increase the font size of the axis titles to 16

# Print the bar chart
bar_chart_colored
```

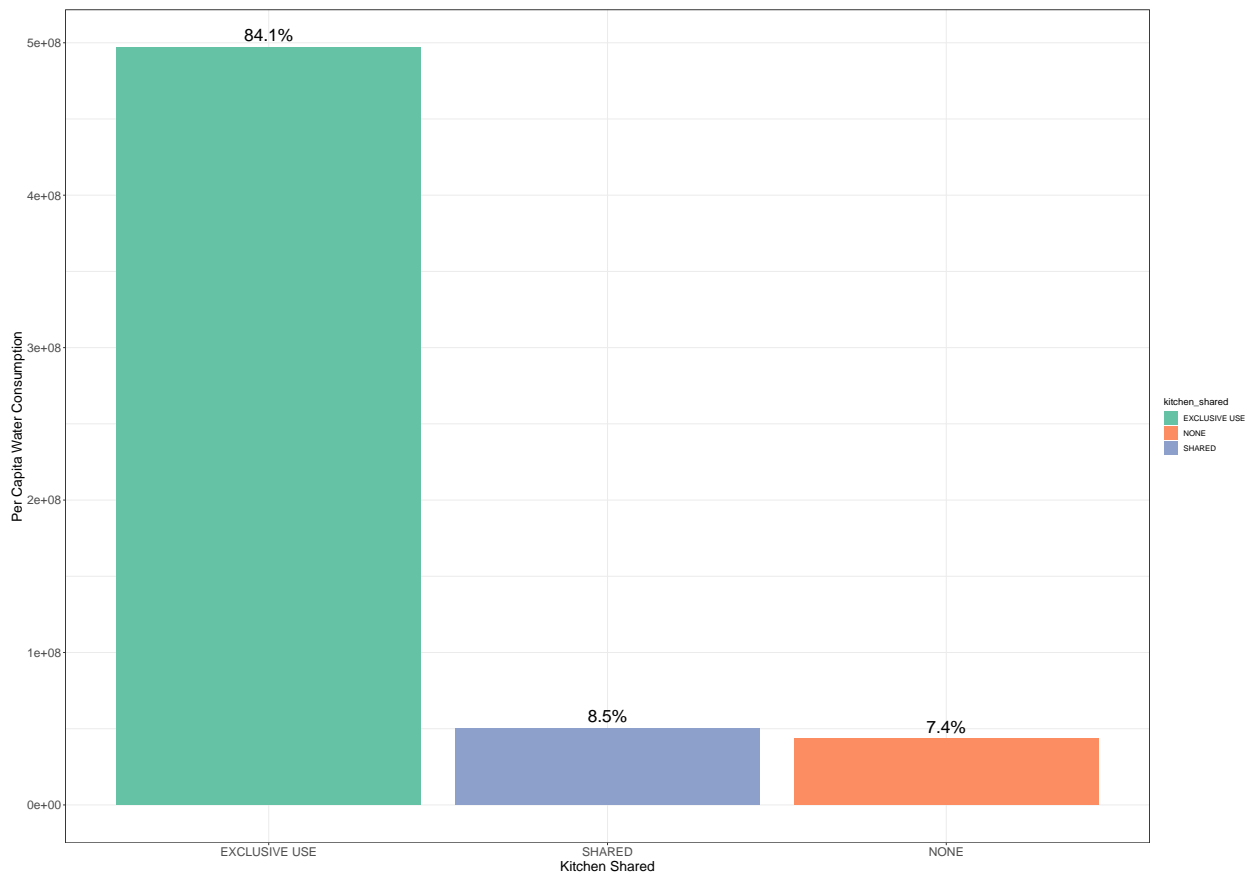


```
# Calculate the total per capita water consumption for each kitchen_shared category, removing NA values
SLC_2007.Subset4 <- SLC_2007.Subset %>%
  filter(!is.na(kitchen_shared)) %>%
  select(kitchen_shared, per_cap_con_all) %>%
  group_by(kitchen_shared) %>%
  summarise(total_per_cap_con_all = sum(per_cap_con_all, na.rm = TRUE))

# Calculate the percentage of per capita water consumption for each kitchen_shared category
SLC_2007.Subset4 <- SLC_2007.Subset4 %>%
  mutate(percentage = total_per_cap_con_all / sum(total_per_cap_con_all))

# Create a bar chart of kitchen_shared and per_cap_con_all, with percentage labels
bar_chart_colored <- ggplot(SLC_2007.Subset4, aes(x = reorder(kitchen_shared, -total_per_cap_con_all), y = per_cap_con_all)) +
  geom_bar(stat = "identity", position = position_dodge(width = 0.9)) +
  geom_text(aes(label = paste0(round(percentage * 100, 1), "%")), position = position_dodge(width = 0.9)) +
  scale_fill_brewer(palette = "Set2") +
  theme_bw() +
  labs(
    x = "Kitchen Shared",
    y = "Per Capita Water Consumption") +
  theme(axis.text = element_text(size = 14),
        axis.title = element_text(size = 16))

# Print the bar chart
bar_chart_colored
```



```
SLC_2007.Subset4 <- select(SLC_2007.Subset, area_code, kitchen_shared, per_cap_con_all, hh_size_all, )
```

```
SLC_2007.Subset4 <- SLC_2007.Subset4 %>%
  mutate(
    hh_size_all = case_when(
      hh_size_all == 1 ~ "1 person",
      hh_size_all == 2 ~ "2 person",
      hh_size_all == 3 ~ "3 person",
      hh_size_all >= 4 ~ "4 or more"
    )
  )
```

```
SLC_2007.Subset4 <- SLC_2007.Subset4 %>%
  filter(area_code != "Unknown")
```

```
SLC_2007.Subset2 <- SLC_2007.Subset %>%
  rename("Type of Toilet" = type_toilet)
```

```
# Calculate summary statistics for each Type of Toilet
summary_table <- SLC_2007.Subset2 %>%
  group_by(toilet_shared) %>%
  summarise(
    Count = n(),
    Min = min(per_cap_con_all, na.rm = TRUE),
```

```

    Q1 = quantile(per_cap_con_all, 0.25, na.rm = TRUE),
    Median = median(per_cap_con_all, na.rm = TRUE),
    Mean = mean(per_cap_con_all, na.rm = TRUE),
    Q3 = quantile(per_cap_con_all, 0.75, na.rm = TRUE),
    Max = max(per_cap_con_all, na.rm = TRUE),
    SD = sd(per_cap_con_all, na.rm = TRUE)
  ) %>%
  as.data.frame()

```

4 Inferential Analysis

4.1 Goal 1: To determine if there is a difference in per capita water consumption based on location (area).

•

4.1.1 Test: One-way ANOVA

```

# Select the columns from your dataset
SLC_2007.Goal_1 <- select(SLC_2007.Subset, area_code, per_cap_con_all)
SLC_2007.Goal_1 <- SLC_2007.Goal_1 %>%
  mutate(area_code = recode(area_code,
    `1` = "KMA",
    `2` = "Other Town",
    `3` = "Rural",
  )
) %>%
  filter(area_code != "4", area_code != "5")

```

4.2 Check for Normality

Run the Linear Model

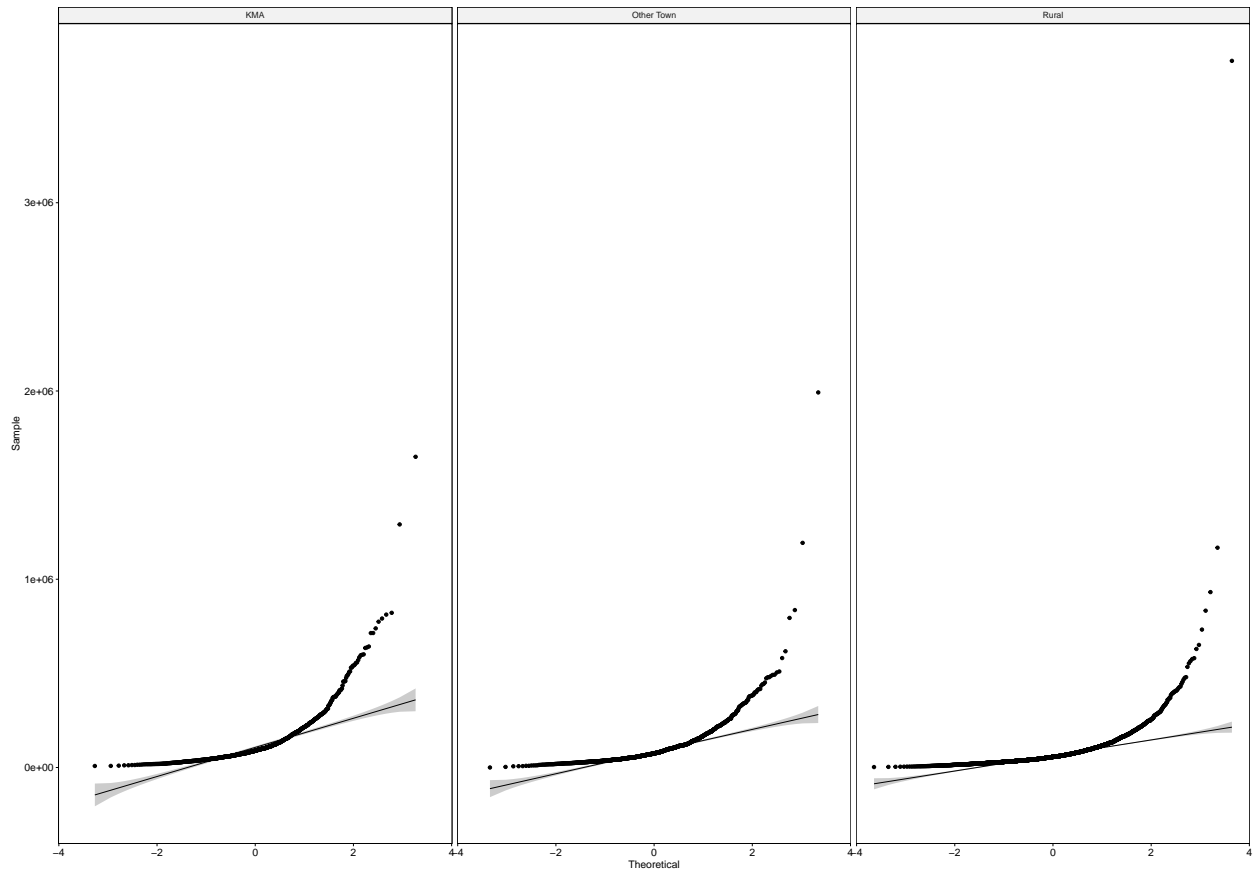
```
aov.model.test <- lm(per_cap_con_all ~ area_code, data = SLC_2007.Goal_1)
```

4.3 Normality by Groups

```

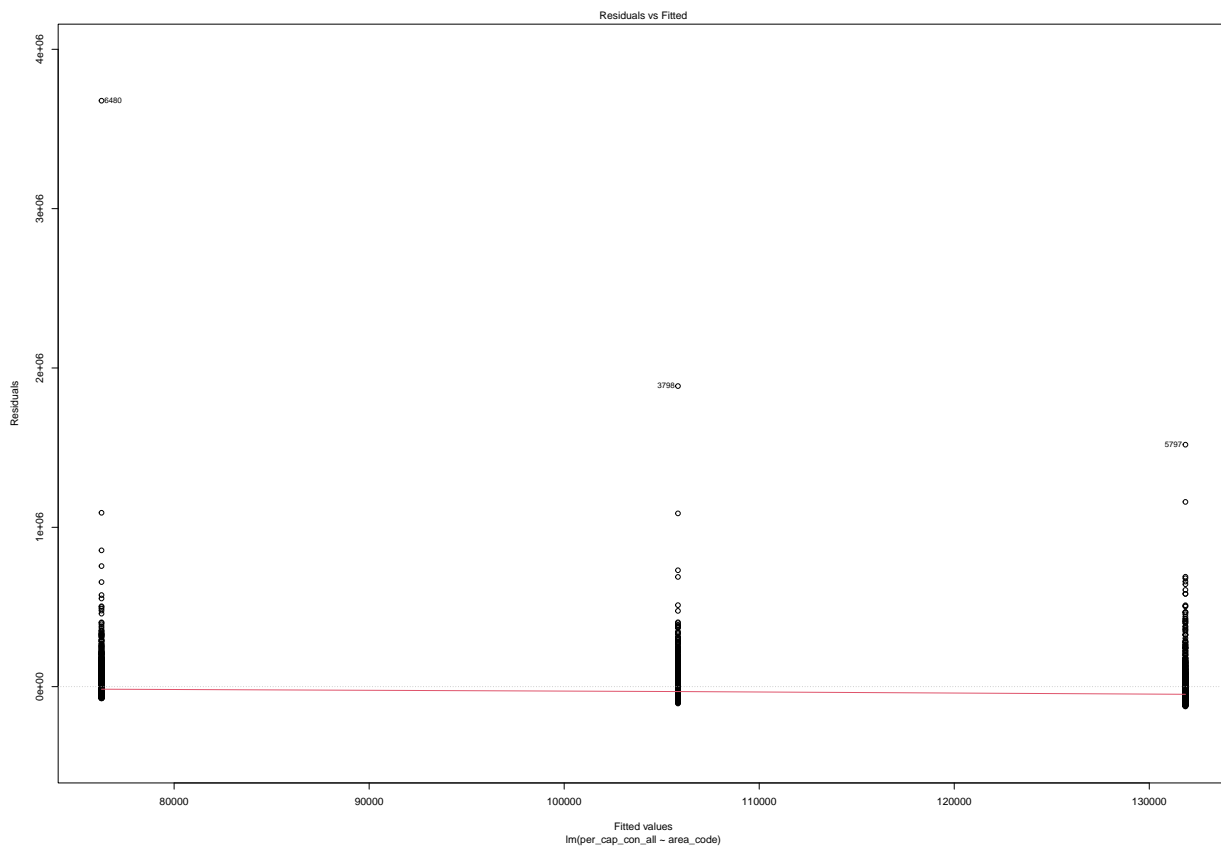
# Check normality by groups, ignoring NA values
ggqqplot(na.omit(SLC_2007.Goal_1), "per_cap_con_all", facet.by = "area_code")

```



Check for equal variance

```
plot(aov.model.test,1)
```



One-Way ANOVA Test

```
aov.test <- aov(per_cap_con_all ~ area_code, data = SLC_2007.Goal_1)
aov.summary <- summary(aov.test)

# Calculate r^2
RSq <- var(predict(aov.test)) / var(SLC_2007.Goal_1$per_cap_con_all, na.rm = TRUE)
RSq <- round(RSq, 4)

# Extract data for the table
anova_table <- data.frame(
  Df = aov.summary[[1]][, "Df"],
  SumSq = aov.summary[[1]][, "Sum Sq"],
  MeanSq = aov.summary[[1]][, "Mean Sq"],
  FValue = aov.summary[[1]][, "F value"],
  Pr = aov.summary[[1]][, "Pr(>F)"]
)

# Add R-squared to the table
anova_table <- rbind(anova_table,
  data.frame(Df = NA,
    SumSq = NA,
    MeanSq = NA,
    FValue = NA,
    Pr = RSq))
```

```

# Convert the anova table to a tbl_summary and then to a kable
anova_tbl_summary <- tbl_summary(
  anova_table,
  missing = "no"
) %>%
  add_n() %>%
  modify_header(label = "**Characteristic**") %>%
  bold_labels() %>%
  as_kable_extra()

# Center and style the kable
styled_anova_table <- anova_tbl_summary %>%
  kable_styling(latex_options = "hold_position", position = "center")

# Display styled kable
styled_anova_table

```

Characteristic	N	N = 3
Df	2	
2		1 (50%)
5860		1 (50%)
SumSq	2	
2609410464361.86		1 (50%)
63378961886048.3		1 (50%)
MeanSq	2	
10815522506.1516		1 (50%)
1304705232180.93		1 (50%)
FValue	1	
120.632658425781		1 (100%)
Pr	2	
4.56747198077927e-52		1 (50%)
0.0395		1 (50%)

¹ n (%)

4.4 multiple comparisons test

```

# pairwise.t.test(SLC_2007.Goal_1$per_cap_con_all, SLC_2007.Goal_1$area_code, p.adjust.method = "bonferroni")
# Perform the pairwise t-test
pairwise_test <- pairwise.t.test(SLC_2007.Goal_1$per_cap_con_all, SLC_2007.Goal_1$area_code, p.adjust.method = "bonferroni")

# Extract data for the table
pairwise_matrix <- pairwise_test$p.value
pairwise_values <- as.data.frame(pairwise_matrix)

# Create a new data frame to store the pairwise t-test results
pairwise_table <- data.frame(
  Comparison = rownames(pairwise_values),
  KMA = pairwise_values[, "KMA"],
  Other_Town = pairwise_values[, "Other Town"]
)

```



```

pairwise_tbl_summary <- tbl_summary(
  pairwise_table,
  missing = "no"
) %>%
  add_n() %>%
  modify_header(label = "**Characteristic**") %>%
  bold_labels() %>%
  as_kable_extra()

# Center and style the kable
styled_pairwise_table <- pairwise_tbl_summary %>%
  kable_styling(latex_options = "hold_position", position = "center")

# Display styled kable
styled_pairwise_table

```

Characteristic	N	N = 2
Comparison	2	
Other Town		1 (50%)
Rural		1 (50%)
KMA	2	
1.93412806581929e-46		1 (50%)
3.43407328972517e-08		1 (50%)
Other__Town	1	
3.40026236052605e-17		1 (100%)

¹ n (%)

4.5 Comparison Plot

```

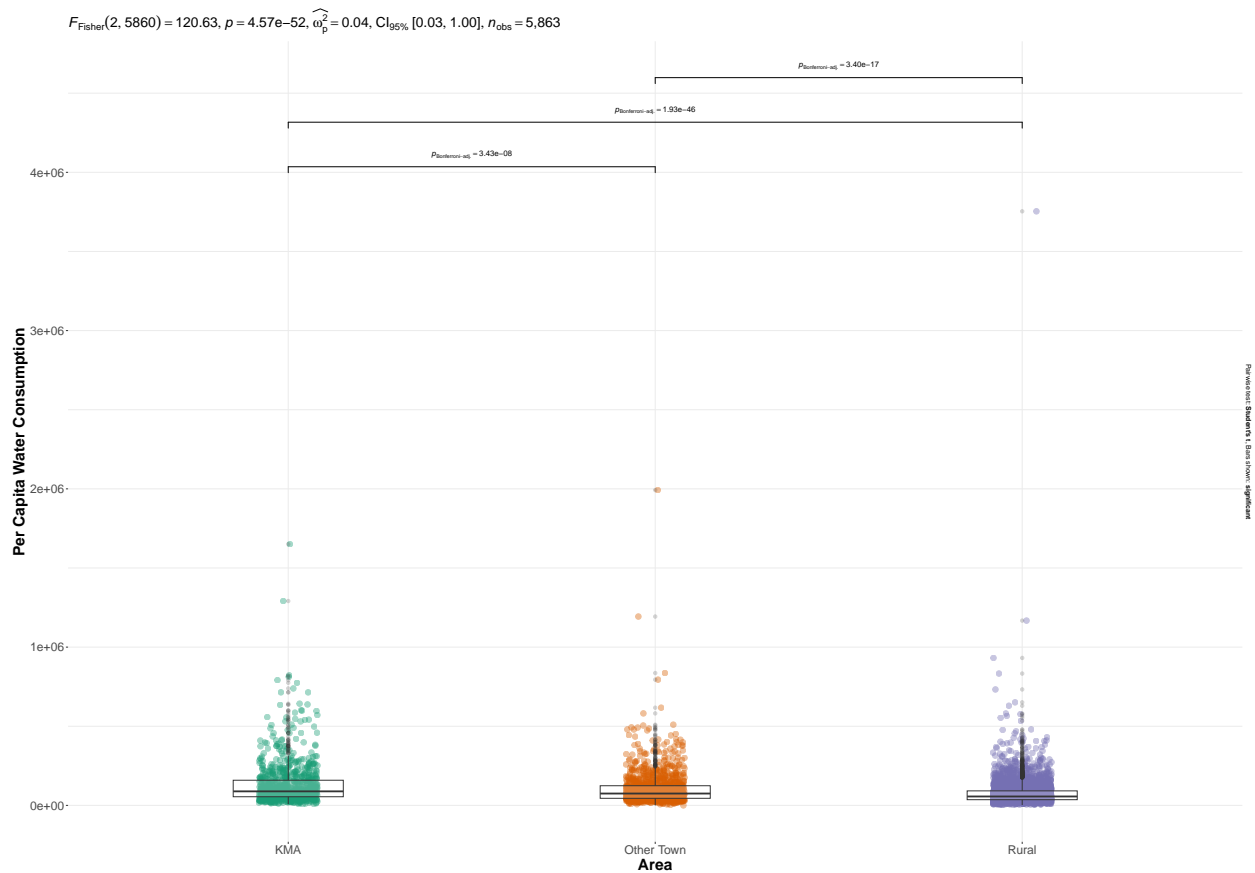
# Create the plot using ggbetweenstats()
anova_plot <- ggbetweenstats(
  data = SLC_2007.Goal_1,
  x = area_code,
  y = per_cap_con_all,
  type = "parametric",
  var.equal = TRUE,
  plot.type = "box",
  pairwise.comparisons = TRUE,
  p.adjust.method = "bonferroni",
  pairwise.display = "significant",
  centrality.plotting = FALSE,
  bf.message = FALSE
)

# Modify y-axis title
anova_plot <- anova_plot +
  ylab("Per Capita Water Consumption") +
  xlab("Area")

```

```
# Customize the theme to enlarge elements
anova_plot <- anova_plot +
  theme(
    text = element_text(size = 16), # Increase base text size
    axis.title = element_text(size = 18), # Increase axis title size
    axis.text = element_text(size = 14), # Increase axis text size
    plot.title = element_text(size = 20, face = "bold"), # Increase plot title size
    strip.text = element_text(size = 16), # Increase facet label text size
    legend.text = element_text(size = 14), # Increase legend text size
    legend.title = element_text(size = 16), # Increase legend title size
    panel.spacing = unit(1, "lines") # Increase space between facets
  )

# Display the plot
anova_plot
```



Goal 2: To determine if there is a difference in per capita water consumption based on whether toilet facilities are shared or not.

•

4.6 Test: Independent Sample t-test

```

# Prepare the dataset
SLC_2007.Goal_2 <- select(SLC_2007.Subset, toilet_shared, per_cap_con_all)
SLC_2007.Goal_2 <- SLC_2007.Goal_2 %>%
  mutate(toilet_shared = recode(toilet_shared,
                                `1` = "EXCLUSIVE USE",
                                `2` = "SHARED",

                                )
  ) %>%
  filter(toilet_shared != "NA")

# Run the independent sample t-test
t_test_result <- SLC_2007.Goal_2 %>%
  filter(toilet_shared != "Unknown") %>% # Remove rows with "Unknown" values
  tbl_summary(
    by = toilet_shared,
    type = c(per_cap_con_all = "continuous"),
    statistic = list(per_cap_con_all ~ "{mean} ({sd})"),
    missing = "no",
    label = list(
      per_cap_con_all ~ "Per Capita Water Consumption"
    )
  ) %>%
  add_difference() %>%
  add_n() %>% # add column with total number of non-missing observations
  modify_header(label = "**Characteristic**") %>% # update the column header
  bold_labels() %>%
  as_kable_extra() %>% # Convert to kableExtra table
  kable_styling(latex_options = "hold_position") # Center the table in the PDF output

# Display the result
t_test_result

```

Characteristic	N	EXCLUSIVE USE, N = 4,880	SHARED, N = 1,244	Difference
Per Capita Water Consumption	6,124	97,879 (115,289)	83,389 (70,901)	14,490

¹ Mean (SD)

² Welch Two Sample t-test

³ CI = Confidence Interval

4.7 Independent Sample t-test

```

# Run the independent sample t-test using Version 2
t_test_result_v2 <- nice_t_test(data = SLC_2007.Goal_2,
                                response = "per_cap_con_all",
                                group = "toilet_shared",
                                warning = FALSE) %>%

  nice_table()

# Display the result
# t_test_result_v2

```

4.8 Goal 3: To determine if there is a relationship between per capita water consumption and household size (add control – area, toilet, kitchen)

•

```
SLC_2007.Goal_3 <- select(SLC_2007.Subset, per_cap_con_all, hh_size_all, area_code, toilet_shared, kitchen_shared)
# recode variables.
SLC_2007.Goal_3 <- SLC_2007.Goal_3 %>%
  mutate(
    toilet_shared = recode(toilet_shared,
                          `1` = "EXCLUSIVE USE",
                          `2` = "SHARED"
    ),
    kitchen_shared = recode(kitchen_shared,
                           `1` = "EXCLUSIVE USE",
                           `2` = "SHARED",
                           `3` = "NONE"
    ),
    area_code = recode(area_code,
                       `1` = "KMA",
                       `2` = "Other Town",
                       `3` = "Rural",
    ),
    hh_size_all = case_when(
      hh_size_all == 1 ~ "1 person",
      hh_size_all == 2 ~ "2 person",
      hh_size_all == 3 ~ "3 person",
      hh_size_all >= 4 ~ "4 or more"
    )
  ) %>%
  filter(area_code != "4", area_code != "5",
         kitchen_shared != "NA",
         toilet_shared != "NA",
  )
```

4.9 multiple linear regression model

```
# Convert nominal variables to factors
SLC_2007.Goal_3_recoded <- SLC_2007.Goal_3 %>%
  mutate(
    toilet_shared = as.factor(toilet_shared),
    kitchen_shared = as.factor(kitchen_shared),
    area = as.factor(area_code),
    hh_size_all = as.factor(hh_size_all)
```

```

)

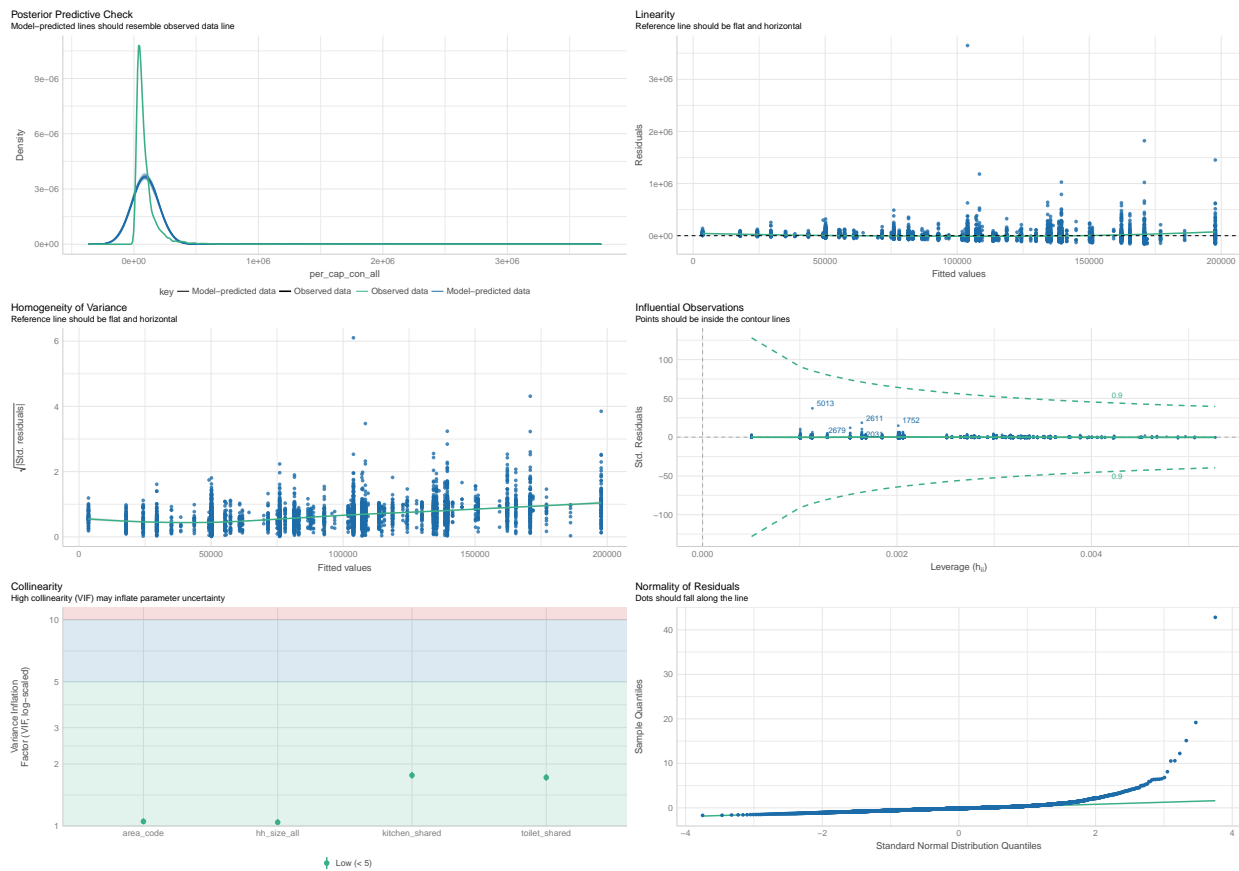
# Run the multiple linear regression model
model <- lm(per_cap_con_all ~ hh_size_all + area_code + toilet_shared + kitchen_shared, data = SLC_2007)
# tab_model(model)

```

4.10 Diagnostic Plots

```
# Diagnostic Plots
```

```
check_model(model)
```



5 Advanced Data Analysis

```
SLC_2007.Subset200 <- sample_n(SLC_2007.Subset, 200)

SLC_2007.Subset200 <- SLC_2007.Subset200 %>%
  filter(area_code != 4 & area_code != 5)
```

5.1 Filter the dataset to only include Location and Per Capita water Consumption

```
SLC_2007.Subset200_filtered <- SLC_2007.Subset200 %>%
  select(area_code, per_cap_con_all, hh_size_all)

SLC_2007.Subset200_filtered <- SLC_2007.Subset200_filtered %>%
  mutate(
    hh_size_all = case_when(
      hh_size_all == 1 ~ "1 person",
      hh_size_all == 2 ~ "2 person",
      hh_size_all == 3 ~ "3 person",
      hh_size_all >= 4 ~ "4 or more"
    )
  )

SLC_2007.Subset200_filtered$area_code <- recode(SLC_2007.Subset200_filtered$area_code,
  "Rural" = 1,
  "KMA" = 2,
  "Other Town" = 3)

SLC_2007.Subset200_filtered$hh_size_all <- recode(SLC_2007.Subset200_filtered$hh_size_all,
  "1 person" = 1,
  "2 person" = 2,
  "3 person" = 3,
  "4 or more" = 4
)
```

5.2 Create the dissimilarity matrix using Gower's distance

```
DistanceMatrix <- daisy(SLC_2007.Subset200_filtered, metric = "gower")
```

5.3 Perform hierarchical clustering

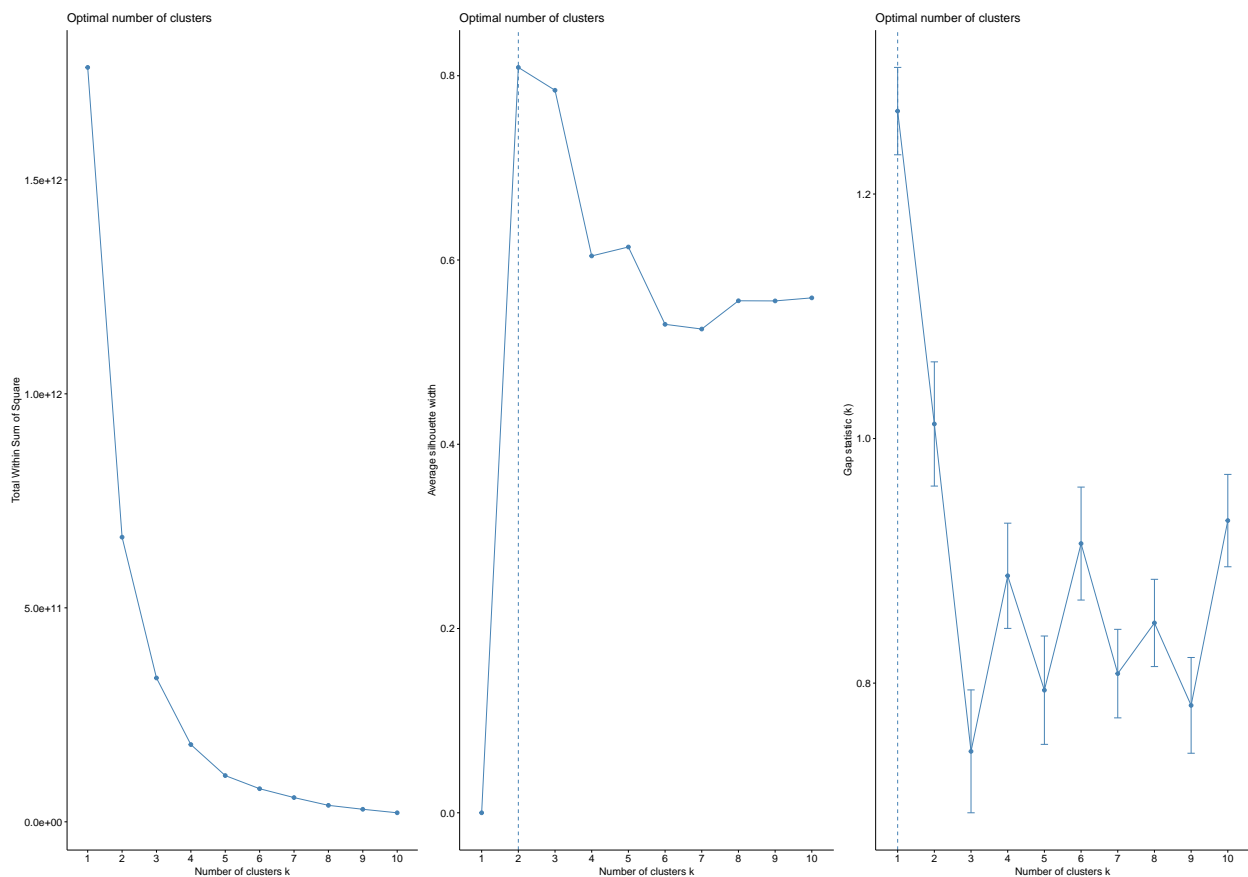
```
hc <- hclust(DistanceMatrix, method = "complete")
```

5.4 Determine the optimal number of clusters

```
plot1 <- fviz_nbclust(SLC_2007.Subset200_filtered, FUN = hcut, method = "wss")
plot2 <- fviz_nbclust(SLC_2007.Subset200_filtered, FUN = hcut, method = "silhouette")
gap_stat <- clusGap(SLC_2007.Subset200_filtered, FUN = hcut, nstart = 25, K.max = 10, B = 50)
```

```
## Clustering k = 1,2,..., K.max (= 10): .. done
## Bootstrapping, b = 1,2,..., B (= 50) [one "." per sample]:
## ..... 50
```

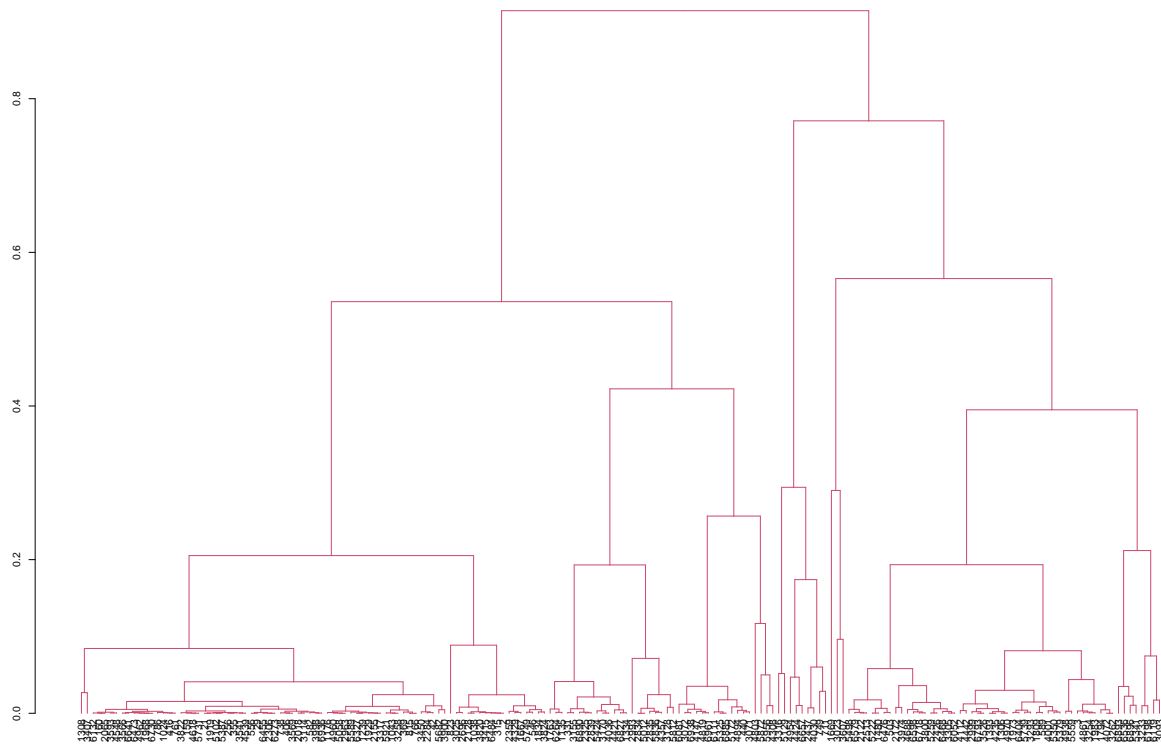
```
plot3 <- fviz_gap_stat(gap_stat)
grid.arrange(plot1, plot2, plot3, ncol = 3)
```



```
# 3 plots side by side
#ggarrange
```

5.5 Visualize the dendrogram

```
dendrogram <- as.dendrogram(hc)
ColourDendrogram <- color_branches(dendrogram, h = 3)
plot(ColourDendrogram)
```



```
## Assign cluster labels to the observations
```

```
clusterLabs <- cutree(hc, k = 2) # Replace with the optimal number of clusters found
SLC_2007.Subset200_clusters <- cbind(SLC_2007.Subset200_filtered, cluster = as.factor(clusterLabs))
```


5.6 Visualize the clusters in a 2D plot

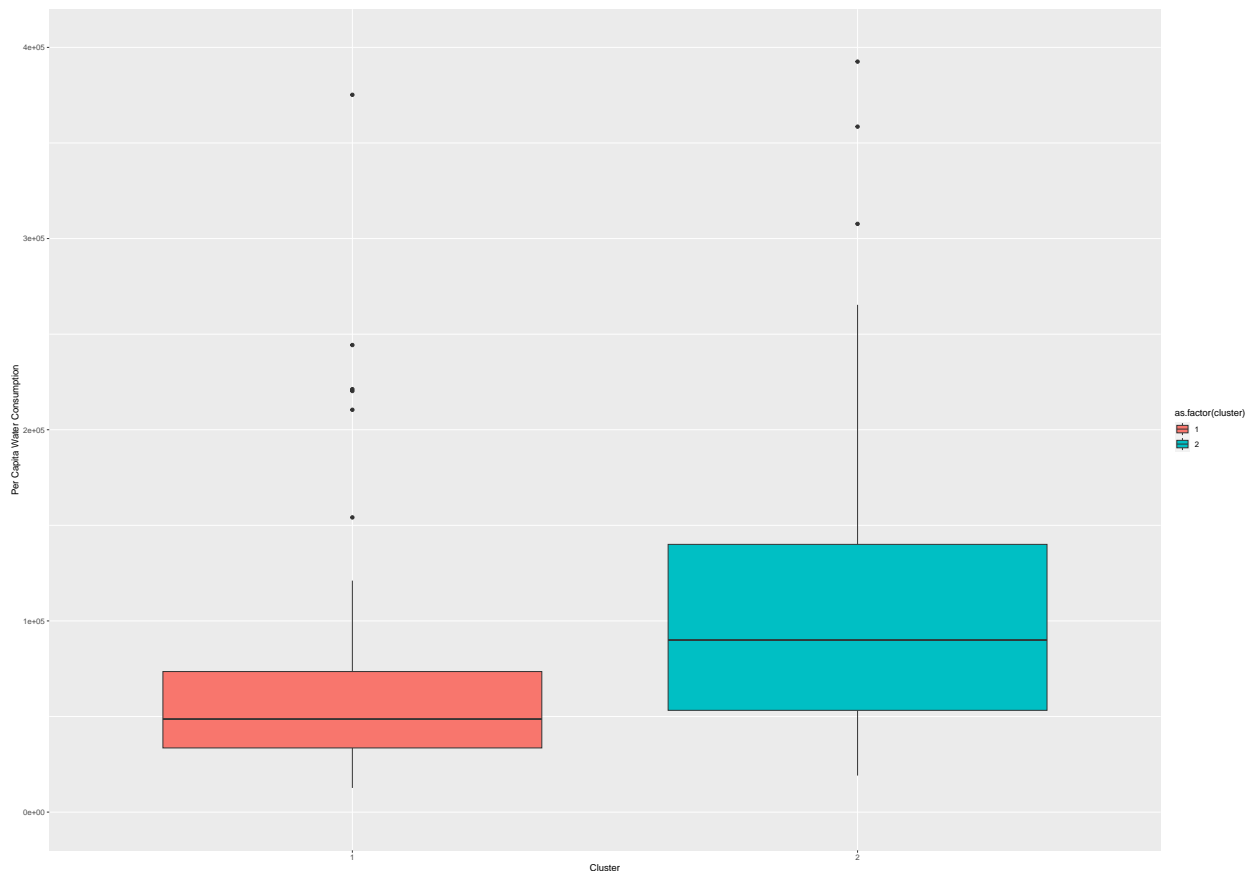
```
fviz_cluster(list(data = SLC_2007.Subset200_filtered, cluster = clusterLabs))
```

5.7 Create a cluster summary

```
cluster_summary <- SLC_2007.Subset200_clusters %>%  
  group_by(cluster) %>%  
  summarise(across(everything(), mean, na.rm = TRUE))
```

5.8 Visualize the cluster summary

```
ggplot(SLC_2007.Subset200_clusters, aes(x = as.factor(cluster), y = per_cap_con_all, fill = as.factor(c  
  geom_boxplot() +  
  coord_cartesian(ylim = c(0, 4e+05)) +  
  labs(x = "Cluster", y = "Per Capita Water Consumption")
```



```

# Create a summary table for per capita water consumption by cluster
summary_table_boxplot <- SLC_2007.Subset200_clusters %>%
  mutate(cluster = as.factor(cluster)) %>%
  select(cluster, per_cap_con_all) %>%
  gtsummary::tbl_summary(by = cluster,
                        missing = "no",
                        type = list(per_cap_con_all = "continuous"),
                        statistic = list(per_cap_con_all = "{mean} ({sd}); Median: {median}; IQR: {p25}"))
  add_difference() %>%
  add_n() %>% # add column with total number of non-missing observations
  modify_header(label = "**Characteristic**") %>% # update the column header
  bold_labels() %>%
  as_kable_extra() %>% # Convert to kableExtra table
  kable_styling(latex_options = "hold_position") # Center the table in the PDF output
# Display the r

# Display the summary table
summary_table_boxplot

```

Characteristic	N	1, N = 119	2, N = 66
per_cap_con_all	185	62,731 (50,524); Median: 48,680; IQR: 33,562-73,519	129,553 (139,887); Median: 90,008; IQR: 62,731-129,553

¹ Mean (SD); Median: Median; IQR: 25%-75%

² Welch Two Sample t-test

³ CI = Confidence Interval

```

# Create a summary table
summary_table <- SLC_2007.Subset200_clusters %>%
  mutate(cluster = as.factor(cluster),
         area_code = recode(area_code,
                           "1" = "Rural",
                           "2" = "KMA",
                           "3" = "Other Town"),
         hh_size_all = recode(hh_size_all,
                              "1" = "1 person",
                              "2" = "2 person",
                              "3" = "3 person",
                              "4" = "4 or more")) %>%
  group_by(cluster) %>%
  select(cluster, area_code, hh_size_all, per_cap_con_all) %>%
  gtsummary::tbl_summary(by = cluster,
                        missing = "no",
                        type = list(area_code = "categorical",
                                   hh_size_all = "categorical",
                                   per_cap_con_all = "continuous"),
                        statistic = list(area_code = "{n} ({p}%)",
                                       hh_size_all = "{n} ({p}%)",
                                       per_cap_con_all = "{mean} ({sd})")) %>%
  add_n() %>% # add column with total number of non-missing observations
  modify_header(label = "**Characteristic**") %>% # update the column header

```

```
bold_labels()%>%
as_kable_extra() %>% # Convert to kableExtra table
kable_styling(latex_options = "hold_position", position = "center") # Center the table in the PDF out

# Display the summary table
summary_table
```

Characteristic	N	1, N = 119	2, N = 66
area_code	185		
KMA		17 (14%)	8 (12%)
Other Town		22 (18%)	9 (14%)
Rural		80 (67%)	49 (74%)
hh_size_all	185		
1 person		0 (0%)	38 (58%)
2 person		0 (0%)	28 (42%)
3 person		29 (24%)	0 (0%)
4 or more		90 (76%)	0 (0%)
per_cap_con_all	185	62,731 (50,524)	129,553 (139,887)

¹ n (%); Mean (SD)

6 Composite Index

6.1 Recode the variables

```
SLC_2007.Subset_Index <- SLC_2007.Subset %>%
  select(kitchen_shared,toilet_shared,water_source_shared,water_meter,per_cap_con_all,hh_size_all,water_meter)

# remove NAs
SLC_2007.Subset_Index <- na.omit(SLC_2007.Subset_Index)

SLC_2007.Subset_Index <- SLC_2007.Subset_Index %>%
  mutate(
    kitchen_shared_recode = ifelse(kitchen_shared == "SHARED", 1, 0),
    toilet_shared_recode = ifelse(toilet_shared == "SHARED", 1, 0),
    water_source_shared_recode = ifelse(water_source_shared == "YES", 1, 0),
    water_meter_recode = ifelse(water_meter == "Group", 1, 0)
  )
```

6.2 Calculate the sum of the recoded variables

```

SLC_2007.Subset_Index <- SLC_2007.Subset_Index %>%
  mutate(shared_facilities_sum = kitchen_shared_recode +
    toilet_shared_recode +
    water_meter_recode
    # water_source_shared_recode
  )

# Calculate the highest possible score
highest_possible_score <- 3

# Normalize the sum to create the index
SLC_2007.Subset_Index <- SLC_2007.Subset_Index %>%
  mutate(shared_facilities_index = shared_facilities_sum / highest_possible_score)

```

6.3 Cronbach Alpha analysis

```

# List the names of the columns (variables) you will include in your index
Cron_Cols <- c("kitchen_shared_recode",
  "toilet_shared_recode",
  "water_meter_recode" )

# Create a subset of the data with only the selected columns
Cron_Data <- subset(SLC_2007.Subset_Index, select = Cron_Cols)

# Run the Cronbach Alpha analysis
Cron.Alpha <- psych::alpha(Cron_Data)

```

```

# Create the alpha table
alpha_table <- data.frame(
  raw_alpha = Cron.Alpha[["total"]][["raw_alpha"]],
  std_alpha = Cron.Alpha[["total"]][["std.alpha"]],
  G6_smc = Cron.Alpha[["total"]][["G6(smc)"]],
  average_r = Cron.Alpha[["total"]][["average_r"]],
  S_N = Cron.Alpha[["total"]][["S/N"]],
  ase = Cron.Alpha[["total"]][["ase"]],
  mean = Cron.Alpha[["total"]][["mean"]],
  sd = Cron.Alpha[["total"]][["sd"]],
  median_r = Cron.Alpha[["total"]][["median_r"]]
)

# Create a kable with the alpha table and ensure it renders correctly in the PDF output
alpha_table_kable <- kable(alpha_table, format = "latex", digits = 2, caption = "Cronbach Alpha Table")
  kable_styling(position = "center")

# Display the kable
alpha_table_kable

```

Table 1: Cronbach Alpha Table

raw_alpha	std_alpha	G6_smc	average_r	S_N	ase	mean	sd	median_r
0.65	0.66	0.61	0.39	1.95	0.05	0.21	0.31	0.33

```
# Extract relevant information from the Cron.Alpha object
alpha_stats <- data.frame(
  items = rownames(Cron.Alpha$alpha.drop),
  raw_alpha = Cron.Alpha$alpha.drop[, "raw_alpha"],
  std_alpha = Cron.Alpha$alpha.drop[, "std.alpha"],
  G6_smc = Cron.Alpha$alpha.drop[, "G6(smc)"],
  mean = Cron.Alpha$item.stats[, "mean"],
  sd = Cron.Alpha$item.stats[, "sd"]
)

# Print overall alpha values
cat("Standardized alpha:", Cron.Alpha$total$std.alpha, "\n")
```

```
## Standardized alpha: 0.661318
```

```
# Formatting and printing the alpha_stats table
formatted_alpha_stats <- alpha_stats %>%
  kable(col.names = c("Items", "Raw Alpha", "Standardized Alpha", "G6 (smc)", "Mean", "SD"),
        format = "latex", caption = "Cronbach's Alpha Analysis", align = 'c', digits = 2) %>%
  kable_styling(latex_options = "hold_position", position = "center") # Center the table in the PDF out

formatted_alpha_stats
```

Table 2: Cronbach's Alpha Analysis

Items	Raw Alpha	Standardized Alpha	G6 (smc)	Mean	SD
kitchen_shared_recode	0.49	0.49	0.33	0.14	0.34
toilet_shared_recode	0.38	0.39	0.24	0.24	0.43
water_meter_recode	0.75	0.76	0.62	0.26	0.44

```
# Check for missing values and handle them
correlation_data <- SLC_2007.Subset_Index %>%
  select(shared_facilities_index, per_cap_con_all, hh_size_all, water_bill) %>%
  na.omit()

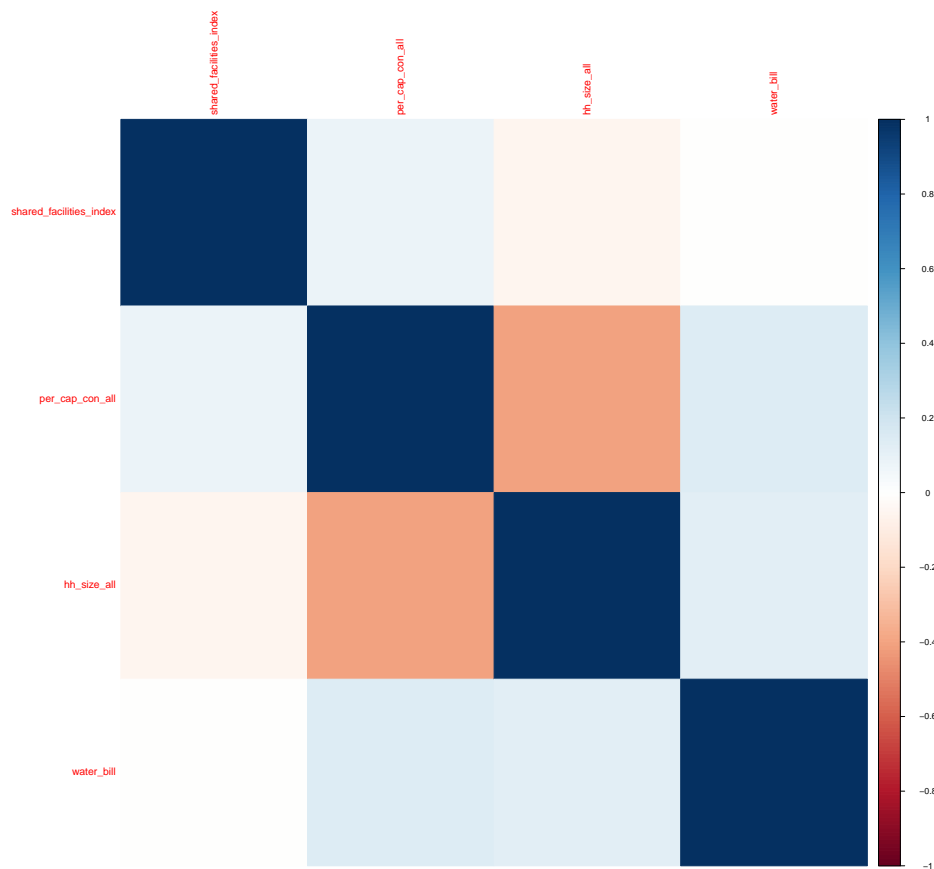
# Ensure data types are numeric
correlation_data$hh_size_all <- as.numeric(correlation_data$hh_size_all)
correlation_data$per_cap_con_all <- as.numeric(correlation_data$per_cap_con_all)
```

Table 3: Correlation Matrix

	shared_facilities_index	per_cap_con_all	hh_size_all	water_bill
shared_facilities_index	1.000	0.080	-0.056	-0.009
per_cap_con_all	0.080	1.000	-0.408	0.142
hh_size_all	-0.056	-0.408	1.000	0.127
water_bill	-0.009	0.142	0.127	1.000

```
correlation_data$water_bill <- as.numeric(correlation_data$water_bill)
# Calculate the correlation matrix
correlation_matrix <- cor(correlation_data)

# Create the correlation plot
corrplot(correlation_matrix, method = "color")
```



```
# Convert the correlation matrix to a table
correlation_table <- kable(correlation_matrix, format = "latex", digits = 3, caption = "Correlation Mat")
kable_styling(position = "center")

# Print the correlation table
correlation_table
```

7 Inferential Analysis of Composite Index

7.1 Goal 5 Is there a relationship between shared facilities index and per capita water consumption?

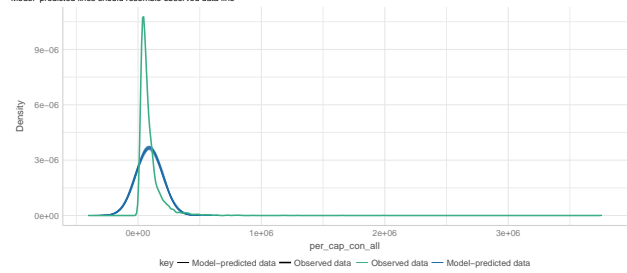
- Test: Simple Linear Regression

```
indexModel <- lm(per_cap_con_all ~ shared_facilities_index, data = SLC_2007.Subset_Index)
# tab_model(indexModel)
```

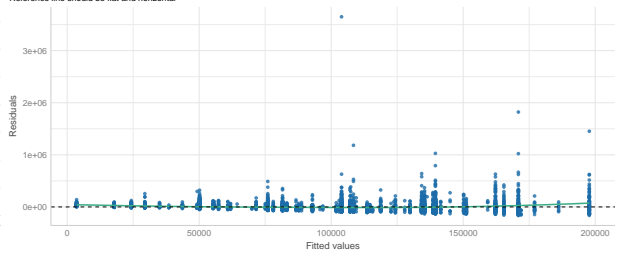
7.2 Diagnostic Plots

```
# Diagnostic Plots
check_model(model)
```

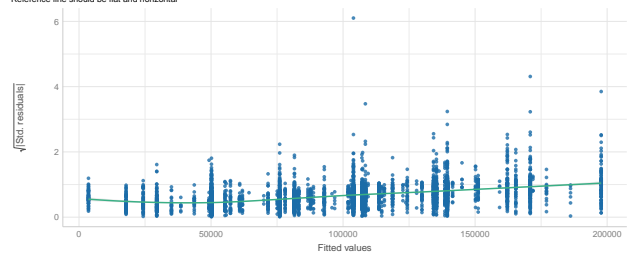
Posterior Predictive Check
Model-predicted lines should resemble observed data line



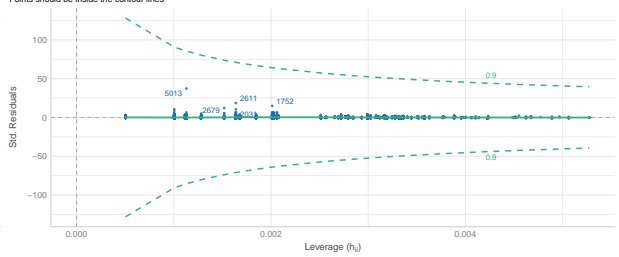
Linearity
Reference line should be flat and horizontal



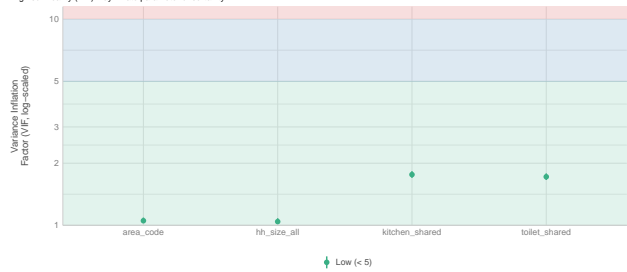
Homogeneity of Variance
Reference line should be flat and horizontal



Influential Observations
Points should be inside the contour lines



Collinearity
High collinearity (VIF) may inflate parameter uncertainty



Normality of Residuals
Dots should fall along the line

